International Journal of Computer Science and Applications

Vol. 9, No. 3, pp. 1 – 19, 2012

AUTOMATED ANNOTATION OF NATURAL IMAGES USING AN EXTENDED ANNOTATION MODEL

GABRIEL MIHAI

Faculty of Automation, Computers and Electronics, University of Craiova, Bvd. Decebal, No.107, Craiova, Dolj, Romania
mihai_gabriel@software.ucv.ro

LIANA STANESCU

Faculty of Automation, Computers and Electronics, University of Craiova, Bvd. Decebal, No.107, Craiova, Dolj, Romania
stanescu@software.ucv.ro

Automated annotation of digital images remains a highly challenging task being used for indexing, retrieving, and understanding of large collections of image data. This paper presents an original extension of an image annotation model using an object oriented approach. The proposed model is an extension of an efficient annotation model called Cross Media Relevance Model. Image's regions are described using a vocabulary of blobs generated from image features using the K-means clustering algorithm. Using SAIAPR TC-12 Dataset of annotated images it is estimated the joint probability of generating a concept given the blobs in an image. The annotation process of each new image starts with a segmentation phase that is using a our original segmentation algorithm based on a hexagonal structure. The information required for the annotation process is stored in an open source object database called db4o. An object oriented database offers suport for storing complex objects as sets, lists, trees or other advanced data structures.

Keywords: Image annotation, image segmentation, ontology, relevance models.

1. Introduction

Automatic image annotation (AIA) has been studied extensively for a several years. AIA is defined as the process by which a computer system automatically assigns metadata in the form of text description or keywords to a digital image. This process is used in image retrieval systems to organize and locate images of interest from a database. This task can be regarded as a type of multi-class image classification with a number of classes equal with vocabulary's size. AIA can be seen also as a multi-class object recognition problem which is a challenging task and an open problem in computer vision.

The importance of this task has increased with the growth of the digital images collections. An important amount of digital pictures is generated each year and thus there is a need for an efficient image management system that is capable to fast searching, browsing by topic (e.g. using Google Picasa [http://picasa.google.com/]) or tagging images (e.g using Flikr [http://www.flickr.com/]).

Content-based Image Retrieval (CBIR) has been studied for several years. The accuracy of CBIR systems is still not sufficient for current needs. Searching images by

content remains a difficult and very challenging task. A text retrieval system can be used for finding rapidly related documents from a vast amount of documents containing keywords. Search engines like Google offers the possibility to search for images using surrounding text and file name. This image search is based on text retrieval because the content of the image is ignored. For this reason sometimes the search performed does not lead to satisfactory results.

In order to avoid this drawback the researchers are looking for another way to search for images. A possible approach is to obtain a textual description from the image and then use text retrieval for searching. A different approach is to combine two modalities for example text and visual features when indexing images. Image retrieval based on text is sometimes called Annotation Based Image Retrieval (ABIR) [Inoue (2004)].

The systems based on ABIR can have some draw-backs. Researchers working in CBIR have identified two limitations. The first limitation is that ABIR requires manual image annotation which is time consuming and costly. The second limitation is that human annotation is subjective and sometimes it is difficult to describe image contents by concepts. An AIA system can solve the first limitation. The second limitation remains a general question and a unsolved problem for computer vision.

AIA is situated on the frontier of different fields: image analysis, machine learning, media understanding and information retrieval. Usually image analysis is based on feature vectors and the training of annotation concepts is based on machine learning techniques. Automatic annotation of new images is possible only after the learning phase is completed. General object recognition and scene understanding techinques are used to extract the semantics from data. This is an extremely hard task because AIA systems have to detect at least a few hundred objects at the same time from a large image database.

AIA is a challenge that has been identified as one of the hot-topics in the new age of image retrieval [Datta R., *et al*. (2008)]. Image annotation is a difficult task for two main reasons:

- *semantic gap* problem – it is hard to extract semantically meaningful entities using just low level image features. Low-level features can be easily extracted from images but they are not completely descriptive for image content. High-level semantic information is meaningful and effective for image retrieval.
- *the lack of correspondence* between the keywords and image regions in the training data.

The semantic gap is due to at least two main problems:

- *semantic extraction problem* - how to extract the semantic regions from image data? Current object recognition techniques do not cover completely this problem.
- *semantic interpretation problem* – is represented by complexity, ambiguity and subjectivity in user interpretation.

Representing the content of the image using image features and then performing non-textual queries like color and texture is not an easy task for users. They prefer instead textual queries and this request can be satisfied using automatic annotation.

There are many annotation models proposed and each model has tried to improve a previous one. These models are splitted in two categories:

(1) Parametric models: Co-occurrence Model [Mori Y., *et al.* (1999)], Translation Model [Duygulu P., *et al.* (2002)], Correlation Latent Dirichlet Allocation [Blei and Jordan (2003)]

(2) Non-parametric models: Cross Media Relevance Model (CMRM) [Jeon J., *et al.* (2003)], Continuous Cross-Media Relevance Model (CRM) [Lavrenko V., *et al.* (2004)], Multiple Bernoulli Relevance Model (MBRM) [Deerwester S.C, *et al.* (1990)], Coherent Language Model (CLM) [Rong J., *et al*. (2004)].

The annotation process implemented in our system is based on CMRM. Using a set of annotated images [Segmented and Annotated IAPR TC-12 dataset] the system learns the joint distribution of the blobs and concepts. The blobs are clusters of image regions obtained using the K-means algorithm. Having the set of blobs each image from the test set is represented using a discrete sequence of blobs identifiers. The distribution is used to generate a set of concepts for a new image.

Each new image is segmented using a our original segmentation algorithm [Burdescu D., *et al.* (2009)] which integrates pixels into a grid-graph. The usage of the hexagonal structure improves the time complexity of the used methods and the quality of the segmentation results.

The meaningful keywords assigned by the annotation system to each new image are retrieved from an ontology created in an original manner starting from the information provided by [Segmented and Annotated IAPR TC-12 dataset]. The concepts and the relationships between them in the ontology are inferred from the concepts list, from the ontology's paths and from the existing relationships between regions.

The remainder of the paper is organized as follows: related work is discussed in Section 2, Section 3 provides details about the segmentation algorithm used, Section 4 contains a description of the annotation process, Section 5 presents details about the evaluation of the annotation task and Section 6 concludes the paper.

2. Related Work

Object recognition and image annotation are very challenging tasks. For this reason a number of models using a discrete image vocabulary have been proposed for the image annotation task. One approach to automatically annotating images is to look at the probability of associating concepts with image regions. [Mori Y., *et al.* (1999)] used a Co-occurrence Model in which they looked at the co-occurrence of concepts with image regions created using a regular grid. To estimate the correct probability this model required large numbers of training samples. Each image is converted into a set of rectangular image regions by a regular grid. The keywords of each training image are propagated to each image region. The major drawback of the above Co-occurrence Model is that it assumes that if some keywords are annotated to an image, they are propagated to each region in this image with equal probabilities.

[Duygulu P., *et al.* (2002)] described images using a vocabulary of blobs. Image regions were obtained using the Normalized-cuts segmentation algorithm. For each image region 33 features such as color, texture, position and shape information were computed. The regions were clustered using the K-means clustering algorithm into 500 clusters called "blobs". The vector quantized image regions are treated as "visual words" and the relationship between these and the textual keywords can be thought as that between two languages, such as French and German. The training set is analogous to a set of aligned bitexts – texts in two languages. Given a test image, the annotation process is similar to translating the visual words to textual keywords using a lexicon learned from the aligned bitexts. This annotation model called Translation Model was a substantial improvement of the Co-occurrence model.

[Jeon J., *et al*. (2003)] viewed the annotation process as analogous to the cross-lingual retrieval problem and used a Cross Media Relevance Model to perform both image annotation and ranked retrieval. The experimental results have shown that the performance of this model on the same dataset was considerably better than the models proposed by [Mori Y., *et al.* (1999)] and [Duygulu P., *et al.* (2002)]. The essential idea is that of finding the training images which are similar to the test image and propagate their annotations to the test image. CMRM does not assume any form of joint probability distribution on the visual features and textual features so that it does not have a training stage to estimate model parameters. For this reason, CMRM is much more efficient in implementation than the above mentioned parametric models.

There are other models like Correlation LDA proposed by [Blei and Jordan (2003)] that extends the Latent Dirichlet Allocation model to words and images. This model is estimated using Expectation-Maximization algorithm and assumes that a Dirichlet distribution can be used to generate a mixture of latent factors.

In [Jeon and Manmatha (2004)] is proposed the use of the Maximum Entropy approach for the task of automatic image annotation. Maximum Entropy is a statistical technique allowing predicting the probability of a label given test data. The image is represented using a language of visterms (visual terms) which are clusters of rectangular regions.

In [Li and Wang (2003)], [Li and Wang (2008)] is described a real-time ALIPR image search engine which uses multi resolution 2D Hidden Markov Models to model concepts determined by a training set. A computational efficiency is obtained in [Li and Wang (2008)] due to a fundamental change in the modeling approach. In [Li and Wang (2003)] every image was characterized by a set of feature vectors residing on grids at several resolutions. The profiling model of each concept is the probability law governing the generation of feature vectors on 2-D grids. Under the new approach, every image is characterized by a statistical distribution. The profiling model specifies a probability law for distributions directly.

In [Monay and Perez (2004)] Latent Semantic Analysis (LSA) [Deerwester S.C, *et al.* (1990)], and Probabilistic Latent Semantic Analysis (PLSA) [Hofmann (2001)] are explored for automatic image annotation. A document of image and texts can be represented as a bag of words, which includes the visual words – vector quantized image

regions and textual words. Then LSA and PLSA can be deployed to project a document into a latent semantic space. Annotating images is achieved by keywords propagation in this latent semantic space.

An improved model of CMRM is proposed in [Lavrenko V., *et al.* (2004)], the Continuous Cross-Media Relevance Model (CRM) which preserves the continuous feature vector of each region and this offers more discriminative power. A further extension of the CRM model called the Multiple Bernoulli Relevance Model (MBRM) is presented in [Feng S. L., *et al.* (2004)]. The keyword distribution of an image annotation is modeled as a multiple Bernoulli distribution, which only represents the existence/nonexistence binary status of each word.

All the above mentioned methods predict each word independently given a test image. They can model the correlation between keywords and visual features but they are not able to model the correlation between two textual words. To solve this problem, in [Rong J., *et al.* (2004)] it is proposed a Coherent Language Model (CLM) extended from CMRM. This model defines a language model as a multinomial distribution of words. Instead of estimating the conditional distribution of a single word it is estimated the conditional distribution of the language model. The correlation between words is explained by a constraint on the multinomial distribution that the summation of the individual words distribution is equal to one. The prediction of one word has an effect on the prediction of another word.

3. The segmentation algorithm

For image segmentation we have used an our original and efficient segmentation algorithm [Burdescu D., *et al.* (2009)] based on color and geometric features of an image. The efficiency of this algorithm concerns two main aspects:

a) minimizing the running time – a hexagonal structure based on the image pixels is constructed and used in color and syntactic based segmentation

b) using a method for segmentation of color images based on spanning trees and both color and syntactic features of regions. A similar approach is used in [Felzenszwalb and Huttenlocher (2004)] where image segmentation is produced by creating a forest of minimum spanning trees of the connected components of the associated weighted graph of the image.

In Fig.1. is presented the hexagonal structure used by the segmentation algorithm:

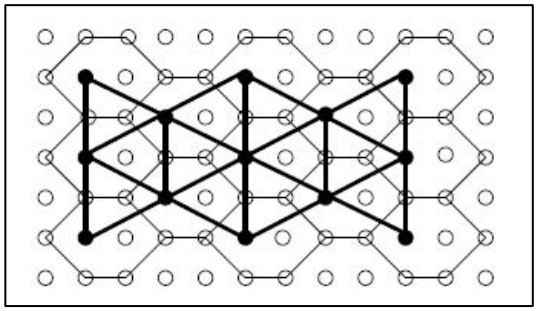


Fig. 1. The grid-graph constructed on the hexagonal structure of an image

A particularity of this approach is the basic usage of the hexagonal structure instead of color pixels. In this way the hexagonal structure can be represented as a grid-graph G = (V, E) where each hexagon h in the structure has a corresponding vertex $v \in V$, as presented in Fig.1. Each hexagon has six neighbors and each neighborhood connection is represented by an edge in the set E of the graph. To each hexagon two important attributes are associated: the dominant color and the coordinates of the gravity center. For determining these attributes were used eight pixels: the six pixels of the hexagon frontier, and two interior pixels of the hexagon.

Image segmentation is realized in two distinct steps:

(1) a pre-segmentation step – only color information is used to determine an initial segmentation. A color based region model is used to obtain a forest of maximum spanning trees based on a modified form of the Kruskal's algorithm. For each region of the input image it is obtained a maximal spanning tree. The evidence for a boundary between two adjacent regions is based on the difference between the internal contrast and the external contrast between the regions

(2) a syntactic-based segmentation – color and geometric properties of regions are used. It is used a new graph which has a vertex for each connected component determined by the color-based segmentation algorithm. The region model contains in addition some geometric properties of regions such as the area of the region and the region boundary. A forest of minimum spanning trees is obtained using a modified form of the Boruvka's algorithm. Each minimum spanning tree represents a region determined by the segmentation algorithm.

4. The annotation process

Details about the annotation process are presented below.

4.1. *The dataset*

We have used for our experiments the segmented and annotated SAIAPR TC-12 [Segmented and Annotated IAPR TC-12 dataset], [Escalante H.J., *et al.* (2010)] benchmark which is an extension of the IAPR TC-12 [IAPR TC-12 Benchmark] collection for the evaluation of automatic image annotation methods and for studying

their impact on multimedia information retrieval. IAPR TC-12 was used to evaluate content-based image retrieval and multimedia image retrieval methods [Clough P., *et al.* (2006)], [Grubinger M., *et al.* (2007)]. SAIAPR TC-12 benchmark contains the pictures from the IAPR TC-12 collection plus: segmentation masks and segmented images for the 20,000 pictures, region-level annotations according an annotation hierarchy, region-level annotations according an annotation hierarchy, spatial relationships information. Each image was manually segmented using a Matlab tool named Interactive Segmentation and Annotation Tool (ISATOOL), that allows the interactive segmentation of objects by drawing points around the desired object, while splines are used to join the marked points, which also produces fairly accurate segmentation with much lower segmentation effort. Each region has associated a segmentation mask and a label from a predefined vocabulary of 275 labels. This vocabulary is organized according to a hierarchy of concepts having six main branches: *Humans*, *Animals*, *Food*, *Landscape-Nature*, *Man-made* and *Other*.

For each pair of regions the following relationships have been calculated in every image: adjacent, disjoint, beside, X-aligned, above, below and Y-aligned. The following features have been extracted from each region: area, boundary/area, width and height of the region, average and standard deviation in x and y, convexity, average, standard deviation and skewness in two color spaces: RGB and CIE-Lab.

4.2. *The database*

db4o [db4objects] is an open-source object-oriented database having bindings to both the .NET and Java platforms and allowing the data objects to be stored exactly in the way they are defined by the application. Unlike string-based query languages db4o offers truly native and object-oriented data access APIs like language integrated queries for querying the database, query by example, retrieval by object graph. The elimination of data transformation in db4o leads to less demand on CPU or persistence operations, which shifts critical resources to the application logic and query processing [Db4o Developer Community].

For db4o there are available the following methods for querying objects:

(1) *Query by Example (QBE)* - a query expression is based on template objects beeing fast and simple to implement. This method is an optimal solution for simple queries that are not using logical operators.

(2) *Simple Object Data Access (SODA)* – a query expression is based on query graphs. This method builds a query graph by navigating references in classes and imposing constraints. SODA has several disadvantages [Paterson J., et al. (2006)] because a query is expressed as a set of method calls that explicitly define the graph and it is not similar in any way to traditional querying techniques.

(3) *Native Queries (NQ)* – this querying approach express the query in a .NET or Java – compliant language by writting a method that returns a boolean value. The method is applied to all objects stored and the list of matching object instances is returned.

(4) LINQ (Language Integrated Query) – is the recommended db4o querying interface for .NET platforms. LINQ allows you to write compile checked db4o queries, which can be refactored automatically when a field name changes and which are supported by code auto-completion tools.

db4o offers support for client/server interactions, each interaction beeing one of the following three types:

(1) Networking – is the traditional way of operating in most database solutions. Remote clients open a TCP/IP connection to send/retrieve data to/from the db4o server.

(2) Embedded – the client and the server are run on the same machine. The communication between the server and the client is the same as in networking mode but the work is entirely made within one process.

(3) Out-of-band signalling – the information sent does not belong to the db4o protocol and does not consist of data objects, but instead is completely user-defined. This mode uses a message passing communication interface.

4.3. *The annotation model based on an object oriented approach*

The Cross Media Relevance Model is a non-parametric model for image annotation that assigns words to the entire image and not to specific blobs – clusters of image regions, because the blob vocabulary can give rise to many errors. Some principles defined for the relevance models [Lavrenko V., et al. (2001)], [Lavrenko V., et al. (2002)] are applied by this model to automatically annotate images and for ranked retrieval. Relevance models were introduced to perform a query expansion in a more formal manner. Given a training set of images with annotations this model allows predicting the probability of generating a word given the blobs in an image. A test image *I* is annotated by estimating the joint probability of a keyword *w* and a set of blobs

$$P(w,b_1,...,b_m) = \Sigma_{J \in T} P(J)P(w,b_1,....,b_m \mid J) \qquad (1)$$

For the annotation process the following assumptions are made:

a) it is given a collection C of un-annotated images
b) each image I from C to can be represented by a discrete set of blobs: $I = \{b_1 ... b_m\}$
c) there exists a training collection T, of annotated images, where each image J from T has a dual representation in terms of both words and blobs: $J = \{b_1 ....b_m ; w_1 ...w_n\}$
d) $P(J)$ is kept uniform over all images in T
e) the number of blobs m and words in each image (m and n) may be different from image to image.
f) no underlying one to one correspondence is assumed between the set of blobs and the set of words; it is assumed that the set of blobs is related to the set of words.

$P(w,b_1,...,b_m \mid J)$ represents the joint probability of keyword *w* and the set of blobs $\{b_1 ... b_m\}$ conditioned on training image *J*. An intuitive interpretation of this probability is how likely *w* co-occurs with individual blobs given that we have observed an annotated image *J*.

In CMRM it is assumed that, given image *J*, the events of observing a particular keyword *w* and any of the blobs $\{b_1 \ldots b_m\}$ are mutually independent, so that the joint probability can be factorized into individual conditional probabilities. This means that) $P(w, b_1, \ldots, b_m \mid J)$ can be written as:

$$P(w, b_1, \ldots, b_m \mid J) = P(w \mid J) \prod_{i=1}^{m} P(b_i \mid J) \qquad (2)$$

$$P(w \mid J) = (1 - \alpha_J) \frac{\#(w, J)}{|J|} + \alpha_J \frac{\#(w, T)}{|T|} \qquad (3)$$

$$P(b \mid J) = (1 - \beta_J) \frac{\#(b, J)}{|J|} + \beta_J \frac{\#(b, T)}{|T|} \qquad (4)$$

where:

(1) P(w|J) , P(w|J) denote the probabilities of selecting the word w, the blob b from the model of the image J.
(2) #(w, J) denotes the actual number of times the word w occurs in the caption of image J.
(3) #(w, T) is the total number of times w occurs in all captions in the training set T .
(4) #(b, J) reflects the actual number of times some region of the image J is labeled with blob b.
(5) #(b, T) is the cumulative number of occurrences of blob b in the training set.
(6) |J| stands for the count of all words and blobs occurring in image J.
(7) |T| denotes the total size of the training set.
(8) The prior probabilities *P(J)* can be kept uniform over all images in *T*

The smoothing parameters α and β determine the degree of interpolation between the maximum likelihood estimates and the background probabilities for the words and the blobs respectively. The values determined after experiments for the Cross Media Relevance Model were $\alpha = 0.1$ and $\beta = 0.9$.

Starting from the principles of the CMRM model we have obtained an object oriented model using the classes presented in table 1 and the mapping presented in table 2.

Table 1. The classes used by the object oriented model

Classes	**Members**	**Member's Type**
Image	*PictureName*	*String*
	Regions	*List<Region>*
Region	*Index*	*int*
	AssignedBlob	*Blob*
	AssignedConcepts	*Concept*
	FeaturesVectorItem	*FeaturesVector*
	MatrixFilePath	*String*
Blob	*Index*	*int*
	AverageFeaturesVector	*FeaturesVector*
FeaturesVector	*Features*	*List<double>*
Concept	*Name*	*String*
	OriginalIndex	*int*
RegionsRelationship	*RegionA*	*Region*
	RegionB	*Region*
	RelationshipMode	*String*
HierarchicalRelationship	*ParentConcept*	*Concept*
	ChildConcept	*Concept*

Table 2. The mapping used between the CMRM model and the object oriented model

CMRM model	**Object oriented model**
$P(w \mid J)$	*public double PWJ(Concept w, Image J, IobjectContainer db, int cardT)*
$P(b \mid J)$	*public double PBJ(Blob b, Image J, IobjectContainer db, int cardT)*
$P(w, b_{1,..}b_m \mid J)$	*public double PWBsJ(Concept w, List<Blob> blobs, Image J, IobjectContainer db, int cardT)*
$P(w, b_{1,..}b_m)$	*public double PWBs(Concept w, List<Blob> blobs, List<Image> T, IobjectContainer db, int cardT)*

For that object oriented model we have made some changes in order to improve the results of the annotation process obtained using the initial version. In [Jeon J., *et al*. (2003)] it was mentioned that for the CMRM model the experimental results have shown a mean precision value equal with 0.33 and a mean recall value equal with 0.37. We considered that these values could be further improved. In order to achieve this target some changes were involved having as a result a modified model. The experimental results will show better values for mean precision and mean recall.

The modified version concerns the following two aspects of the annotation task that will be taken into account when computing the probabilities:

- *only the concepts* that were associated with the clusters identified based on the regions of the new image will be considered
- *only the images* having regions associated with the clusters identified based on the regions of the new image will be considered

Using only the concepts and the images associated with the identified clusters, more accurate values are obtained for the computed probabilities. In the initial version all concepts and images were taken into account. The main drawback of this version was represented by the fact that it was possible to have several concepts that were not relevant at all (or assigned to other clusters than the ones identified) for a given image, but their frequency in the training set was high, so a major contribution to the probability value. Because the probability is calculated as a sum of the contribution of each concept, high probability values were not always accurate.

The two versions are included in the annotation method presented below as example:

```
Public List<Concept> AnnotateImage (string imagePath, int n, bool modifiedVersion) {
-- Obtaining the list of Blob objects from the database
Ienumerable<Blob> allBlobs = from Blob b in db select b;
-- Obtaining the list of Image objects from the trainning set T existing in the database
Ienumerable<Image> T = from Image img in db select img;
-- Obtaining the list of Region objects by segmenting the image given by path
List<Region> regions = SegmentImage (imagePath);
```

```
-- Obtaining the list of distinct Blob objects associated with the regions detected above
List<Blob> blobs = DetectBlobs (regions);
--Obtaining the lists of Concept and Image objects from the database by taking into account  the version
Ienumerable<Concept> concepts;
Ienumerable<Image> I;
if (modifiedVersion)
{
   --this is the modified version
   -- selecting only the Concept objects belonging to the regions assigned to the clusters that were      identified
         concepts = from Region r in db where r.ContainedIn(blobs).Equals(1) select r.Concept;
     --selecting only the Image objects having regions assigned to the clusters that were     identified
        I = from Image img in db where img.ContainedIn (blobs).Equals (1) select img;
}
else
{
    -- this is the standard version
  -- selecting all Concept objects from the database
   concepts = from Concept w in db select w;
-- using all images from the trainning set T
   I = T;
}
--sorting the list of Concept objects ascending by name
    concepts = concepts .OrderBy (p=>p.Name);
-- equivalent with |T|
int cardT = concepts.Count() + allBlobs.Count();
-- the list containing Probability objects, one object will be computed  for each Concept object
List<Probability> probabilities = new List<Probability>();
double value;
foreach (Concept w in concepts){
 -- calculating the probability for a Concept object
   value = PWBs(w, blobs, T,db, cardT, I);
  -- creating a new Probability object to store the probability value computed above
    Probability p = new Probability ();
    p.Concept = w;
    p.Value = value;
   probabilities.Add (p);
}
-- the list of Probability object is sorted descending based on the Value field
probabilities = probabilities.OrderByDescending(pd => pd.Value).ToList();
--returning only the first n relevant Concept objects to be used for annotating the image
List<Concept> annotationConcepts = probabilities.GetFirstConcepts (n);
return annotationConcepts;
}
```

4.4. *Steps involved by the annotation process*

The annotation process contains several steps:

- *Obtaining the ontology* – the information provided by the dataset is processed by the *Importer module*. The concepts associated with images and their hierarchical structure is identified. The *Ontology creator module* is using that information to generate the ontology. The files containing feature values (extracted from image regions) are processed and stored in the database. The hierarchical structure of the ontology obtained is presented in Fig.2.

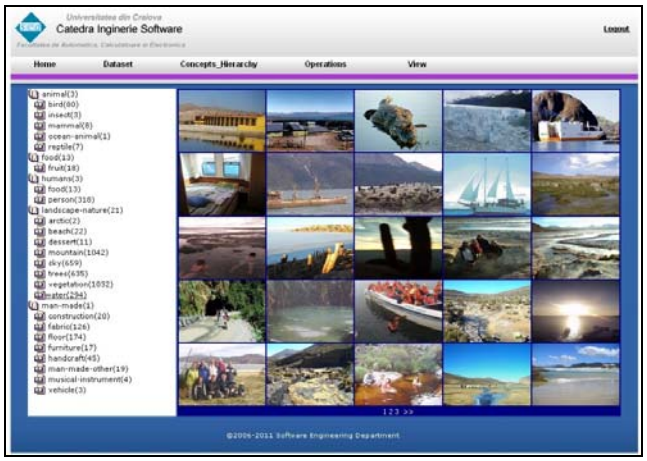


Fig. 2. Hierarchical structure of the ontology

- *Obtaining the clusters* – we have used K-means algorithm to quantize the feature vectors obtained from the training set and to generate blobs. After the quantization, each image in the training set was represented as a set of blobs identifiers. For each blob it is computed a median feature vector and a list of concepts that were assigned to the test images that have that blob in their representation. The clustering process is summarized in Fig.3.

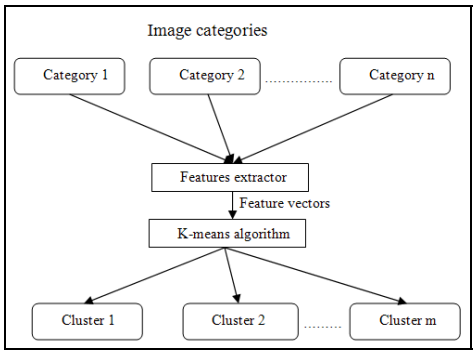


Fig.3. Clustering process

- *Image segmentation* – the Segmentation module is using the segmentation algorithm described in Section 3 to obtain a list of regions from each new image. In Fig.4. it is presented the list of regions obtained after segmentation together with the annotation result.

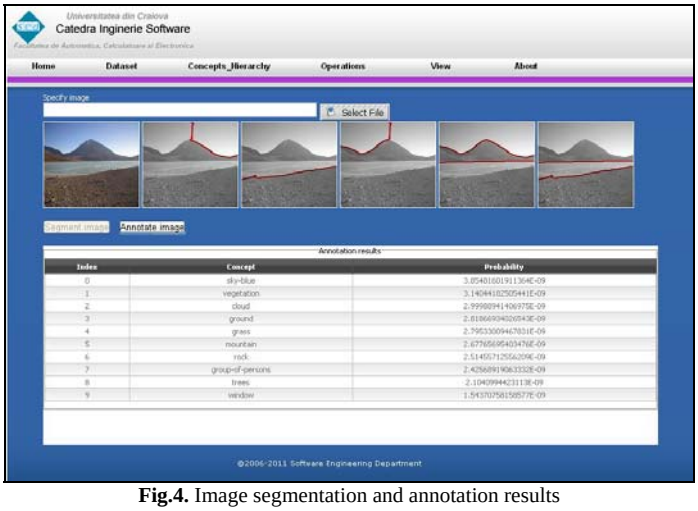


Fig.4. Image segmentation and annotation results

- *Automated image annotation* – this task is performed according with the steps involved by the *AnnotateImage* method presented above. An example is given in Fig.5.

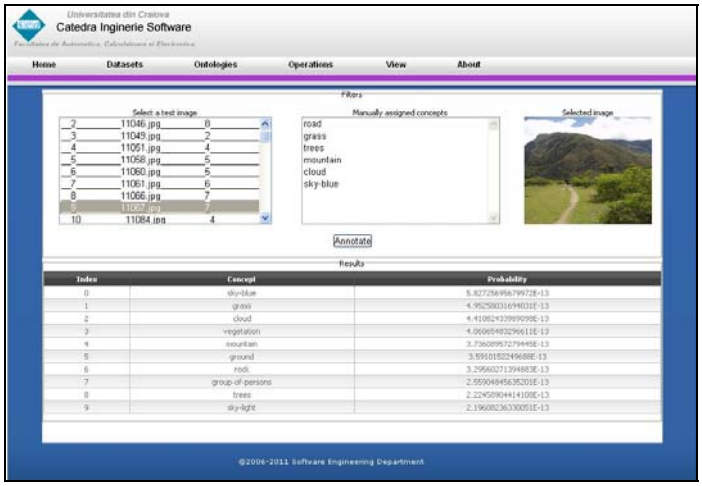


Fig.5. Image annotation

The entire annotation process is summarized in Fig.6.

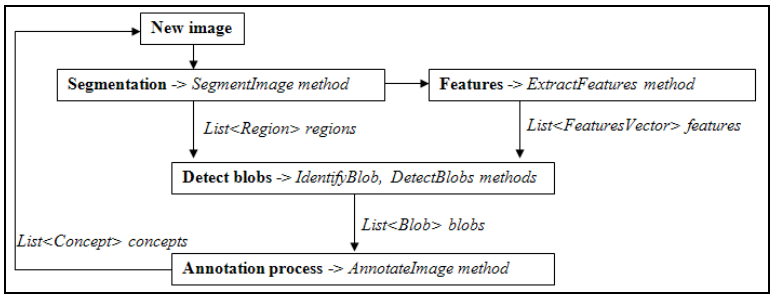


Fig.6. Image annotation process

All tasks involved by this process are implemented in a system having the architecture presented in Fig.7.

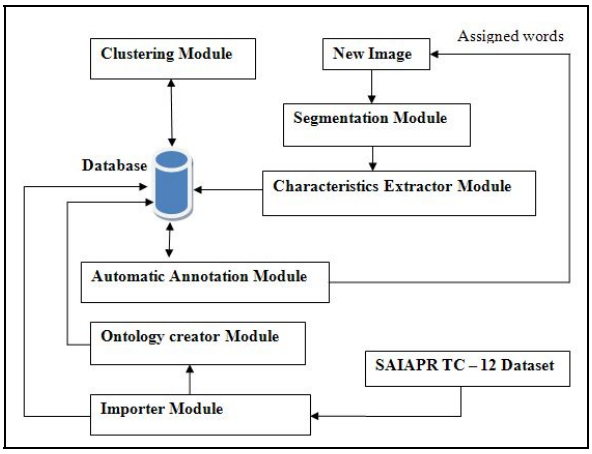


Fig.7. System's architecture

5. Evaluation of the Annotation Task

The annotation task can be evaluated based on several measures.

5.1 *Measures for the evaluation of the annotation task*

Evaluation measures [30] are considered to evaluate the annotation performance of an algorithm. Let T' represent a test set, $J \in T'$ be a test image, W_J be its manual annotation set and W_J^a be its automatic annotation set. The performance can be analyzed from two perspectives:

(1) *Annotation perspective*. Two standard measures that are used for analyzing the performance from the annotation perspective are:

- *Accuracy*. The accuracy of the auto-annotated test images is measured as the percentage of correctly annotated concepts and for a given test image $J \subset T'$ is defined as [30]:

$$accuracy = \frac{r}{|J|} \qquad (5)$$

where variable *r* represents the number of correctly predicted concepts in *J*. The disadvantage of this measure is represented by the fact that it does not take into account for the number of wrong predicted concepts with respect to the vocabulary size |W|.

- *Normalized score* (*NS*). It is extended directly from accuracy and penalizes the wrong predictions. This measure is defined as [30]:

$$NS = \frac{r}{|W_J|} - \frac{r'}{|W| - |W_J|} \qquad (6)$$

where variable *r'* denotes the number of wrong predicted concepts in *J*.

(2) *Retrieval perspective*. Retrieval performance measures can be used to evaluate the annotation quality. Auto-annotated test images are retrieved using concepts from the vocabulary. The relevance of the retrieved images is verified by evaluating it against the manual annotations of the images. Precision and recall values are computed for every concept in the test set. Precision is represented by the percentage of retrieved images that are relevant. Recall is represented by the percentage of relevant images that are retrieved. For a given query concept *wq*, precision and recall are defined as [30]:

$$precision(w_q) = \frac{|\{J \in T' \mid w_q \in W_J^a \wedge w_q \in W_J\}|}{|\{J \in T' \mid w_q \in W_J^a\}|} \qquad (7)$$

$$recall(w_q) = \frac{|\{J \in T' \mid w_q \in W_J^a \wedge w_q \in W_J\}|}{|\{J \in T' \mid w_q \in W_J\}|} \qquad (8)$$

The average precision and recall over different single-concept queries are used to measure the overall quality of automatically generated annotations.

It can be useful to measure the number of single-concept queries for which at least one relevant image can be retrieved using the automatic annotations. This metric completes the average precision and recall by providing information about how wide the range of concepts that contribute to the average precision and recall is. It is defined as [30]:

$$|\{w_q \mid precision(w_q) > 0 \wedge recall(w_q) > 0\}| \qquad (9)$$

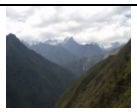
5.2 *Experimental results*

(1) Annotation perspective

In order to evaluate the annotation system we have used a testing set of 400 images that were manually annotated and not included in the training set used for the CMRM model. This set was segmented using the segmentation algorithm described above and a list of concepts having the joint probability greater than a threshold value was assigned to each image. Then the number of relevant concepts automatically assigned by the annotation system was compared against the number of concepts manually assigned by computing

an accuracy value for both modules. In table 3 the *Accuracy_1* column contains the values obtained using the initial version of the CMRM model and *Accuracy_2* contains the values obtained the modified version of the CMRM model. The average accuracy value obtained for the initial model was 0.46 and the average accuracy value obtained for the modified model was 0.54. It can be observed that from the accuracy point of view the modified version produces better results.

Table 3. Accuracy values obtained for both models

	Name	**Image**	**Accuracy 1**	**Accuracy 2**
0	25.jpg		0.5	0.75
1	28.jpg		0.75	0.75
2	46.jpg		0.33	0.66
3	60.jpg		0.33	0.66
4	76.jpg		0.5	0.75
5	82.jpg		0.25	0.75
6	91.jpg		0.33	0.33
7	97.jpg		0.5	0.75
8	132.jpg		0.33	0.33
9	176.jpg		0.5	0.5

(2) Retrieval perspective

The precision and recall charts are presented in Fig.8. and Fig.9., where it was used the following convention to distinguish between the two models: the values corresponding to concepts ending with underline (e.g sky_) belong to the proposed modified model.

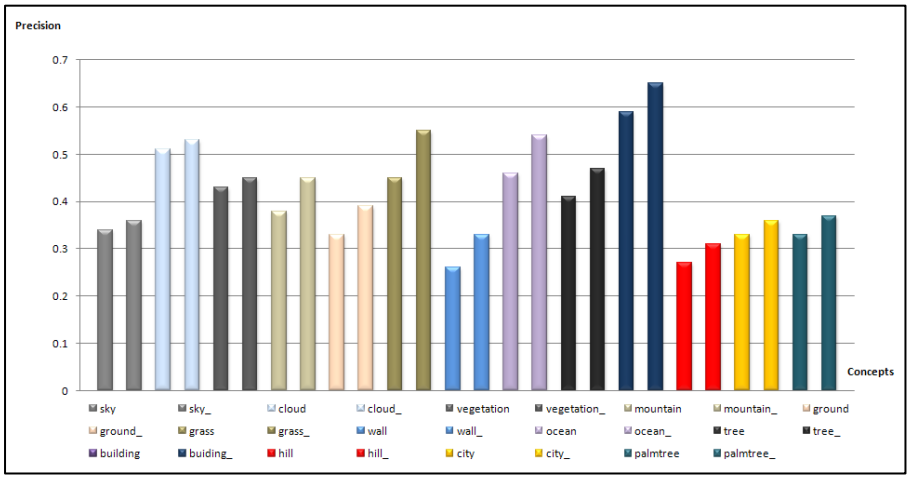


Fig.8. Precision chart for both models

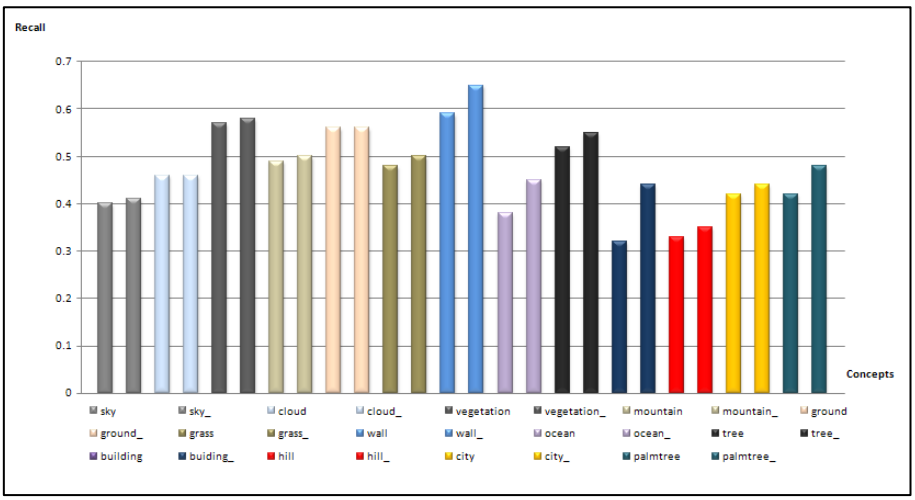


Fig.9. Recall chart for both models

After computing the precision and recall values for all concepts (not all concepts were shown in the charts due to space limitation) it was computed a mean precision equal to 0.38 (0.34 obtained using the standard version) and a mean recall equal to 0.44 (0.36 obtained using the standard version). It can be observed that the values corresponding to

our proposed modified model are always greater than the values of the initial model. This is justified because the modified version is taken into account only the concepts and the images considered 'relevant'. Based on the experimental results it can be concluded that the modified version produces better annotation results.

6 Conclusions and Future Work

The paper describes the extension of an image annotation model that can be used for annotating natural images. The CMRM annotation model has proved to be very efficient by several studies. This model learns the joint probability of concepts and blobs based on a well know benchmark: SAIAPR TC-12. This benchmark contains a large-size image collection comprising diverse and realistic images, includes an annotation vocabulary having a hierarchical organization, well defined criteria for the objective segmentation and annotation of images. Because the quality of an image region and the running time of the segmentation process are two important factors for the annotation process we have used a segmentation algorithm based on a hexagonal structure which was proved to satisfy both requirements: a better quality and a smaller running time. Each new image was annotated with concepts taken from an ontology created starting from the information provided by the benchmark: the hierarchical organization of the vocabulary and the spatial relationships between regions. For storing the information required by the annotation process it was used an object oriented database called db4o. The object oriented approach has simplified the way of describing the modified version. The experimental results realized from two perspectives (annotation and retrieval) have proved that our proposed modified model produces better results that the initial model. In the future it is intended to evaluate the modified version from the semantic base image retrieval point of view, using the two methods provided by CMRM: Annotation-based Retrieval Mode and Direct Retrieval Model.

References

http://picasa.google.com/
http://www.flickr.com/
db4objects, http://www.db4o.com/
Db4o Developer Community, http://developer.db4o.com/
"Segmented and Annotated IAPR TC-12 dataset", http://imageclef.org/SIAPRdata
"IAPR TC-12 Benchmark", http://imageclef.org/photodata
Blei, D.M; Jordan M. I (2003): Modeling annotated data. Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval pp. 127 – 134.
Burdescu, D.; Brezovan, M.; Ganea, E; Stanescu, L.(2009): A New Method for Segmentation of Images Represented in a HSV Color Space, Lecture Notes in Computer Science, 5807,pp. 606-617.
Clough, P.; Grubinger, M.; Deselaers, T.; Hanbury, A. ; Müller, H. (2006): Overview of the ImageCLEF 2006 photographic retrieval and object annotation tasks, Evaluation of Multilingual and Multimodal Information Retrieval – 7th Workshop of the CLEF, LNCS, 4730, Springer, Alicante, Spain, pp. 579–594.

Datta, R.; Joshi, D.; Li, J.; Wang, J.Z. (2008): Image retrieval: ideas, influences, and trends of the new age, ACM Computing Surveys, 40(2), pp. 1–60.

Deerwester, S.C; Dumais, S.T.; Landauer, T.K.; Furnas, G.W.; Harshman, R.A (1990): Indexing by latent semantic analysis. *Journal of the Society for Information Science*, 41(6), pp.391–407.

Duygulu, P.; Barnard, K., de Freitas, N.; Forsyth, D. (2002): Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In Seventh European Conf. on Computer Vision, pp. 97–112.

Escalante, H.J.; Hernández, C. A.; Gonzalez, J. A.; López, A. L.;Montes, M.; Morales, E. F. ; Sucar, L. E.; Villaseñor L.; Grubinger, M. (2010) : The segmented and annotated IAPR TC-12 benchmark,Computer Vision and Image Understanding,114, Issue 4, pp. 419-428.

Feng, S. L.; Manmatha, R. ; Lavrenko, V. (2004): Multiple bernoulli relevance models for image and video annotation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1242–1245.

Felzenszwalb P.; Huttenlocher, D. (2004) : Efficient Graph-Based Image Segmentation, Intl J. Computer Vision, 59(2).

Grubinger, M.; Clough, P; Hanbury, A.; Müller, H. (2007): Overview of the ImageCLEF 2007 photographic retrieval task, Advances in Multilingual and Multimodal Information Retrieval – 8th Workshop of CLEF, LNCS, 5152, Springer, Budapest, Hungary, pp. 433–444.

Hofmann, T. (2001): Unsupervised learning by probabilistic latent semantic analysis. *MachineLearning*, 42(1-2), pp.177–196.

Inoue, M. (2004): On the need for annotation-based image retrieval. Workshop on Information Retrieval in Context.

Jeon, J.; Lavrenko, V.; Manmatha, R. (2003): Automatic Image Annotation and Retrieval using Cross-Media Relevance Models. In: Proceedings of the 26th Intl. ACM SIGIR Conf., pp. 119–126.

Jeon J. ; Manmatha R. (2004) : "Using maximum entropy for automatic image annotation.," in CIVR, pp. 24–32.

Lavrenko, V.; Croft, W. (2001): Relevance-based language models, Proceedings of the 24th annual international ACM SIGIR conference, pp. 120-127.

Lavrenko, V.; Choquette, M.; Croft, W. (2002): Cross-lingual relevance models, Proceedings of the 25th annual international ACM SIGIR conference, pp. 175-182.

Lavrenko, V.; Manmatha, R.; Jeon J. (2004): A model for learning the semantics of pictures. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*.

Li, J.; Wang, J. (2003): Automatic linguistic indexing of pictures by a statistical modeling approach. IEEE Transactions on Pattern Analysis and Machine Intelligence 25.

Li, J.; Wang, J.Z. (2008):Real-time computerized annotation of pictures, IEEE transactions on pattern analysis and machine intelligence, 30(6), pp. 985-1002.

Monay F.; Perez D. G. (2004) : Plsa-based image auto-annotation: constraining the latent space. In Proceedings of ACM International Conference on Multimedia (ACM MULTIMEDIA), pp. 348–351.

Mori, Y., Takahashi, H., Oka, R.(1999): Image-to-word transformation based on dividing and vector quantizing images with words. In: MISRM'99 First Intl. Workshop on Multimedia Intelligent Storage and Retrieval Management .

Paterson, J.; Edlich, S.; Hoerning, H.; Hoerning, R. (2006): The Definitive Guide to db4o, Apress.

Rong J.; Chai, J.Y; Si, L. (2004) : Effective automatic image annotation via a coherent language model and active learning. In *Proceedings of ACM International Conference on Multimedia (ACM MULTIMEDIA)*, pp. 892–899.

Shah, B.; Benton, R.; Wu, Z., Raghavan, V. (2007): Automatic and Semi-Automatic Techniques for Image Annotation. In Semantic – Based Visual Information Retrieval book, chapter VI, Yu-Jin Zhang, IRM Press.