Feature extraction for document image segmentation by pLSA model

Takuma Yamaguchi, and Minoru Maruyama
Department of Information Engineering
Shinshu University, Nagano, 380–8553, Japan
s07t213@shinshu-u.ac.jp, maruyama@cs.shinshu-u.ac.jp

Abstract

In this paper, we propose a method for document image segmentation based on pLSA (probabilistic latent semantic analysis) model. The pLSA model is originally developed for topic discovery in text analysis using "bag-of-words" document representation. The model is useful for image analysis by "bag-of-visual words" image representation. The performance of the method depends on the visual vocabulary generated by feature extraction from the document image. We compare several feature extraction and description methods, and examine the relations to segmentation performance. Through the experiments, we show accurate content-based document segmentation is made possible by using pLSA-based method.

1. Introduction

OCR (optical character recognition) systems are useful and have been used successfully. Current OCR systems are very flexible and can recognize documents written in various kinds of languages such as English, Japanese, Korean, Russian etc. Even if documents are hand-written, they can be recognized properly. Moreover, some advanced systems (e.g. InftyReader [1] developed by InftyProject [2]) can handle documents which include not only the ordinary texts but also various kinds of "objects" such as mathematical formulae, figures, tables etc. These flexible OCR systems usually consist of multiple recognition "engines", each of which is designed to handle only the limited types of objects in a document. To obtain the correct recognition result, each engine should be applied to the appropriate part of the document. It is not known in advance which engine is the most appropriate one for each part of the given document. One possible method to overcome the difficulty is to apply all the engines, which can evaluate "confidence" of their output, to each part of the document. Comparing the confidence values of multiple recognition engines, the most appropriate result may be selected. This naive strategy is, however, inefficient due to unnecessary application of inappropriate engines. Moreover, designing the method that can give rise to the confidence value of the recognition result of each engine is not easy.

In this paper, we consider to segment the document image before applying the recognition engines. If the given document is segmented into the appropriate parts, then we only have to apply single engine to each part. This could improve the efficiency of the recognition significantly. For the document image segmentation we use the probabilistic topic models [1, 2, 3, 4] originally developed for document analysis, such as document type classification, information retrieval etc. Recently topic models have been applied to visual pattern recognition problems such as image categorization, image annotation etc [5, 6, 7]. Since the probabilistic topic models are originally used in the field of document analysis, they are defined on words, documents, and corpora. To apply the models to image analysis, we have to define visual words, which are counterparts of ordinary "words" in text documents. A method to generate visual vocabulary via visual feature extraction is required.

In this paper, we propose a method for document image segmentation based on the pLSA (probabilistic latent semantic analysis) model [1, 2, 3]. The performance of the method depends on the visual vocabulary generated by feature extraction from the document image. We compare several feature extraction and description methods, including SIFT, Haar wavelet, and examine the relations to segmentation performance. With the experiments, we also show the pLSA-based method can give very good document image segmentation results.

2. pLSA model

Probabilistic Latent Semantic Analysis (pLSA) is a generative statistical model for text analysis [1, 2, 3]. The model is used to discover topics in a document with the

[1] http://www.sciaccess.net/en/InftyReader/index.html

[2] http://www.inftyproject.org/en/index.html

978-0-7695-3337-7/08 $25.00 © 2008 IEEE
DOI 10.1109/DAS.2008.48

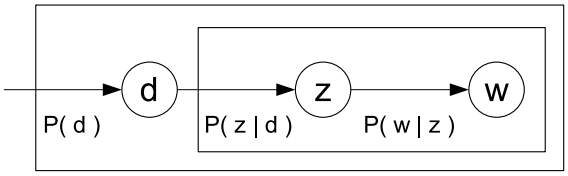


Figure 1. Graphical model representation of pLSA

bag-of-words document representation, where spatial relationships between features are ignored.

Let D be a collection of N documents $D = \{d_1, \ldots, d_N\}$. Each document d is a set of words. A word w is an element of the vocabulary $w \in W = \{w_1, \ldots, w_V\}$. Additionally, there is a hidden (latent) topic variable $z \in Z = z_1, \ldots, z_K$ associated with each occurrence of a word w in a document d. The pLSA model is parameterized by $P(w|z)$ and $P(z|d)$. The document is generated as follows:

1. A document d is selected with probability $P(d)$.
2. For each word in the document, a topic z is selected with $P(z|d)$.
3. A word w is generated with probability $P(w|z)$.

It is assumed that the distribution of words given a latent topic z, $P(w|z)$ is conditionally independent of the document. The probabilistic graphical model of pLSA is shown in figure 1. Marginalizing over topics z following joint probability is obtained.

$$P(w, d) = P(d) \sum_{z \in Z} P(w|z)P(z|d), \tag{1}$$

The model parameters $P(w|z)$ and $P(z|d)$ are estimated by maximizing the data log-likelihood using an Expectation Maximization (EM) algorithm [8]. The log-likelihood is given by

$$\begin{aligned} \mathcal{L} &= \sum_{d \in D} \sum_{w \in W} n(w, d) \log P(w, d) \\ &= \sum_{d \in D} \sum_{w \in W} n(w, d) \log P(d) \\ &\quad + \sum_{d \in D} \sum_{w \in W} n(w, d) \log \sum_{z \in Z} P(w|z)P(z|d), \end{aligned} \tag{2}$$

where $n(w, d)$ stores the number of occurrences of a word w in document d. EM algorithm has two steps. The first is an expectation step (E-Step), where posterior probabilities are computed for the latent variables, based on the current estimates of the parameters. The second is a maximization step (M-Step), where parameters are updated based on the so-called expected complete data log-likelihood which depends on the posterior probabilities computed in the E-Step. The EM algorithm for pLSA is:

E-Step:

$$P(z|w, d) = \frac{P(w|z)P(z|d)}{\sum_{z \in Z} P(w|z)P(z|d)} \tag{3}$$

M-Step:

$$P(w|z) = \frac{\sum_{d \in D} n(w, d)P(z|w, d)}{\sum_{w \in W} \sum_{d \in D} n(w, d)P(z|w, d)} \tag{4}$$

$$P(z|d) = \frac{\sum_{w \in W} n(w, d)P(z|w, d)}{\sum_{z \in Z} \sum_{w \in W} n(w, d)P(z|w, d)} \tag{5}$$

After training, the estimated $P(w|z)$ parameters are used to estimate $P(z|d_{new})$ for new documents d_{new} through a "folding-in" method. In the folding-in process, EM algorithm is used in a similar manner to the training process. The folding-in method for pLSA is:

E-Step:

$$P(z|w, d_{new}) = \frac{P(w|z)P(z|d_{new})}{\sum_{z \in Z} P(w|z)P(z|d_{new})} \tag{6}$$

M-Step:

$$P(z|d_{new}) = \frac{\sum_{w \in W} n(w, d_{new})P(z|w, d_{new})}{\sum_{z \in Z} \sum_{w \in W} n(w, d_{new})P(z|w, d_{new})} \tag{7}$$

where $P(w|z)$ is kept fixed.

3. Image representation

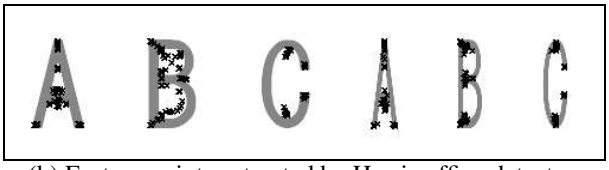
To apply the pLSA model to images, visual words should be detected based on the image feature extraction. The image representation, which consists of a set of visual words, is derived through extracting feature points in an image, and then describing the appearance around the feature points.

3.1. Feature point detection

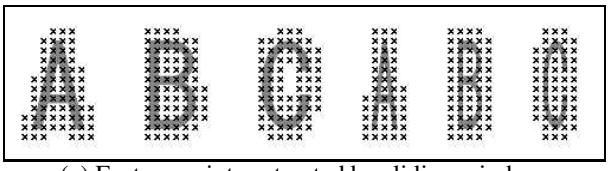
In this research, two types of feature extraction methods are examined. The first is Harris-affine interest point detector [9]. The detecting algorithm relies on the combination of corner points detected thorough Harris corner detection [10], multi-scale analysis through Gaussian scale-space and affine normalization using an iterative affine shape adaptation algorithm [11]. Software of Harris-affine interest point detector is available from the Oxford University Visual Geometry Group [3]. The method can give rise to the

[3] http://www.robots.ox.ac.uk/~vgg/research/affine/

(a) Sample input image.



(b) Feature points extracted by Harris-affine detector.



(c) Feature points extracted by sliding window.

Figure 2. Examples of feature point detection.

stable sparse representation which is expected to be robust to changes in scale and translation. The second is a dense feature extraction by sliding a window. The window is sampled at every 4 pixels. A center point of the window is selected as a feature point. If there is no black pixels in a window, the feature detection is discarded.In this research, the window size is 16×16 pixels. The detected feature points of a sample image (Figure 2-(a)) are shown in Figure 2-(b) and Figure 2-(c).

3.2. Feature description

We use two different representations for describing appearance around the feature points, SIFT (Scale Invariant Feature Transform) descriptor [12] and Haar wavelet [13]. SIFT descriptors are derived from windowed histograms of gradient magnitudes at varying locations and orientations, normalized to correct for contrast and saturation effects. This approach provides some invariance to lighting and poses changes. We use 128 dimensional SIFT descriptor. The binary of Oxford University Visual Geometry Group can also calculate the SIFT features on interest points which extracted by Harris-affine detector. Haar wavelet is a powerful image feature for object recognition. The 2 dimensional Haar decomposition of a square image with n^2 pixels consists of n^2 wavelet coefficients. Since we use a 16×16 pixels search window, it is represented as a 256 dimensional vector.

3.3. Visual words and visual vocabulary

PLSA model is applied to images by using a visual analogue of a word. The visual vocabulary is obtained by vector quantization of image features. We use k-means clustering for the vector quantization. First, k-means algorithm is applied to image features of all training data. The features in a same cluster are treated as a same visual word, and the centers of each cluster are the representatives of the visual words. Therefore, the number of clusters is the size of the visual vocabulary. Testing data is transformed to bag-of-words representation using the visual vocabulary which obtained from training data.

3.4. Image segmentation

The goal of our work is to segment a document image into regions so that the categories of the items in each region are the same. For that purpose, after image feature extraction described above, grouping of the feature points is carried out based on the proximity of their spatial proximity. Each group, which is represented as a set of visual words, is treated as a document. PLSA model is defined on these documents. In this paper, we use both k-means clustering and grid-based method for grouping of feature points. K-means clustering is applied to the set of feature points $\{(x_i, y_i)\}$. In the grid-based method, a document image is divided into a set of windows. Each group (i.e. document) is made up of feature points within a same window. The size of the each window is 300×300 pixels. After document extraction, each document is classified.

4. Classifier

We use three different methods for classification of extracted "documents" in images. The first is k-nearest neighbors (k-NN) algorithm with Euclidean distance function. A "document" is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest training samples. The second is Support Vector Machine (SVM). SVM is a learning method based on margin maximization principle. SVM performs binary classification by finding optimal separating hyperplane in feature space. Suppose that a set of training examples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ are given, the SVM classify the input $\boldsymbol{x}$ based on following function

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i) - b, \tag{8}$$

where $K(\boldsymbol{x}, \boldsymbol{y})$ is a kernel function which defines the in-

ner product in the feature space. Coefficients α_i are non-zero only for the subset of the input data called support vectors. The performance of SVM depends on the kernel. We use Gaussian radial basis function, which outperformed the other commonly used kernels in the preliminary experiments. The kernel is given as

$$K(\boldsymbol{x}, \boldsymbol{y}) = \exp\left(-\frac{||\boldsymbol{x} - \boldsymbol{y}||^2}{2\sigma^2}\right) \quad (9)$$

Since the problem is multi-class classification problem, we employ one-vs-rest method. For each class, an SVM is learned to discriminate that a class from others. The best class is decided by the SVM that gives the highest value.

The last is a pLSA-based classifier. With this method, when a novel document d_{new} is given, classification is carried out based on the probability $P(class|d_{new})$, which is derived via pLSA model. A "document" is assigned to the class with highest probability $P(class|d_{new})$, which is computed as follows:

$$P(class|d_{new}) = \sum_{z \in Z} P(class|z)P(z|d_{new}) \quad (10)$$

where we assume $p(class|z, d) = p(class|z)$ holds for any document d. For novel document d_{new}, $p(z|d_{new})$ is obtained by fold-in procedure described in section 2. $p(class|z)$ is estimated from labeled examples as follows :

$$p(class|z) \propto p(z|class)p(class) \quad (11)$$

We approximate $p(z|class)$, $p(class)$ as follows:

$$p(class) \approx \frac{N_c}{N}, \quad (12)$$

$$p(z|class) \approx \frac{1}{N_c} \sum_{\{i|category(d_i)=c\}} p(z|d_i) \quad (13)$$

where N_c is the number of documents of category c, and N is the total number of examples.

In this paper, we assume that each document is subject to a single category. However, even if a document consists of elements of multiple categories, the proposed method can be extended so that the classification is carried out based on $P(class|w, d_{new})$. The $P(class|w, d_{new})$ is given by:

$$P(class|w, d_{new}) = \sum_{z \in Z} P(class|z)P(z|w, d_{new}), \quad (14)$$

where the $P(class|z)$ and the $P(z|w, d_{new})$ are obtained from Eq. (11) and (6) respectively.

5. Experiments of document image segmentation

5.1. Datasets

We collected scanned images of mathematical formulas, printed Japanese, printed English and hand written texts from scientific papers. These images are 6M–9M pixels, and the resolution is 300 dpi. The number of images of each type is shown in Table 5.1, and the sample images are shown in figure 3-(a)–(d). The mathematical formula image consists of a number of mathematical formulas that are extracted from some papers. The printed Japanese images may include mathematical formulas and English words. We exclude the areas which contain too many English words or mathematical formulas, from the printed Japanese data sets. We also remove mathematical formula areas in printed English images. Additionally, graph and picture regions are erased from all images. Feature point detection, and then "document" detection by k-means clustering and grid-based method are applied to the images. The number of k-means clusters is decided so that the mean value of the number of interest points in documents is around 1,000. The number of "documents" extracted from the images depends on the choice of feature detector. When Harris-affine detector is applied, about 5,000 "documents" are obtained from 78 images.

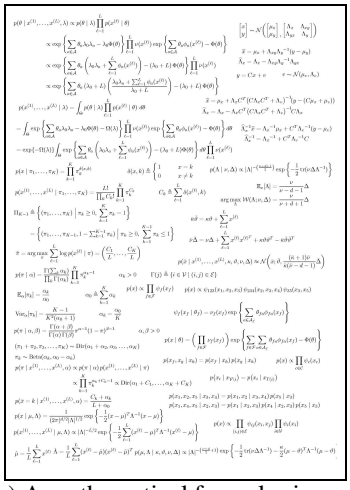
Table 1. The number of images of each category.

	# of images
mathematical formula	12
printed Japanese paper	34
printed English paper	15
hand written paper	17
	78

Throughout our experiments, the "document" categories we consider are {printed Japanese, printed English, math formula, handwritten Japanese}.

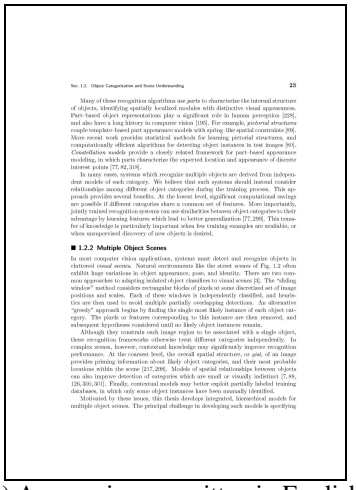
5.2. Image representations

To decide optimal image representation for document image segmentation with pLSA and the folding-in process, we compared the feature point detectors, the feature descriptors and the grouping methods for "document" extraction. Training data contains 400 document images (each class includes 100 documents), and the number of documents in test data is 1,200 (300 documents each). Since



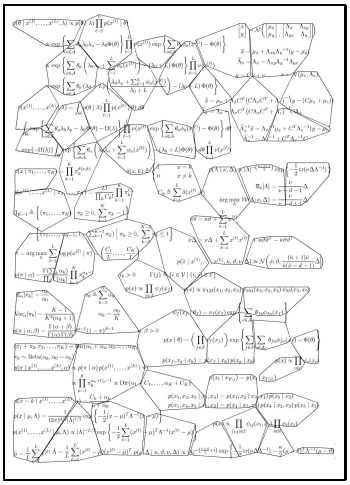
(a) A mathematical formulas image.

(b) A paper image written in Japanese.

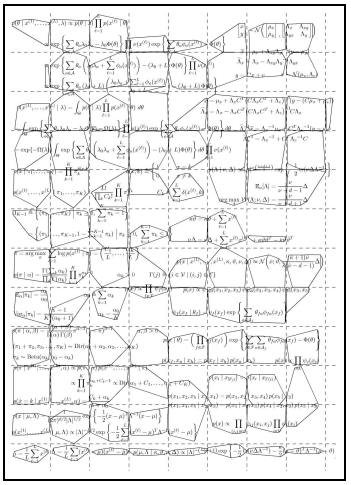


(c) A paper image written in English.

(d) A handwritten paper image.



(e) Visual documents by k-means.



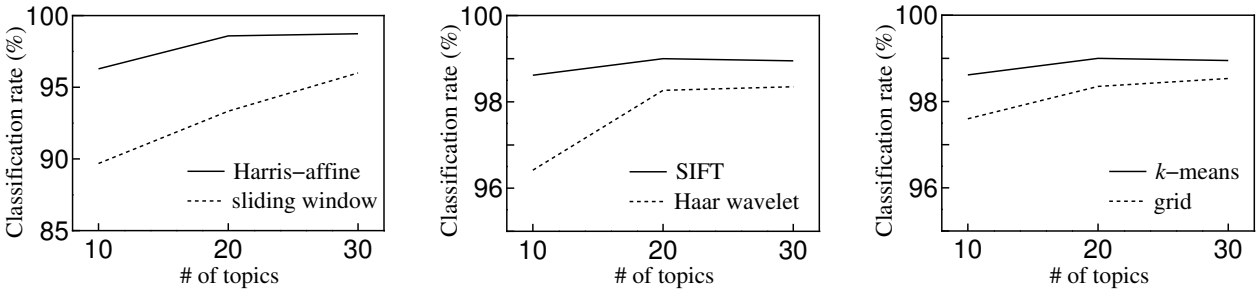
(f) Visual documents by 300×300 grid.

Figure 3. Sample images and visual documents.

we set the size of visual vocabulary to 500, each document is expressed as 500 dimensional vector. The iteration count of the EM algorithm is 100 for pLSA. The class label of testing documents are decided with $P(class|d_{test})$ (pLSA-based classifier). In this paper, all classification results are obtained by means of 5 times cross-validations. Performance comparison by different feature point detectors is shown in Figure 4-(a). Harris-affine detector outperforms the sliding-window. Figure 4-(b) shows difference of efficiency between feature descriptors, SIFT descriptor is better than Haar wavelet description. Creating document representation with k-means gives good results than fixed grid (Figure 4-(c)). In our experiments, "documents" which have a small number of visual words (feature points) are rejected. Possibility of such "documents" are extracted with fixed grid is higher than k-means. Therefore, using k-means is practical. We have reached a conclusion that Harris-affine interest point detector, SIFT descriptor and creating documents by k-means clustering, are useful for pLSA image representation.

5.3. Classifiers

Next, we consider the classifiers. Training data includes from 100 to 400 documents, the number of "documents" in testing data is 2,000. Figure 5 shows the recognition results using pLSA-based classifier. If the number of topics is 4 (that is equal to the number of categories), good recognition rate was not obtained. This result suggests "topic" is not necessarily same as "category". Thus, it is difficult to classify documents with $P(z|d)$. For classification task,



(a) Harris-affine vs. sliding window. (b) SIFT vs. Haar wavelet. (c) k-means vs. grid.

Figure 4. Comparison of classification performance.

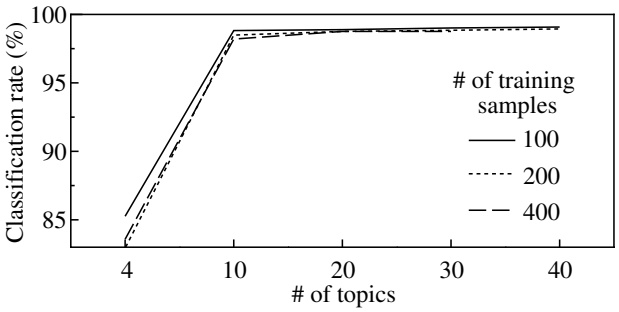


Figure 5. Classification results by pLSA-based classifier.

it is appropriate to use the model in which a document in each category can have words generated by multiple topics. Table 2 shows results with three classifiers, SVM, k-NN ($k = 5$) and pLSA-based classifier (the number of topic is 20). The highest rate is obtained by pLSA-based classifier, but the computation time is the worst. This is the result with the fixed number of (100) iterations in EM algorithm (folding-in procedure). It is possible to reduce the computation time by judging converge of EM algorithm.

Table 2. Results by three classifiers and the computation time.

	# of training samples			computation
	100	200	400	time [s]
pLSA	98.9%	98.8%	98.8%	14.8
SVM	98.5%	98.5%	98.8%	3.5, 5.2, 7.6
k-NN	97.1%	97.9%	98.4%	2.9

5.4. Topic based document representation

In the previous experiments, document is represented by 500 dimensional (word frequency) vector. Another representation can be obtained by using topic distribution of the document. With the representation, each document is represented as a K-dimensional vector $(P(z_1|d), \ldots, P(z_K|d))$, where K is the number of topics. Usually vocabulary size V is much less than K (#topics). In this case, dimensionality of the resultant vector can be greatly reduced. Figure 6 and 7 shows the results with SVM and k-NN via the topic representation, where the number of topics is from 10 to 40. The results are marginally improved with k-NN and the topic representation, but got worse with SVM.

Figure 8 shows an example of image segmentation by pLSA-based classifier.

6. Conclusions

In this paper, we have described a method for document image segmentation based on pLSA model. To apply the pLSA model, it is necessary to represent images by "bag-of-visual words". The performance depends on the representation generated by feature extraction from the images. We compare several feature extraction and description methods, and examine the relations to segmentation performance. We have reached a conclusion that Harris-affine interest point detector, SIFT descriptor and creating documents by k-means clustering, are useful for image representation. In our experiments, good document image segmentation results are obtained with the pLSA-based method. When the method is used to select engines for document image recognition, it is preferable that the computation time is much faster. Development of the fast method for building visual document/word representation is the future work.

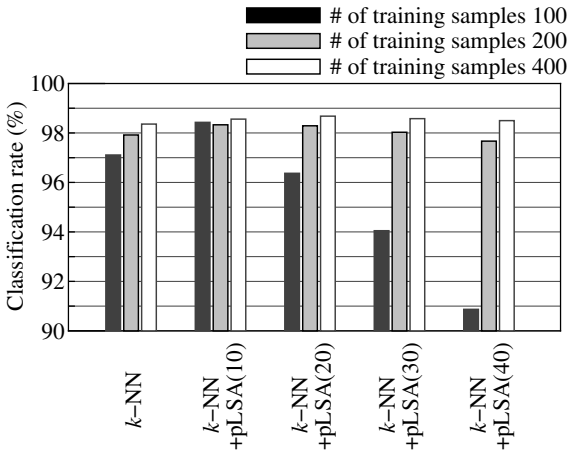


Figure 6. K-NN with topic-based representation.

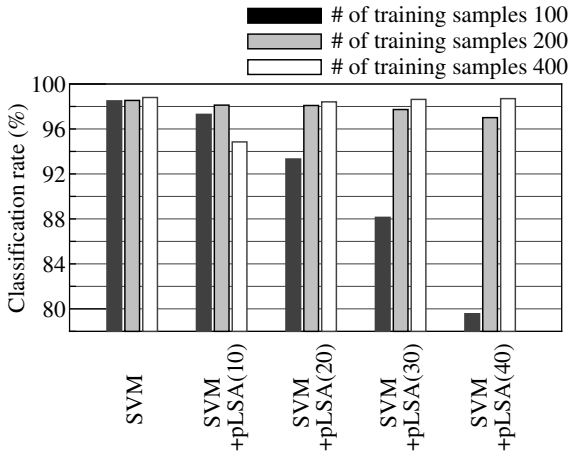


Figure 7. SVM with topic-based representation.

References

[1] T. Hofmann, "Probabilistic latent semantic indexing", *Proceedings of Special Interest Group on Information Retrieval (SIGIR), Berkeley, CA, USA*, 1999, pp. 50–57.

[2] T. Hofmann, "Probabilistic latent semantic analysis", *Proceedings of Uncertainty in Artificial Intelligence (UAI), Stockholm, Sweden*, 1999, pp. 289–296.

[3] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis", *Machine Learning*, 42, 2001, pp. 177–196.

[4] D. Blei, A. Ng and M. Jordan, "Latent dirichlet allocation", *Journal of Machine Learning Research*, 3, 2003, pp. 993–1022.

[5] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories", *Proceedings of Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA*, 2005, pp. 524–531.

[6] J. Sivic, B. Russell, A. Efros, A. Zisserman and W. Freeman, "Discovering objects and their location in images", *Proceedings of International Conference on Computer Vision (ICCV), Beijing, China*, 2005.

[7] A. Bosch, A. Zisserman and X. Munoz, "Scene classification via plsa", *Proceedings of the European Conference on Computer Vision (ECCV), Graz, Austria*, 2006.

[8] A. Dempster, N. Laird and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm", *Journal of the Royal Statistical Society B*, 39(1), 1977, pp. 1–38.

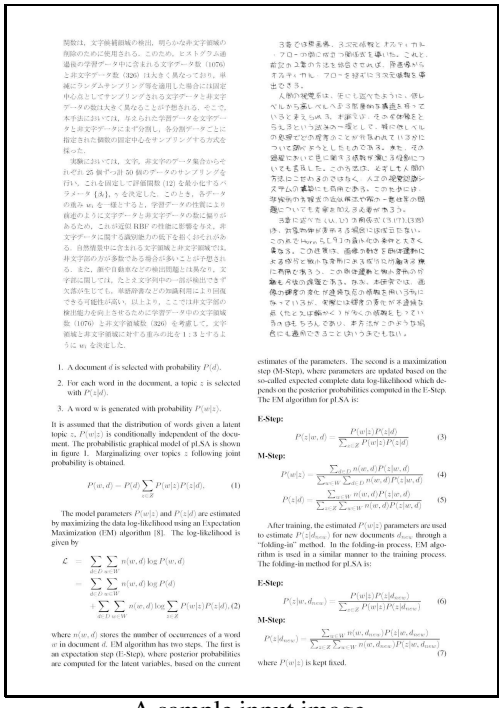
[9] K. Mikolajczykm and C. Schmid, "Scale & affine invariant interest point detector", *International Journal on Computer Vision (IJCV)*, 60(1), 2004, pp. 63–86.

[10] C. Harris and M. Stephens, "A combined corner and edge detector", *In Alvey Vision Conference*, 1988, pp. 147–151.

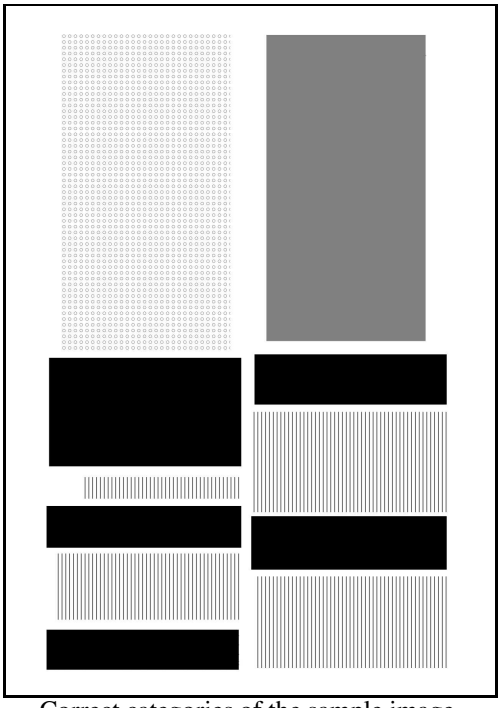
[11] T. Lindeberg and J. Garding, "Shape-adapted smoothing in estimation of 3-d shape cues from affine deformations of local 2-d brightness structure", *Image and Vision Computing*, 15(6), 1997, pp. 415–434.

[12] D. Lowe, "Distinctive image features from scale-invariant keypoints", *International Journal of Computer Vision (IJCV)*, 60(2), 2004, pp. 91–110.

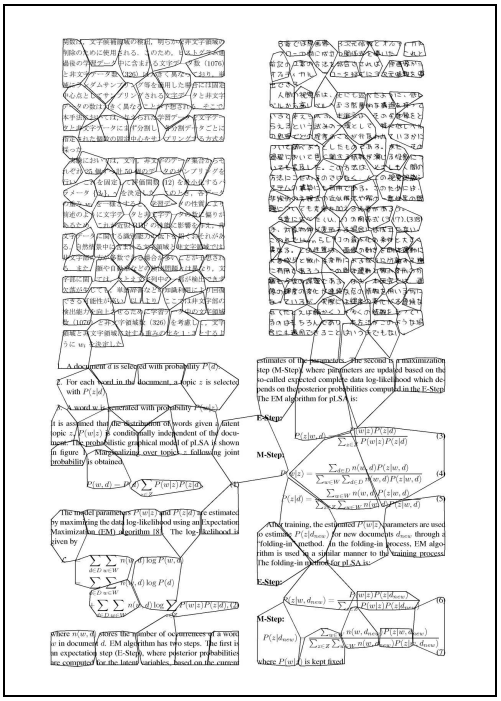
[13] E. Stollnitz, T. DeRose and D. Salesin, "Wavelet for computer graphics : A primer, part 1", *IEEE Computer Graphics and Applications*, 15(3), 1995, pp. 76–84.



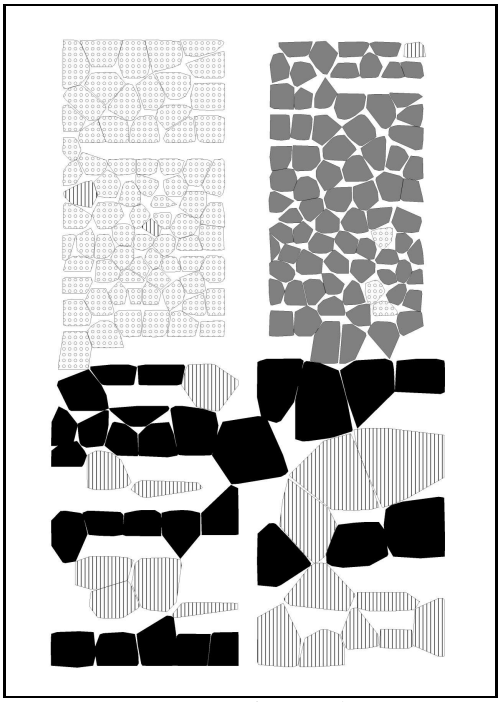
A sample input image.



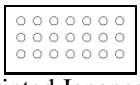
Correct categories of the sample image.



Extracted visual documents.



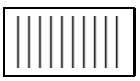
Segmentation result.



Printed Japanese.

Handwritten Japanese.

Printed English.



Mathematical formulae.

Figure 8. An example of image segmentation by pLSA-based classifier.