BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **17**, No 2
Sofia • 2017 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.1515/cait-2017-0020

Content-Based Image Retrieval for Multiple Objects Search

Gábor Szűcs, Dávid Papp

Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, 2nd Magyar Tudósok Krt., H-1117 Budapest, Hungary
E-mails: szucs@tmit.bme.hu pappd@tmit.bme.hu

Abstract: *The progress of image search engines still proceeds, but there are some challenges yet in complex queries. In this paper, we present a new semantic image search system, which is capable of multiple object retrieval using only visual content of the images. We have used the state-of-the-art image processing methods prior to the search, such as Fisher-vector and C-SVC classifier, in order to semantically classify images containing multiple objects. The results of this offline classification are stored for the latter search task. We have elaborated more search methods for combining the results of binary classifiers of objects in images. Our search methods use confidence values of object classifiers and after the evaluation, the best method is selected for thorough analysis. Our solution is compared with the famous web images search engines* (*Google, Bing and Flickr*)*, and there is a comparison of their Mean Average Precision* (*MAP*) *values. It can be concluded that our system reaches the benchmark; moreover, in most cases our method outperforms the others, especially in the cases of queries with many objects.*

Keywords: *Visual content, image search, classification, search engine, multiple objects.*

1. Introduction

Images have been used for many years in many areas like press, advertising, medical fields, spatial data management, etc. But in the past few years, the number of images has increased in huge amount due to the growth of internet. So, in large and varied collection, users of different domains face a problem of retrieving images relevant to the user query. Thus, Content-Based Image Retrieval (CBIR) has received considerable attention as a consequence of collections with tremendous amounts of multimedia contents. There is a growing interest on CBIR and in more general Content-Based Multimedia Retrieval [11, 29] from both academia, e.g., “Image FARMER” [3], and industry, such as Flickr, Bing and Google images search products on the Web. In a recent paper [23] the different methods of image retrieval systems and major categories of the state-of-the-art techniques are presented; furthermore, there is a survey [13] including many publications to describe the research aspects in this area such as feature extraction, feature matching, semantic gap reduction and measurements for performance evaluation.

In the image retrieval, the images are indexed on the base of the text that is related to the images. Queries are matched to this text to produce a set of search results, but many irrelevant images can be found in these results [12]. For improvement of image search a basic mechanism, so called Relevance Feedback [28, 31], can be used; state-of-the-art method Hybrid Feedback mechanism [32] – based on both images and their attributes – also provides possibility to enhancement. However, in our work we have not used any feedback from end users, because our aim was to develop an image search system without the user´s help.

Hybrid Intelligent Systems (HIS) are free combinations of computational intelligence techniques to solve a given problem, covering all computational phases from data normalization up to final decision making. The HIS can be used in CBIR as well, for example, in order to create hybrid information descriptors [33]. Multi-Classifier Systems (MCS), as subcategory of HIS [30] focus on the combination of classifiers form heterogeneous or homogeneous modelling backgrounds to give the final decision, and our work belongs to this subcategory.

MCS is able to solve different classification tasks, as medical data classification [25], medical image retrieval [21], image annotation [19], and image retrieval [33]. In Multiple Queries for Image Retrieval (MQIR) systems [2] the end user would like to retrieve an interesting image by multiple queries (e.g., by more query images), which allows for a more expressive formulation of the query object, including different viewpoints and/or viewing conditions. This MQIR is investigated in work [10], but our goal is to retrieve not only one object with more viewpoints, but also multi objects. There is already an existing solution for multi-object images retrieval by fusing several features, but only for binary images [16].

The aim of our work was to research and implement an image search system based on only visual information for large variety of images. We planned this system to be capable of retrieving different visual objects and more objects together by a hybrid solution. Our work differs from previous works mentioned above, because we have focused on multiple objects. In the literature, there is another similar work – using Multiple Query Basic Matching (MQBM) method [10] – dealing with multiple query, so the query can contain more sample images; but in our work, we use only text queries instead of image queries. At the end of our research we have implemented and tested this solution. In the next sections the results of our work will be presented.

2. Offline image classification

The above-mentioned systems (CBIR systems) are in the new trend of computer vision beside the traditional trend of computer vision in robotics. Our work also belongs to this new trend, where the classification of images is a central problem [7].

We have elaborated an image classification method for image-based plant identification problem [26], and this offline classifier has been developed further for image search. Our implemented system deals with semantic image search, where visual objects (vehicles) are the goals of the queries; furthermore, metadata were not available for search, only the content information of images can be used. Since the aim was the search on unknown photos, it was necessary to have a training data on the base of which machine learning methods were used to analyse the images. Thereafter, the results from the analysis were available to search. The semantic information is stored in a database before the query (offline); the preparation of this offline part is discussed in this chapter.

2.1. Learning phase

As mentioned above, machine learning methods were used to train from a sample set, which consists of two image sets, called training and validation image set. The latter can be used for calibration of the trained model during the validation phase of the training procedure. The first step is the representation of the images based on their content (pixels, shapes and textures) and semantic information. Therefore, a usual technique in computer vision is used to represent an image, the BoW (Bag-of-Words) model [9, 15], where images are treated as documents. According to this, visual "words" ("code-words") in images need to be defined, which can be achieved by the following steps:

- Feature detection.
- Feature description.
- "Codebook" generation.

Many different feature types can be detected in an image, e.g., edges, corners, ridges, as "interesting" part of an image. As feature detection, dense sampling method is used in our solution. After that, a local image patch around every feature was extracted by SIFT (Scale-Invariant Feature Transform) algorithm [17, 18]. Many possible feature extraction methods are available for images, but we have chosen SIFT, because this is a widely-used method in practice and in theoretical works (as well) with some possible further development of this method, like RootSIFT [2]. Subsequently, PCA (Principal Component Analysis) [1, 14] reduces the dimensions of the descriptor vectors from 128 to 80. The final step in the offline classification is the conversion of descriptor vectors to code-words, which also produces a codebook. In this phase, GMM (Gaussian Mixture Model) [22, 27] is trained to determine the codebook. It is a parametric probability density function represented as a weighted sum of (in our case 256) Gaussian component densities. GMM parameters were estimated on the base of the training set by using the iterative EM (Expectation Maximization) algorithm, but an initial model is needed for EM. In our training procedure, the k-means clustering was performed over all the vectors with 256 clusters, which resulted the initial model for EM.

As a result of the above algorithms, a codebook with 256 code-words was available for further calculations, which can be considered as a concise representation of the image set. According to the codebook it is possible to create a descriptor that specifies the distribution of the visual code-words in any image, called high-level descriptor. To represent an image with high-level descriptor, the GMM based Fisher vector [24] has been calculated. In order to present the details about the generation of the high-level descriptor, the explanation of the GMM based Fisher vector is described below.

Let $X = \{x_t,\ t = 1,\dots,\ T\}$ be the set of T local descriptors extracted from an image. We assume that the generation process of X can be modelled by a probability density function p with parameters λ. Set X can be described by the gradient vector, the gradient of the log-likelihood describes the contribution of the parameters to the generation process:

$$(1)\qquad \nabla_\lambda \log p(X|\lambda).$$

Intuitively, the gradient of the log-likelihood describes the direction in which parameters should be modified to best fit the data. This gradient vector can then be classified using any discriminative classifier; the Fisher information matrix F_λ is suggested for this purpose:

$$(2)\qquad F_\lambda = E_x\left[\nabla_\lambda \log p(X|\lambda)\, \nabla_\lambda \log p(X|\lambda)'\right].$$

Fisher kernels can be used on visual vocabularies, where the vocabularies of visual words are represented by means of a GMM. The λ set of parameters of the GMM contains the weight (w_i), mean vector (μ_i) and covariance matrix (Σ_i) of Gaussian i, $i = 1, 2, \dots, N$. Each Gaussian represents a word of the visual vocabulary: w_i encodes the relative frequency of word i, μ_i – the mean of the word, and Σ_i – the variation around the mean. We denote

$$(3)\qquad L(X|\lambda) = \log p(X|\lambda).$$

Under an independence assumption, it can be derived

$$(4)\qquad L(X|\lambda) = \sum_{i=1}^{N} \log p(x_i|\lambda).$$

We assume that the covariance matrices are diagonal (as any distribution can be approximated by a weighted sum of Gaussians with diagonal covariance). We can use the notation $\sigma_i^2 = \mathrm{diag}(\Sigma_i)$. After this assumption, the partial derivatives of L can be determined with respect to all the parameters (w_i, μ_i, σ_i), and the gradient vector is just a concatenation of these partial derivatives [20]. We have used these vectors in our work as the final representation of the images.

In order to train a classification model (classifier) based on training image set, a variation of SVM (Support Vector Machine) is used, the C-SVC (C-Support Vector Classification) [6] with RBF (Radial Basis Function) kernel [4]. The SVM is basically a binary linear classifier, thus, in order to extend it to a number of classified categories, the one-against-all technique is used. During this method, a binary classifier is created for each category in the training set.

The two hyper-parameters (C from C-SVC and γ from RBF kernel) were optimized by a grid search with two-dimensional grid. The algorithm was trained with the training image set, and then validated on the validation set, while in each iteration the hyper-parameters were different. The parameter pair that gave the best result is selected to train the final classification model (for each category) based on the whole image set.

This learning phase is state-of-the-art in image classification. There are some other solutions in the literature, and these are compared in work [5]. Possibly, some other methods would have been applied, but based on our experience [26] we agree with the literature that the method described above is one of the most accurate object classification methods.

2.2. Preparing images in the search space

In this phase, the images from the search space (available unknown pictures) were examined, and semantic predictions (class labels) were calculated about them based on the final classification model; the labels were stored in the offline database in a specific structure. The details of this preparation phase are described below.

Firstly, the Fisher vectors of each image were computed, as it was discussed in the previous section. The codebook was already available, i.e. it was not required to generate a new one, since our purpose was to classify the pictures in the search space based on the code-words from the training set.

Then, an RBF based kernel matrix was built from the Fisher vectors of the available unknown pictures and training images. Each C-SVC classifier used this matrix and the hyper-parameters were the same as in the final classification models (for each classification tasks). In addition, since each classifier is assigned to a category, the generated model for a classifier is responsible to separate the designated category from the other ones. Thus, a classifier is able to provide a confidence value, which shows a certainty of the category in a given picture.

Finally, each image in the search space was classified with each classifier. As a result, each image was assigned to a vector with φ elements, where φ denotes the number of trained categories. Each element of a vector contains the confidence value given by the appropriate classifier. These vectors were stored in the offline database.

3. Content-based image searching methods

Our content-based image search system is specialized in search queries that are obtained by combining multiple objects (in the query, the names of the objects can be used). The set of searchable objects is enclosed, because the number of trained categories is constant. The task of image searching is the ranking of the images according to their relevance to the given search query.

Our contribution is the set of developed and implemented methods by using the available offline semantic information that resolves the image search. The difference between these image retrieval methods is how to calculate the relevance value of an image from the confidence values provided by the binary classifiers. The reason we have created a number of methods is that their results can be compared, so the most effective one can be selected. The developed and implemented content-based methods are described below.

3.1. Sum methods

This method computes the relevance value by sum of the corresponding confidence values of the objects that are specified in the search query. There are two variations of this method.

S1: There is not any kind of filtering before the ranking, so the search space includes all of the images. The sorting of the returned result list is based on the following aggregate variable in descending order:

(5) $$p(k) = \sum_{i=1} p_i(k) \middle|\ i \in \text{query},$$

where *i* denotes the identity of the object that is included in the "query" expression, *k* is the index of the image and $p_i(k)$ is the designated confidence value.

S2: The ranking is preceded by filtering. The filtering condition is that at least one of the confidence values belonging to the query objects should be greater than a given threshold. This can reduce the size of the search space. The aggregate variable that determining the ranking is computed as follows:

(6) $$p(k) = \begin{cases} \sum_i p_i(k) \big|\ i \in \text{query}, \exists p_i(k) > \text{threshold}), \\ \quad 0 \qquad\qquad\qquad \text{otherwise.} \end{cases}$$

3.2. Product methods

This method is similar to the sum method, but the relevance value is obtained by multiplying the confidence values. There are two versions of it as well.

P1: Similar to the S1 sub-method, the P1 does not perform any filtering on the pictures of the search space. The sorting of the returned result list is based on the following aggregate variable in descending order:

(7) $$p(k) = \prod_i p_i(k) \big|\ i \in \text{query},$$

where *i*, *k* and $p_i(k)$ represent the same as stated previously.

P2: The filtering condition is that each of the designated confidence values should exceed a specified threshold. In this case, the relevance value computed as follows:

(8) $$p(k) = \begin{cases} \prod_i p_i(k) \big|\ i \in \text{query}, \forall p_i(k) > \text{threshold}), \\ \quad 0 \qquad\qquad\qquad \text{otherwise.} \end{cases}$$

3.3. Minimum method

As the name suggest, as relevance value of an image, this method selects the lowest of the corresponding confidence values of the objects that are specified in the search query. The notation of the method is only briefly MIN, and it can only be used without filtering. The returned result list is based on the following aggregate variable in descending order:

(9) $$p(k) = \min\{p_i(k)\} \ \big|\ i \in \text{query}.$$

3.4. Multi-label classification with S1

This is the most complex method. First, a multi-label classification helps to determine which trained categories appear in a given image. A general approach is used in this part of the algorithm. The following steps should be executed for each image before constructing aggregate variable:

1. The algorithm starts with an empty list called *L*, and with another list called *K*, which contains the trained categories.
2. The algorithm selects the highest confidence value. This value represents the category, which is the most likely to occur in the given picture. This category is added to *L*, and removed from *K*.
3. Until *K* is not empty:
   a. the highest confidence value of the remaining categories is selected, then its category is removed from *K*;
   b. if this confidence value exceeds 75% of the previously selected one, then its category is added to *L*, otherwise *L* remains unchanged and the cycle is interrupted.
4. Finally, *L* contains the predicted categories for the examined image.

Then, *N* containers are formed, where *N* is the number of objects in the search query. The containers are numbered from 1 to *N*. After that, each image was placed in the appropriate container, based on the following: the elements number of the *L* list and the number assigned to the container should be equal. According to this, images with empty *L* list were excluded from the search space. Furthermore, the empty containers were also removed. As a result, a semi-sorted ranking arises. The last step is a further sorting by a modified version of the S1 sub-method in which the images do not move between the containers. The returned result list is the content of the remaining containers in descending order. The method is briefly denoted as ML.

3.5. Difference method

In this method, the maximum value of certainty (like the probability: 1) were used to compute the complement of the confidence values. The obtained values were multiplied and then complemented again. This method can only be used without filtering, and the relevance value can be calculated as follows:

$$p(k) = 1 - \prod_{i} (1 - p_i(k)) \Big| \; i \in \text{query}, \tag{10}$$

where i, k and $p_i(k)$ represent the same as stated previously. In short, it is called DIF.

4. Experiment for image search

The goodness of the implemented system was tested by a concrete image search experiment. The results were compared to the results of three well-known web image search engines (Google, Bing and Flickr). These engines are mostly text-based retrieval systems, except Google, which is able to search by query images as well. At search by a query sample image this search engine retrieves similar pictures based on the content, so Google can be considered as text-based and content-based image retrieval systems simultaneously, while our developed system uses only visual content information.

4.1. Search queries

Since we dealt with combined image search, the search queries are made up of multiple components (this is the basic unit of a query). During the experiment, seven different search queries were used, where a component indicates exactly one object. Five of them consist of two, and two of them consist of three components, as follows: {airplane, bus}, {airplane, car}, {bus, car}, {bus, motorbike}, {car, motorbike}, {airplane, bus, car}, {bus, car, motorbike}. In the case of S2 the threshold was set to 0.5, and 0.1 in the case of P2.

4.2. Training and test data

The image set used for training came from the website of Pascal VOC competition [8]. The data set published in 2007 was downloaded for our research. These pictures were originally derived from Flickr. This data set contains images from 20 different categories, including the ones mentioned above. It is advisable to train categories which are included in neither one of the search queries. It is necessary, because the non-relevant part of the search space should be also covered by training. Therefore, additional categories were added to the training data, under the notation of “Others”. Since the content of each irrelevant image cannot be covered, random categories were selected. The first four columns of Table 1 summarize the sample set with 2037 images (i: Images, o: Objects, A: Aeroplane, B: Bus, C: Car, M: Motorbike, O: Others). The image collection for test requires a set of images, which should contain relevant ones, according to the created search queries.

Table 1. Training and test sets

Category	Train		Validation		Test
	i	o	i	o	i
A	112	151	126	155	287
B	97	115	89	114	424
C	376	625	337	625	518
M	120	167	125	172	309
O	391	559	372	560	335

To prepare this image collection, the selected web image search engines were used, i.e., Google, Bing and Flickr, and the images in the result list are gathered. With this solution, our semantic image search system is searching among nearly the same images as the web image search engines.

For each search queries, the first 50 elements of the returned result list were downloaded at each search engines. Therefore, the image collection contains total of 1050 images. During the search, the components in the search queries were separated by “&” character. In order to evaluate the results, the label of the pictures should be known. Each image was examined individually, and labelled manually by following a few basic principles:

- An image is assigned to all of the categories that it appears in.
- An object (and its category) appears in a picture, if at least 20% of it is recognizable, and if its exterior can be seen.
- Hybrid vehicles are not considered adequate objects (for example, a car with wings is neither a car nor an aeroplane).
- An image is assigned to the Others category, if none of the selected vehicles appear in it.

The result of these annotations can be seen in the last column of Table 1. The annotations help us to decide whether an image is relevant or not, according to a given search query. An image is considered to be relevant, if all listed objects are displayed separately on it. It is important to note that it is difficult to generalize what constitutes relevant result, especially in the case of multi-object search (for example it is possible that a user wants to search for hybrid vehicles).

Each search key was tested with all implemented image search methods. In order to compare our results, the returned result lists were evaluated to the 50th element. In addition, the downloaded lists with also 50 elements were evaluated. AP (Average Precision) and its average: MAP (Mean Average Precision) indicators were used for evaluation, which are derived from precision and recall based on the confusion matrix:

$$\text{Average Precision} = \sum_{i=1}^{M} \text{precision}(i) \cdot \Delta\text{recall}(i). \tag{11}$$

5. Results

In this chapter, the results of the described experiment were summarized. In the Table 2 the AP (Average Precision) values can be seen, depending on the search queries and methods. These values were measured at the 50th element. As it is shown, from the proposed methods the MIN method resulted the highest AP value, except in the case of {bus, car} search query, where the Product Methods resulted the best AP. In three cases, the MIN exceeds the results of the web search engines, and in three other cases, the Flickr was the best. It is important to note that in the case of {bus, motorbike} search query, Flickr resulted the same image in the first seven places.

Table 2. Average Precision values at first 50 hits for each method

1	A	A	B	B	C	A	B
2	B	C	C	M	M	B	C
3	–	–	–	–	–	C	M
Google	0.0799	0.0211	0.0338	0.0465	0.0150	0.1402	0.0151
Bing	0.0922	0.0762	0.0176	0.0804	0.0388	0.0602	0.0154
Flickr	0.0405	0.0182	0.0440	0.1930	0.0892	0.0385	0.0278
S1	0	0.0645	0.0150	0.0015	0.0221	0.0437	0.0028
S2	0	0.0645	0.0150	0.0015	0.0221	0.0437	0.0028
P1	0.0332	0.0750	0.0183	0.0206	0.0337	0.1429	0.0161
P2	0.0332	0.0750	0.0183	0.0206	0.0337	0.0162	0.0009
MIN	0.0829	0.1153	0.0147	0.0639	0.0488	0.1582	0.0580
ML	0.0328	0.0733	0.0128	0.0006	0.0451	0.1345	0
DIF	0	0.0073	0.0104	0.0006	0.0020	0.0005	0

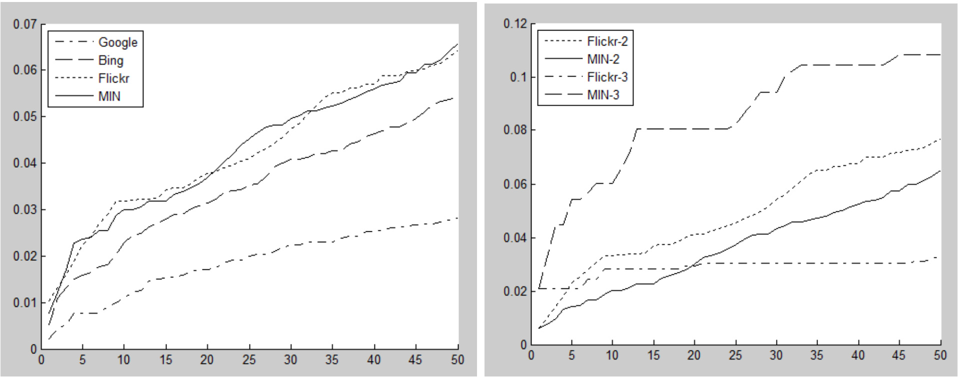
In addition to the above metrics, it is important to know how many relevant results exactly a method gives a specific search query. These are summarized in Table 3, where the rows correspond to the search queries. The first column contains the maximum number of relevant images, then the number of returned relevant images by Google, Bing, Flickr and MIN (only this was selected, because of the most effective one), respectively. MIN method gives the most relevant pictures (more than 14) on average in the first 50 places as can be seen in the last row.

Table 3. Number of returned relevant images for each search term

Search query	Total	Google	Bing	Flickr	MIN
A, B	64	10	**16**	11	14
A, C	101	10	20	5	**23**
B, C	159	16	11	**20**	11
B, M	82	12	12	**23**	15
C, M	126	8	14	**20**	17
A, B, C	42	10	5	3	**11**
B, C, M	56	5	7	6	**10**
Average		10.1	12.1	12.6	**14.4**

The MAP value defined in (2) can be calculated as the average of the AP values. This will give the goodness of the methods (including the web search engines as well) in the case of these search queries. Therefore, the MAP values of each method were calculated. Among them, MIN method provides the highest value, so on average, this method performs the best.

The diagram in Fig. 1a shows that the system we have created provides more relevant images than Google, Bing or Flickr search engines for these search queries, under the conditions of this experiment. The MAP values are shown in the vertical axis, the horizontal axis represents the number of elements in the beginning of the result list. On the graph Google, Bing, Flickr and MIN methods were marked with dash-dot, dashed, dotted and solid lines, respectively; they have reached 0.0281, 0.0544, 0.0657, 0.0644 value in MAP respectively.



a.

b.

Fig. 1. MAP values of MF, Google, Bing and Flickr methods

Since the curves of Flickr and MIN methods are often crossed and the difference of their MAP values is small, therefore these methods were compared in another diagram in Fig. 1b. The results of the queries with two and three components were separated, and MAP values were calculated for each type. The numbers in the legend refer to the components number. The MAP value of Flickr is 0.0770 in the case of the queries with two components, and 0.0332 in the case of three components, while MIN resulted 0.0651 and 0.1081 respectively.

Based on the results, we can conclude that if a search query consists of more objects, our content-based search system will achieve better results with large likelihood than the web search engines. Considering the search queries of three objects, the maximum number of relevant images is small. Accordingly, Google, Bing and Flickr returned much less relevant images in these cases, while MIN method returned larger number of relevant images.

6. Conclusions

The developed image search system is specialized in search queries that are obtained by combining multiple objects. Since only the visual content information of the images was available for search in our system, it was necessary to implement a classification algorithm that helps to determine semantic information for the images of the search space. The semantic information is stored in a database, called offline database, which should be refreshed whenever the search space includes a new image. Several methods were developed and implemented that resolves the combined search, by using the available offline semantic information stored in the database. The difference between them is how to calculate the relevance value of an image from confidence values of object classifiers, and we have selected the best one for our system. Our solution is compared with the famous web images search engines (Google, Bing and Flickr), where web search engines can use metadata and the surroundings of the images as well, however our system should rank only smaller number of images. There was a comparison of their MAP values, and based on the results it can be concluded that our system reaches the benchmark, moreover in most cases our method outperforms the others, especially in the cases of queries with many objects.

We have a plan to continue this research; further work focuses on more visual objects and the acceleration. One of the acceleration possibilities of our solution is a specialized multicore architecture that can be used in slow calculations in the field of computer vision.

References

1. Abdi, H., L. J. Williams. Principal Component Analysis. – Wiley Interdisciplinary Reviews: Computational Statistics, Vol. **2**, 2010, No 4, pp. 433-459.
2. Arandjelovic, R., A. Zisserman. Three Things Everyone Should Know to Improve Object Retrieval. – In: Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012, pp. 2911-2918.

3. Banda, J. M., R. A. Angryk, P. C. Martens. Image FARMER: Introducing a Data Mining Framework for the Creation of Large-Scale Content-Based Image Retrieval Systems. – International Journal of Computer Applications, Vol. **79**, 2013, No 13, pp. 8-13.
4. Bao, Y., T. Wang, G. Qiu. Research on Applicability of SVM Kernel Functions Used in Binary Classification. – In: Proc. of International Conference on Computer Science and Information Technology, Springer, India, 2014, pp. 833-844.
5. Chatfield, K., V. Lempitsky, A. Vedaldi, A. Zisserman. The Devil Is in the Details: An Evaluation of Recent Feature Encoding Methods. – In: Proc. of British Machine Vision Conference, BMVA Press, September 2011, pp. 76.1-76.12.
6. Cortes, C., V. Vapnik. Support-Vector Networks. – Machine Learning, Vol. **20**, 1995, No 3, pp. 273-297.
7. Daróczy, B. Z., D. Siklósi, A. Benczúr. SZTAKI @ ImageCLEF 2012 Photo Annotation. – In: Working Notes of the ImageCLEF 2011 Workshop at CLEF 2012 Conference, Rome, Italy, 17-20 September 2012, pp. 1-6.
8. Everingham, M., L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. – International Journal of Computer Vision, Vol. **88**, 2010, No 2, pp. 303-338.
9. Fei-Fei, L., R. Fergus, A. Torralba. Recognizing and Learning Object Categories. – Computer Vision and Pattern Recognition (CVPR), 2007.
10. Fernando, B., T. Tuytelaars. Mining Multiple Queries for Image Retrieval: On-the-Fly Learning of an Object-Specific Mid-Level Representation. – In: Proc. of IEEE International Conference on Computer Vision (ICCV'2013), 3-6 December 2013, pp. 2544-2551.
11. Gosselin, P. H., D. Picard. Machine Learning and Content-Based Multimedia Retrieval. – In: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, April 2013, pp. 251-260.
12. Hoque, E., O. Hoeber, G. Strong, M. Gong. Combining Conceptual Query Expansion and Visual Search Results Exploration for Web Image Retrieval. – Journal of Ambient Intelligence and Humanized Computing, 2013, pp. 1-12.
13. Kaur, H., K. Jyoti. Survey of Techniques of High Level Semantic Based Image Retrieval. – International Journal of Research in Computer and Communication Technology (IJRCCT), Vol. **2**, 2013, No 1, pp. 15-19.
14. Ke, Y., R. Sukthankar. PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. – In: Proc. of 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'2004., Vol. **2**, 2004, pp. II-506-II-513.
15. Lazebnik, S., C. Schmid, J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. – In: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, New York, Vol. **2**, 2006, pp. 2169-2178.
16. Liu, D., S. Wang, Y. Liu, F. Zeng, J. Wu, W. Li. Tree Representation and Feature Fusion Based Method for Multi-Object Binary Image Retrieval. – Journal of Information & Computational Science, Vol. **10**, 2013, No 4, pp. 1055-1064.
17. Lowe, D. G. Object Recognition from Local Scale-Invariant Features. – In: International Conference on Computer Vision, Corfu, Greece, 1999, pp. 1150-1157.
18. Lowe, D. G. Distinctive Image Features from Scale-Invariant Keypoints. – International Journal of Computer Vision, Vol. **60**, 2004, No 2, pp. 91-110.
19. Murthy, V. N., E. F. Can, R. Manmatha. A Hybrid Model for Automatic Image Annotation. – In: Proc. of International Conference on Multimedia Retrieval, ACM, 2014, p. 369.
20. Perronnin, F., C. Dance. Fisher Kernels on Visual Vocabularies for Image Categorization. – In: IEEE Conference Computer Vision and Pattern Recognition (CVPR'07), 2007, pp. 1-8.
21. Ramamurthy, B., K. R. Chandran. CBMIR: Content Based Medical Image Retrieval Using Multilevel Hybrid Approach. – International Journal of Computers Communications & Control, Vol. **10**, 2015, No 3, pp. 382-389.
22. Reynolds, D. A. Gaussian Mixture Models. Encyclopedia of Biometric Recognition. Springer, February 2008.

23. Riad, M., K. Elminir, S. Abd-Elghany. A Literature Review of Image Retrieval Based on Semantic Concept. – International Journal of Computer Applications, Vol. **40**, 2012, No 11, pp. 12-19.

24. Sánchez, J., F. Perronnin, T. Mensink. Improved Fisher Vector for Large Scale Image Classification. – In: Proc. of 11th ECCV: Part IV, 5-11 September 2010, pp. 143-156.

25. Seera, M., C. P. Lim. A Hybrid Intelligent System for Medical Data Classification. – Expert Systems with Applications, Vol. **41**, 2014, No 5, pp. 2239-2249.

26. Szűcs, G., D. Papp, D. Lovas. Viewpoints Combined Classification Method in Image-Based Plant Identification Task. – In: L. Cappellato, N. Ferro, M. Halvey, W. Kraaij, Eds. Working Notes for CLEF 2014 Conference, Sheffield, UK, September 15-18, 2014, pp. 763-770.

27. Tomasi, C. Estimating Gaussian Mixture Densities with EM A Tutorial. (Tech. Rep., Duke University). – Chinese Journal of Electron Devices, 2004, pp. 15-18.

28. Tronci, R., G. Murgia, M. Pili, L. Piras, G. Giacinto. Imagehunter: A Novel Tool for Relevance Feedback in Content Based Image Retrieval. – In: New Challenges in Distributed Information Filtering and Retrieval. Berlin, Heidelberg, Springer, 2013, pp. 53-70.

29. Wan, G. G., Z. Liu. Content-Based Information Retrieval and Digital Libraries. – Information Technology and Libraries, Vol. **27**, 2013, No 1, pp. 41-47.

30. Woźniak, M., M. Graña, E. Corchado. A Survey of Multiple Classifier Systems as Hybrid Systems. – Information Fusion, Vol. **16**, 2014, pp. 3-17.

31. Yang, Y., F. Nie, D. Xu, J. Luo, Y. Zhuang, Y. Pan. A Multimedia Retrieval Framework Based on Semi-Supervised Ranking and Relevance Feedback. – IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. **34**, 2012, No 4, pp. 723-742.

32. Zhang, H., Z. J. Zha, Y. Yang, S. Yan, Y. Gao, T. S. Chua. Attribute-Augmented Semantic Hierarchy: Towards Bridging Semantic Gap and Intention Gap in Image Retrieval. – In: Proc. of 21st ACM International Conference on Multimedia, 2013, ACM, pp. 33-42.

33. Zhang, M., K. Zhang, Q. Feng, J. Wang, J. Kong, Y. Lu. A Novel Image Retrieval Method Based on Hybrid Information Descriptors. – Journal of Visual Communication and Image Representation, Vol. **25**, 2014, No 7, pp. 1574-1587.