

TWO-TIER IMAGE ANNOTATION MODEL BASED ON A MULTI-LABEL CLASSIFIER AND FUZZY KNOWLEDGE REPRESENTATION SCHEME

Marina Ivasic-Kos¹, Miran Pobar¹, Slobodan Ribaric²

¹*Department of Informatics, University of Rijeka, Rijeka, Croatia*

²*Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia*

marinai@uniri.hr

Keywords: Image annotation, Knowledge representation, Inference algorithms, Fuzzy Petri Net, Multi-label image classification

Abstract: Automatic image annotation deals with automatically assigning useful keywords to an unlabelled image. The major goal is to bridge the so-called semantic gap between the available image features and keywords that people might use to annotate images. Although different people will most likely use different words to annotate the same image, most people can use object or scene labels when searching for images. We propose a two-tier annotation model where the first tier corresponds to object level and the second tier to scene level annotation. In the first tier, images are annotated with labels of objects present in them, using multi-label classification methods on low-level features extracted from images. Scene level annotation is performed in the second tier, using an inference engine of the fuzzy knowledge-representation scheme based on the Fuzzy Petri Net (KRFPN) and the object labels obtained at the first tier. The scenes and object classes are graphically represented by places in the KRFPN scheme and the relationships between these classes are represented by transitions. The inference engine of the KRFPN supports scene recognition, efficient inconsistency checking of object labels, as well as inference of more general concepts. The proposed model was experimentally tested for annotation of a dataset of outdoor images and the results were compared to the published results obtained on the same image collection. Different subsets of features composed of dominant colours, image moments, and GIST descriptors as well as different classification methods (RAKEL, ML-kNN and Naïve Bayes) were tested in the first tier. Due to the fuzzy-knowledge representation scheme, the obtained image annotation is enriched with new, more abstract concepts that are close to the concepts people use to interpret these images.

1. INTRODUCTION

Image retrieval, search and organization became a problem due to the huge number of images produced daily. In order to simplify these tasks, different approaches for image retrieval have been proposed that can be roughly divided into those that compare visual content (content based image retrieval) and those that use text descriptions of images (text based image retrieval) [Smeulders et al. 2000, Datta et al. 2008].

Image retrieval based on text appeared to be easier, more natural and more suitable for people in most everyday cases. This is because it is much easier to write a keyword based query than to provide image

examples, and it is likely that the user does not have an example image of the query. Also, images corresponding to the same keywords can be very diverse. For example, a person can search for a different view of the same town that looks very different to an image he already has, in which case content-based retrieval would not be the best choice. On the other hand, with a text query very diverse images can be retrieved with the same keywords, e.g. Rijeka (town, river...).

To be able to retrieve images using text, they must be labelled or described in the surrounding text, and the problem is that most of the images are neither of that. Manually providing image annotation is a tedious and expensive task, especially when dealing with a large number of images, so automatic image annotation appeared as a solution.

Automatic annotation methods deal with visual features that can be extracted from the raw image data, such as colour, texture, structure, etc. and can automatically assign metadata in form of keywords from a controlled vocabulary to an unlabelled image. The major goal is to bridge the so-called semantic gap [Hare et al. 2006] between the available features and the keywords or interpretation of the images that could be useful to humans.

This problem is challenging because different people will most likely annotate the same image with different words that reflect their knowledge about the context of the image, their experience, cultural background, etc. However, most people when searching for images use object or scene labels. Therefore, in this paper we focus on automatic image annotation on scene and object levels, Fig. 1.

Object labels correspond to objects that can be recognized in an image, like *sky*, *trees*, *tracks* and *train* for the image in Fig. 1. Scene labels represent the context of the whole image, like *SceneTrain* or more general *City*, and can be either directly obtained as a result of global classification of image features [Oliva and Torralba, 2001] or inferred from object labels as was proposed in our approach.



		
Object labels	<i>tracks, train, cloud, sky, trees,</i>	<i>snow, polar bear</i>
Scene label	<i>SceneTrain, Transportation</i>	<i>ScenePolarbear, WildLife, Arctic</i>

Figure 1. Examples of images and their annotation on object and scene levels

In this paper, we propose a two-tier annotation model for automatic image annotation, where the first tier corresponds to object and second to scene level annotation. An overview of the proposed model is given in Section 3 after sections with related work. The first assumption is that there can be many objects in each image, but an image can be classified into one scene. The second is that there are typical objects of which scenes are composed. Since many object labels can be assigned to an image, the object level annotation was treated as a multi-label problem and appropriate multi-label classification methods RAKEL and ML-kNN have been used. On the other hand, the scene level annotation task was performed using an inference engine of the fuzzy knowledge-representation scheme based on the Fuzzy Petri Net (KRFPN) and the object labels obtained

at the first tier. The usage of fuzzy knowledge-representation scheme reflects on the major contributions of this paper that are as follows:

- use of inference engine that is integrated into the fuzzy knowledge-representation scheme for inconsistency checking of classification results on object level annotation and for automatic scene recognition;
- use of representation model based on Fuzzy Petri Nets to graphically represent knowledge about domain images;
- combination of multi-label classification and graph-based approach to image annotation;
- adaptive two-tier annotation model in which each level can be independently used and modified.

The remainder of the paper is organized as follows: In Section 4 the KRFPN fuzzy-knowledge representation scheme formalism is described and an example of the scheme adapted to the outdoor image domain is presented. The application of the fuzzy inference engine for inconsistency checking of the classification results at the object level of annotation and the application for scene recognition is given in Sections 5 and 6, respectively. The performance of the proposed two-tier automatic annotation system was evaluated on outdoor images considering different feature subsets (dominant colours, moments, GIST descriptors) and compared to the published results obtained on the same image database as detailed in Section 7. The paper ends with a conclusion and directions for future work, Section 8.

2. RELATED WORK

Automatic image annotation (AIA) has been an active research topic in recent years due to its potential impact on image retrieval, search, image interpretation and description. AIA approaches proposed so far can be divided in various ways, e.g. according to the theory they are most related to (statistical theory and machine learning, logical reasoning and artificial intelligence) [Deruyver et al. 2009] or semantic level of concepts that are used for annotation (flat or structured vocabularies) [Tousch et al., 2012].

Classical AIA approaches belonging to the field of machine learning look for a mapping between image features and concepts on object or scene levels. Classification and probabilistic modelling have been extensively used for that purpose. Methods based on classification like one described in [Li and Wang, 2008] treat each of the semantic keywords or concepts as independent classes and assign each keyword to one classifier. Methods based on translation model [Duygulu et al. 2002] and methods which use latent semantic analysis [Monay and Gatica-Perez, 2003] fall into the category of probabilistic methods that aim to learn a relevance model to represent correlations between images and keywords. A recent survey of research made in that field can be found in [Datta et al. 2008, Zhang et al. 2012].

Lately graph based methods have been intensively investigated to apply logical reasoning on images and many graph-based image analysis algorithms have proved to be successful. For image annotation, this graph is a conceptual graph which encodes the interpretation of the image. [Pan et al., 2004] have proposed a graph based method for image annotation in which images, annotations and regions are considered as three types of nodes of a mixed media graph. In [Liu et al. 2008], automatic image annotation is performed using two graph-based learning processes. In the first graph, the nodes are images and the edges are relations between images,

and in the second graph the nodes are words and the edges are relations between words. In [Deng et al. 2009] authors intend to illustrate each of the concepts from the WordNet ontology with 500-1000 images in order to create public image ontology, the ImageNet.

Within the project aceMedia, in [Mezaris et al. 2009] ontology is combined with fuzzy logic to generate concepts from the beach domain. In [Athanasiadis et al. 2009], the same group of authors has used a combination of different classifiers for learning concepts and fuzzy spatial relationships.

In [Ivašić-Kos et al. 2010] a framework based on fuzzy Petri Nets is proposed for image annotation on object level. In the fuzzy knowledge base, nodes are features and objects. Co-occurrence relations are defined between objects and attribute relations are defined between objects and features.

[Binder et al. 2013.] have proposed a method called Output Kernel Multi-Task Learning (MTL) to improve ranking performance by transfer information between classes. [Zhang et al. 2014] have defined a graph-based representation for loosely annotated images where each vertex is defined as a collection of discriminative image patches annotated with object category labels. The edge linking two nodes models the co-occurrence relationship among different objects in the same image.

3. OVERVIEW OF THE TWO-TIER IMAGE ANNOTATION MODEL

Images of outdoor scenes commonly contain one or more objects of interest like *person*, *boat*, *dog*, *bridge* and different kinds of background objects such as *sky*, *grass*, *water* etc. However, people often think about these images as a whole, interpreting them as scenes, for example, *tennis match* instead of *person*, *court*, *racquet*, *net*, and *ball*. To make the image annotation more useful for organizing and retrieval of images, it should contain both object and scene labels. Object labels correspond to classes whose instances can be recognised in an image. Scene labels are used to represent the context or semantics of the whole image, according to common sense and expert knowledge.

The overview of the proposed two-tier automatic image annotation model using a multi-label classifier and fuzzy knowledge representation scheme is depicted in Figure 2. The input to the system is an unlabelled image and the results of automatic annotation are object and scene labels. First, from each image, low-level features are extracted which represent the geometric and photometric properties of the image. Each image is then represented by the m -component feature vector $\mathbf{x} = (x_1, x_2, \dots, x_m)^T$. Obtained features are used for object classification. The assumption is that can be more than one object relevant for image annotation, so multi-label classification methods are used. The result of the image classification at the first tier is o object labels from the set C of all object labels. The object labels are used for scene-level classification supported by the image-domain knowledge base at the second tier. Each scene in the knowledge base is defined as an aggregation of typical object classes. Therefore, it is possible to conclude which scene is most likely for the given set of object labels. Furthermore, chunks of knowledge, particularly those related to the relationships among objects and scenes, can also be used to check the consistency of object labels and to discard those that do not fit the context.

The knowledge base is represented with a knowledge-representation scheme based on the Fuzzy Petri Net [Ribarić and Pavešić, 2009]. The proposed scheme has the ability to cope with uncertain, imprecise, fuzzy

knowledge about concepts and relations, as well as to make conclusions about concepts and their relations. Definition of knowledge-representation scheme related to image-domain and using of fuzzy inference engine for scene recognition and inconsistency checking follows.

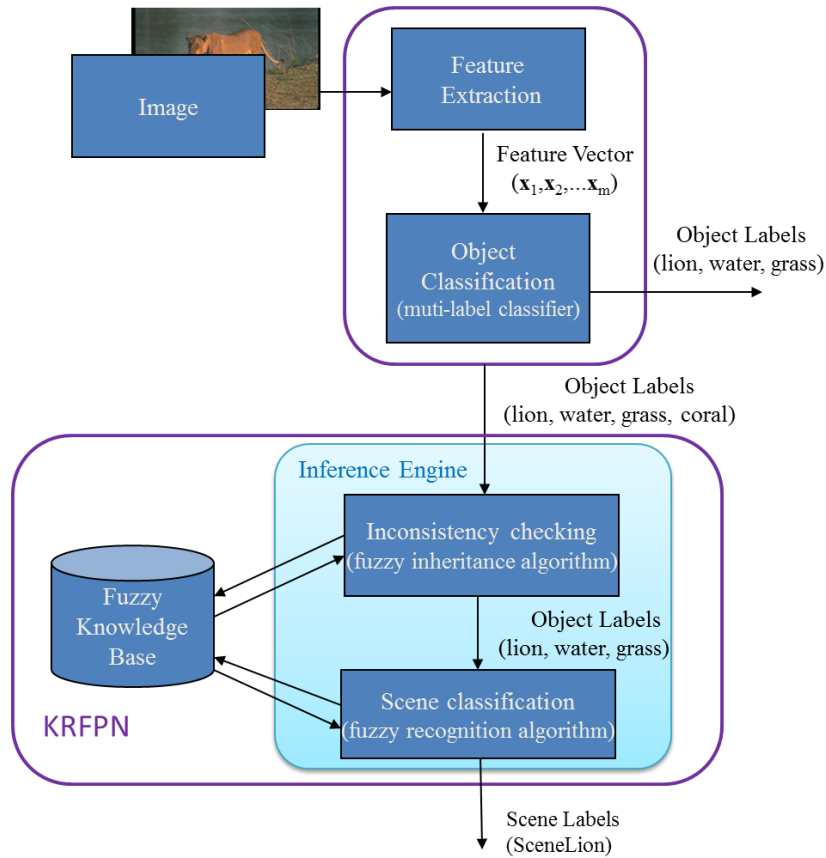


Figure 2. Framework of two-tier automatic image annotation system

3.1 Feature Sets

The variety of perceptual and semantic information about scenes and objects on the outdoor image could be contained in global low-level features such as dominant colour, spatial structure, colour histogram, texture, etc. Therefore, for both the object and scene level annotation we used the same features extracted from images. We have used a feature set made up of dominant colours of the whole (global dominant colours) and parts of the image (region based dominant colours), colour moments and the GIST descriptor.

The colour histogram was calculated for each of the RGB colour channels of the whole image. Next, histogram bins with the highest values for each channel were selected. These bins correspond to dominant colours in decreasing order. After experimenting with different numbers of dominant colours (3, 6, 8, 12, 16, 24 and 36), we have chosen to use 12 dominant colours per channel (referred to as DC) in each image as features for our classification tasks.

The information about the colour layout of an image was preserved using 5 local RGB histograms, from which dominant colours were extracted in the same manner as for the whole image. The local dominant colours are referred to as DC1 to DC5. To calculate the DC1, DC2 and DC3 local features, a histogram was computed for each cell of a 3x1 grid applied to each image. The DC4 feature was computed on the central part

of the image, presumably containing the main image object, and DC5 feature on the surrounding part that would probably contain the background, Fig. 3. The size of the central part was 1/4 of the diagonal size of the whole image, and of the same proportions. The size of DC vector is 36 and the size of local DC vectors (DC1..DC5) is 180. Additionally, we have computed the colour moments (CM) for each RGB channel: mean, standard deviation, skew and kurtosis. The size of CM feature vector is 12.



Figure 3. The arrangement of a) 3x1 image grid and b) central and background regions from which the dominant colours features were computed

The GIST image descriptor [Oliva and Torralba, 2001] that was proved to be efficient for scene recognition was also used as a region-based feature. It is a structure-based image descriptor that refers to the dominant spatial structure of the image characterized by properties of its boundaries (e.g., the size, degree of openness, perspective) and its content (e.g., naturalness, roughness). The spatial properties are estimated using global features computed as a weighted combination of Gabor-like multi scale-oriented filters. In our case, we used 8x8 encoding samples in the GIST descriptor within 8 orientations per 8 scales of image components, so the GIST feature vector has 512 components.

We performed the classification tasks using all the extracted features, in which case the size of the feature vector was 740. Since the size of feature vector is large in proportion to the number of images, we have also tested the classification performance using five subsets.

3.2 Image Annotation on Object Level

We attempt to label both foreground and background objects assuming that they are all useful for image annotation. Since we want to annotate the image with all object labels, the annotation of objects at the first tier is treated as a multi-label classification problem. Multi-label classification can be formally expressed as $\varphi: E \rightarrow \mathcal{P}(C)$, where E is a set of samples, $\mathcal{P}(C)$ is a power set of set of labels C and there exists at least one example e_j that is mapped into two or more classes, i.e. $\exists e_j \in E : |\varphi(e_j)| \geq 2$.

Methods most commonly used to tackle a multi-label classification problem can be divided into two different approaches [Tsoumakas and Katakis, 2007]. In the first, the multi-label classification problem is transformed into more single-label classification problems [Madjarov et al., 2012], known as problem transformation approach. The aim is to transform the data so that any classification method designed for single-label classification can be applied. On the other hand, algorithm adaptation methods extend specific learning algorithms in order to handle multi-label data directly.

For the multi-label classification task we have used the Multi-label k-Nearest Neighbour (ML-kNN) [Zang and Zhou, 2007], a lazy learning algorithm derived from the traditional kNN algorithm, and RAKEL (RANdom k-labELsets) [Tsoumakas and Vlahavas, 2007] that is an example of data adaptation methods. The

RAKEL algorithm considers a small random subset of labels and uses a single-label classifier for the prediction of each element in the powerset of this subset. In this way, the algorithm aims to take into the account label correlations using single-label classifiers. We used RAKEL with kNN and C4.5 classification tree as base classifiers. The base classifiers are applied on subtasks with manageable number of labels. It was experimentally shown that for our task the best results are obtained using RAKEL with the C4.5 as base classifier. We also used the Naïve Bayes (NB) classifier along with data transformation. The data was transformed so that each instance with multiple labels was replaced with elements of binary relation $\rho \subseteq E \times C$ between a set of samples E and a set of class labels C . A ordered pair $(e, c) \in E \times C$ can be interpreted as “ e is classified into c ” and is often written as epc . For example, if an image $e_{15} \in E$ was annotated with labels $\varphi(e_{15}) = \{lion, sky, grass\}$, $lion, sky, grass \in C$, it was transformed into three single-label instances $e_{15}\rho_{lion}$, $e_{15}\rho_{sky}$, $e_{15}\rho_{grass}$.

4. A FUZZY KNOWLEDGE-REPRESENTATION FORMALISM

The assumption is that there are typical objects of which scenes are composed, so each scene is treated as an aggregation of objects selected as typical, based on the used data set. Also, as scene inference depends on the objects that are obtained as classification results in the first tier of the system, it is useful to detect and discard objects that do not fit the context. The relationships between objects, particularly co-occurrence relations are used for this purpose.

To model relationships between objects and scenes and relationships among objects in an image, knowledge-representation formalism has to be used and domain knowledge needs to be included. Considering that automatic object classification is subject to errors and that knowledge about concepts is often incomplete, the ability to make conclusions from imprecise, fuzzy knowledge becomes necessary. A knowledge-representation scheme based on the Fuzzy Petri Net, named KRFPN, [Ribarić and Pavešić, 2009] is adapted at the second tier for this purpose.

4.1 Definition of the KRFPN Scheme Adapted for Image Annotation

The elements of the knowledge base used for interpretation of images from a part of a Corel image base [Carbonetto et al. 2004] are presented using the KRFPN scheme [Ribarić and Pavešić, 2009].

The KRFPN scheme is defined as 13-tuple: $KRFPN = (P, T, I, O, M, \Omega, \mu, f, c, \alpha, \beta, \lambda, Con)$, (1)

where:

$P = \{p_1, p_2, \dots, p_n\}$, $n \in \mathbb{N}$ is a set of places; a function $\alpha: P \rightarrow D$ maps a place from a set P to a concept from a set $d \in D$ used for image annotation. A α is a bijective function, so its inverse $\alpha^{-1}: D \rightarrow P$ is used in the schema too; set of concepts consists of object and scene labels, $D = C \cup SC$, $C = \{Airplane, Train, Shuttle, Ground, Cloud, Sky, Coral, Dolphin, Bird, Lion, Mountain, etc.\}$, $SC = \{SceneAirplane, SceneBear, SceneBird, SceneElephant, SceneFox, SceneCheetah, SceneGoat, SceneLion, ScenePolarbear, SceneRabbit, SceneTiger, SceneTrain, SceneWolf, SceneZebra, Inland, Mountains, Sea, Seaside, Space, Other\}$. Pairs of mutually contradictory concepts are not defined, but can be added, if needed.

$T = \{t_1, t_2, \dots, t_m\}, m \in \mathbb{N}$ is a set of transitions; a function $\beta: T \rightarrow \Sigma$ maps a transition from a set T to a relationship r from a set $\Sigma = \{\text{occurs_with}, \text{not_occurs_with}, \text{consists_of}, \text{is_part_of}\}$. The *occurs_with* is a relationship between object classes that models the joint occurrence of object classes in the image. A pair $(\text{occurs_with}, \text{not_occurs_with}) \in \text{Con} \subseteq (\Sigma \times \Sigma)$ contains mutually contradictory relations that are defined between objects classes. The aggregation relationship *consists_of* is defined between a scene class that has a role of aggregation and object classes that have the role of components of aggregation. For a relationship *consists_of* an inverse relationship $-(\text{consists_of}) = \text{is_part_of}$ is defined. The value of a transition, $f: T \rightarrow [0, 1]$, corresponds to the degree of truth and confidence related to the relationship mapped to that transitions and is defined according to the used training dataset. The link between places and transitions is given with the input and output functions, $I: T \rightarrow \mathcal{P}(P) \setminus \emptyset$ and $O: T \rightarrow \mathcal{P}(P) \setminus \emptyset$.

$M = \{m_1, m_2, \dots, m_r\}, r \geq 0$ is a set of tokens that are used to define execution of a Fuzzy Petri Net (FPN). The tokens' distribution within places is given as $\Omega(p) \in \mathcal{P}(M)$, where $\mathcal{P}(M)$ is a power set of M . The initial distribution of tokens Ω_0 defines the initial marking vector $\mu_0 = (\mu(p_1), \mu(p_2), \dots, \mu(p_n))$. In our case, in the initial marking, a place can have no or at most one token, $\mu(p_i) \in \{0, 1\}$. A place that contains one or more tokens is called a marked place and it is important for execution of the transition. Additionally, each token is associated with a value $c: M \rightarrow [0, 1]$ that corresponds to the degree of truth or confidence related to the concept mapped to the place where the token is. The complete information about a token m_i is given by the pair $(p_j, c(m_i))$, where the first component specifies the place where the token is located and the second one its value. The value of a token in an initial distribution can be set to the estimated a posteriori probability of the concept that is associated with that marked place.

$\lambda \in [0, 1]$ is a threshold value and has influence on inference procedures. It is usually set to a low value, determined experimentally, in our case 0.01.

4.2 Graphical representation of the KRFPN

The KRFPN scheme can be represented by a bipartite directed graph containing two types of nodes: places and transitions. Graphically, the places $p_i \in P$ are represented by circles and the transitions $t_j \in T$ by bars. A token $m_1 \in M$ is represented by dot within a place. The directed arcs between the places and transitions, and the transitions and places represent the transition input $I(t_j) \subseteq P$ and output $O(t_j) \subseteq P$ functions, respectively (Figure 3).

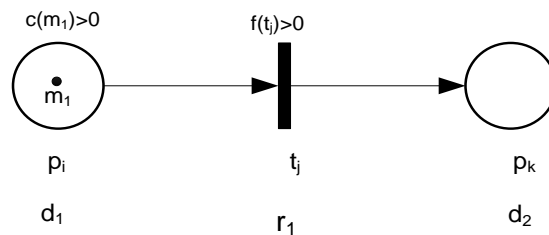


Figure 4. A generic form of a chunk of knowledge in the Fuzzy Petri Net formalism

In a semantic sense, each place from the set P corresponds to a concept $d_i \in D$ and any transition from set T to a relation $r_k \in \Sigma$. In our scheme, elements of set D are objects and scene classes. The elements of set T are relations between object classes and relations between object and scene classes. The assigned value $c(m_1)$ to a token at the input place $p_i \in I(t_j)$ expresses the degree of uncertainty and confidence of a concept d_i mapped to a particular place p_i , and the transition value $f(t_j)$ corresponds to the degree of uncertainty of a relationship r_i mapped to a transition t_j . The assigned values implement fuzziness in the scheme and can be expressed by truth scales, where 0 means “not true” and 1 “always true”.

4.3 Dynamic properties of the KRFPN

Dynamic properties of a KRFPN are related to firing of the enabled transitions in the Fuzzy Petri Net – FPN, i.e. the execution of a FPN. A transition is enabled when every input place of the transition is marked, i.e., if each of the input places of the transition has at least one token whose value $c(m_i)$ exceeds the threshold value $\lambda \in [0, 1]$. By firing, tokens simultaneously move from all the transition’s input places $p_i \in I(t_j)$ to the corresponding output places $p_k \in O(t_j)$. In Figure 3, there is only one input place for the transition t_j , $I(t_j) = p_i$ and only one output place $O(t_j) = p_k$. After the transition firing, a new token value $c(m_2)$ at the output place is obtained as $c(m_1)f(t_j)$ (Fig. 4). Firing of a transition is in accordance with the basic firing rules of the original PN [Chen et al. 1990].

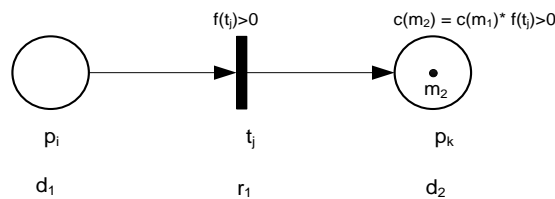


Figure 5. A new token value is obtained in the output place after firing.

The dynamic properties of the scheme are important for the definition of inference-engine. The inference engine on the KRFPN scheme consists of three automated reasoning processes: fuzzy inheritance, fuzzy recognition and fuzzy intersection. All the steps of the inference algorithms are given in [Ribarić and Pavešić, 2009], and below is a brief description of their application for inconsistency checking and scene recognition.

4.4 Modelling the confidence related to relationships and concepts

Given that the mapping between concepts and image features is often unreliable, and due to incomplete knowledge of the concepts, uncertainty is implemented into the scheme by means of transition and token values. The truth values of the relationships *consists_of* and *occurs_with* are computed using data in the training set, as explained below.

To define the truth value of the aggregation relationships *consists_of* it is assumed that a scene may contain several characteristic object classes, so the relation among the scene and object classes is an aggregation relationship where the scene plays the role of the aggregation and the elementary classes have the

role of the components of the aggregation. Analysing the data in the training set, common occurrence of object classes in the scene class is determined. The modified Bayes rule is used to form a set MS that contains object classes that are the most representative and discriminative for the given scene class. A set MS for a specific scene class $SC_i \forall_i$ is given by:

$$MS(SC_i) = \{C_k : \arg_i P(SC_i|C_k) \approx \arg_k \frac{P(X = C_k|Z = SC_i)}{P(X = C_k)} \geq \varepsilon\}. \quad (2)$$

$MS(SC_i)$ is a set of all those object classes $C_k, k = 1, 2, \dots$ that participate in a scene class SC_i with the posterior probability $P(SC_i|C_k), \forall_k C_k$ exceeding the marginal value $\varepsilon \geq 0.05$ that was experimentally determined. The $P(C_k)$ is the prior probability for a given object classes C_k obtained from the training set.

The truth value $f(t_l)$ of the transition that corresponds to the aggregation relationship *consists_of* between object and scene classes was determined using the Bayes rule for the posterior probability $P(SC_i|C_k), \forall_k C_k \in MS$ for each scene:

$$f(t_l) = P(SC_i|C_k) = \frac{P(C_k|SC_i)P(SC_i)}{\sum_{j=1}^{|SC_i|} P(C_k|SC_j)P(SC_j)}, \quad (3)$$

where t_l is the transition between concepts C_i and SC_i .

In Figure 5, a part of a knowledge base is presented, showing the relationships among a particular scene class SC_i and its component object classes from set $MS(SC_i)$ defined by the former procedure. For example, the degree of truth of the relation *consists_of* between the “Seaside” class and its component class “water”, determined by (3) is 0.95.

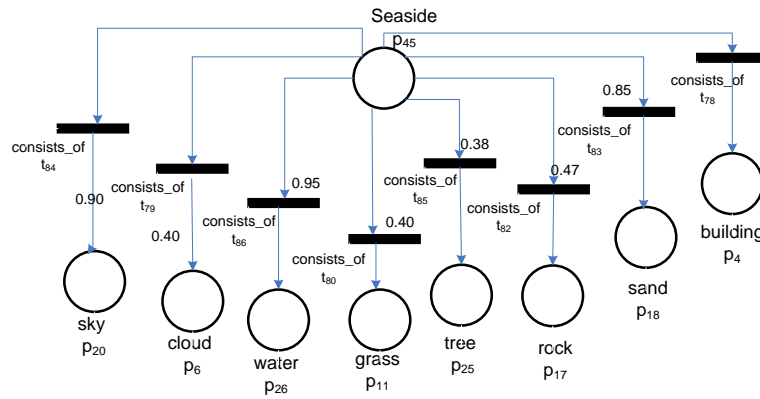


Figure 6. Relations *consists_of* among the scene “Seaside” and its object components

To define the truth value of the relationship *occurs_with*, co-occurrence of classes C_j and C_i is analyzed. This can be formally defined as:

$$P(C_j|C_i) = \frac{P(C_j \cap C_i)}{P(C_i)}, \quad i \neq j. \quad (4)$$

The *occurs_with* relationship is used to validate the results of the classification in the first tier and to check whether the results are consistent.

Spatial relationships between the objects like *above*, *next to*, and position such as *at the top*, *at the bottom*, have not been used in this experiment since these relationships could not be automatically computed from used

feature sets and are different from the natural relations. Although, if it turns out to be useful, spatial relationships as well as any new concept or relation, can easily be added to the scheme.

5. INCONSISTENCY CHECKING USING A FUZZY-INHERITANCE ALGORITHM

It is to be expected that some of the object labels obtained at the first tier of the system do not fit the context due to classification errors, so the facts included in the knowledge base related to the *occures_with* relation are used to check their consistency. The *occures_with* relation for each obtained object class should be analysed using the fuzzy-inheritance algorithm. It is assumed that the class with the highest confidence corresponds to the context, so those classes that have a lower truth value are analysed first.

For instance, let the unlabelled image e in Figure 6, represented with a feature vector \mathbf{x} , be given for automatic annotation. The multi-label classifiers generate a set of object classes $C_i \in \mathcal{C}$ for a given feature vector \mathbf{x} , as explained in Section 3.2. For this example the obtained classification result at the first tier of the automatic image annotation system is $\varphi(e) = \{sky, water, shuttle, rock, sand\}$. Note that the label *shuttle* is a result of misclassification because it is not present in the image.



Figure 7. Example of unlabelled image

Every obtained object class is checked for inconsistency using the fuzzy-inheritance algorithm in order to verify whether there is an *occures_with* relationship defined between that object class and other object classes in $\varphi(e)$. If the relation *not_occures_with* exists between checked object and all other objects in $\varphi(e)$, the object label is eliminated from the set $\varphi(e)$. The fuzzy-inheritance algorithm is based on the inheritance set of the KRFPN which is represented with a fuzzy inheritance tree, concepts that are derived from a reachability set of the ordinary Petri nets. The reachability set is defined as the smallest set of all reachable distributions of tokens starting from an initial distributions and recursively applying the firing of enabled transitions to obtain the immediately reachable distribution of tokens and is graphically represented by a reachability tree [Chen et al. 1990]. Main difference between reachability set and the inheritance set of KRFPN arise from the semantic interpretation of places. Namely, tokens in the output place of transitions associated with the places that represent the properties of concepts (here object labels of a scene) have to be frozen in order to stop further firing of the transitions. Also, inheritance tree can be bounded by $k+1$ levels, where k is the predefined number of levels. The root nodes $\pi_0^i, i = 1, 2, \dots$ of the inheritance trees are formed according to the i initially marked places and the corresponding degrees of truth. The nodes of inheritance trees have the form $\pi(p_j, c(m_l))$ $j = 1, 2, \dots, p, l = 1, 2, \dots, r, 0 \leq r \leq |M|$, where $c(m_l)$ is the value of a token m_l in place p_j .

For instance, for the elementary class *shuttle*, the appropriate place in the knowledge-representation scheme is determined by the function $\alpha^{-1}(shuttle) = p_{19}$, $shuttle \in C$, (Figure 7).

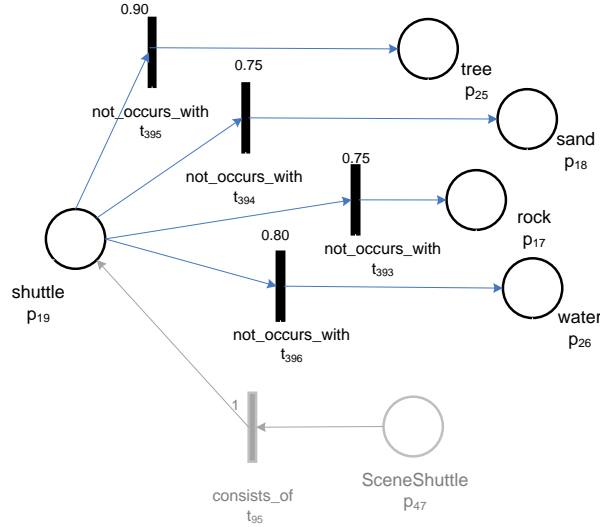


Figure 8. A part of KRFPN scheme related to elementary class “*shuttle*”

According to the initially marked place, the initial token distribution is created $\Omega_0 = (\emptyset, \emptyset, \dots, \{m_1\}, \dots, \emptyset)$ so that in place p_{19} there is one token m_1 , $\mu(p_{19}) = 1$. As explained, the corresponding root node of the inheritance tree is $\pi_0(p_{19}, \{1.0\})$. The inheritance tree is formed by firing the enabled transitions (whose firing creates new nodes) until the condition for stopping the algorithm is satisfied or the desired depth of the inheritance tree is reached. Figure 8 shows the inheritance tree on the KRFPN scheme and the appropriate semantic interpretation of the inheritance paths for the object class *shuttle*. For each of the inheritance paths, the measure of truth is determined by the token value in a leaf node (the node in which the algorithm stops). The arcs of the inheritance tree are marked with the label of a transition $t_j \in T$ and the value (t_j) , for example, $t_{394} = 0.75$. The generation of the inheritance trees may stop on a pre-defined level k or, as in this case, on terminal (T) or frozen (F) nodes. These nodes are frozen because they are the output nodes of the transitions that represent the co-occurrence relationship at which the hierarchical structure ends.

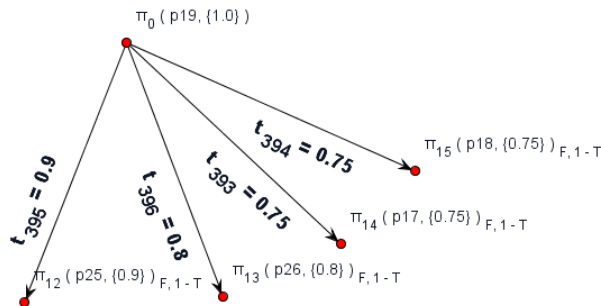


Figure 9. Inheritance tree for the class “*shuttle*” (Fig.8)

The obtained inheritance tree for the concept *shuttle* gives the conclusion that the class *shuttle* does not co-occur with the classes *tree*, *sand*, *water* and *rock*, so it can be concluded that the concept *shuttle* most likely does not match the context of the image depicted in Fig. 6 and should be discarded. Furthermore, the

concept, which is detected as an intruder because it does not belong to the context could be replaced with a concept that has similar properties but this opportunity is not used in this system.

After checking for the inconsistency, the refined image annotation at the object level is $\varphi(\mathbf{e}) = \{sky, water, rock, sand\}$.

The inheritance trees for the obtained object classes *sky*, *rock*, *sand* and *water* show that they can appear together in the image, so it can be concluded that they match the likely context.

6. SCENE CLASSIFICATION USING THE FUZZY-RECOGNITION ALGORITHM

For the task of scene classification for a new, unknown image, the fuzzy-recognition algorithm based on the inverse KRFPN scheme ($-KRFPN$) is used [Ribarić and Pavešić, 2009]. The $-KRFPN$ scheme is obtained by interchanging the position of the input I and the output O functions for the transition T in the 13-tuple. Additionally, by changing the position of the input and output functions, the relation mapped to the transition is transformed into its corresponding inverse relation. For example, for the relation *consist_of* in the KRFPN scheme its inverse relation *is_part_of* is used in the $-KRFPN$ scheme, i.e., $-(consist_of)=is_part_of$. Also, the co-domain of the associated function $c: M \rightarrow [0,1]$ that assigns values to the tokens is expanded by $c_r: M \rightarrow [-1, 1]$ so that in the case of an exception, a token may be associated with a negative value.

The procedure for the scene recognition is as follows. The results of the image annotation obtained at the first tier after inconsistency checking, are the input to the scheme used for further image annotation at the second tier. For the inference of unknown scene classes, it is assumed that a scene class is an aggregation of characteristic object classes.

The object classes C_i are mapped to the places $\{p_1, p_2, \dots, p_n\}$ using the function $\alpha^{-1}: C_i \rightarrow p_k$. If defined, the confidence based on a posterior probability of each object class C_i can be used as the token value $c_r(m_i)$ in the place p_k , e.g. if Naïve Bayes classifier is used. Otherwise, if confidence value for the classification result is unknown, the token value is set to 1.

For instance, let us take an image e depicted in Fig. 6. If the results of the image annotation at the first tier are object classes that exist in the knowledge base with the corresponding degrees of truth: (sky {0.5}, sand ({0.7}), rock ({0.4}), water ({0.6}), then by using the function α^{-1} the initially marked places are determined ($\alpha^{-1}(sky) = p_{20}$, $\alpha^{-1}(sand) = p_{18}$, $\alpha^{-1}(rock) = p_{17}$, $\alpha^{-1}(water) = p_{26}$). A small part of a $-KRFPN$ scheme with initially marked places and the corresponding token value is given in Fig. 9.

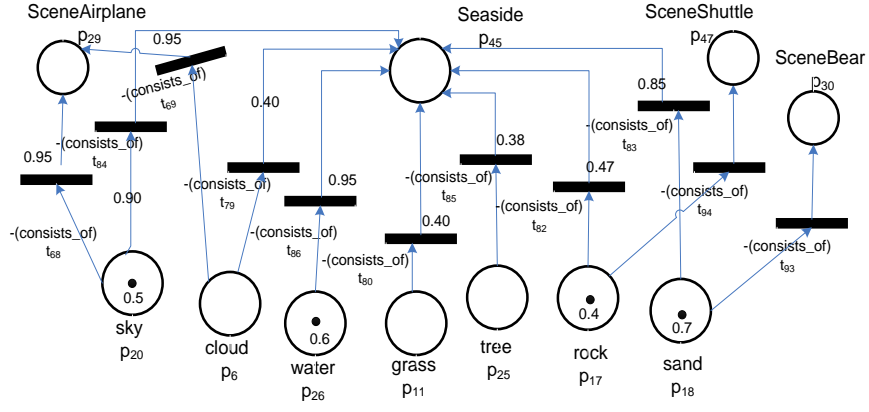
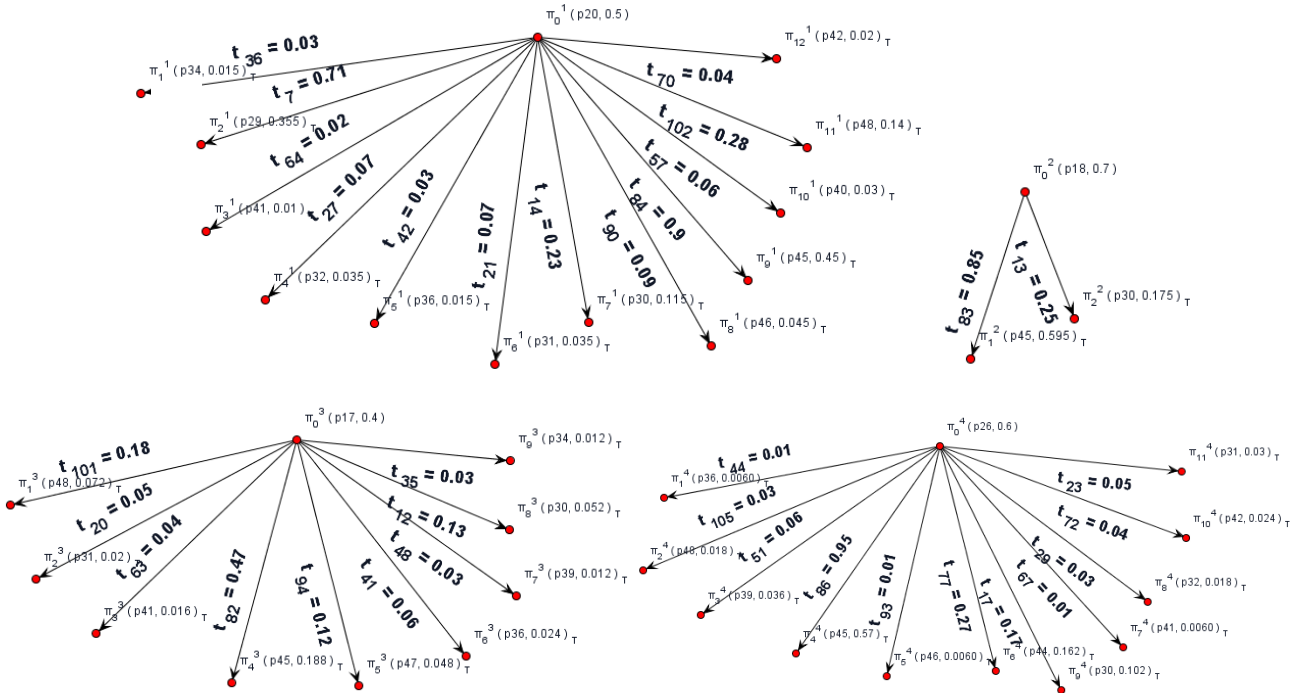


Figure 10. A small part of the inverse -KRFPN scheme for the scene recognition of the image depicted in Fig. 6.

According to the initially marked places and the corresponding degrees of truth, four root nodes $\pi_0^i, i = 1, \dots, 4$ of the recognition trees will be formed: $\pi_0^1(p_{20}, \{0.5\}), \pi_0^2(p_{18}, \{0.7\}), \pi_0^3(p_{17}, \{0.4\}), \pi_0^4(p_{26}, \{0.6\})$.

Figure 10 shows the corresponding recognition trees in the -KRFPN scheme with enabled transitions, starting from the root nodes. By firing the enabled transitions of the -KRFPN scheme, new nodes at the next higher level of the recognition tree are created and appropriate values $c_r(m_{new}) = c_r(m_k)f(t_l)$ of the tokens are obtained, where t_l is the transition between concepts C_i and SC_l , $c_r(m_k) = P(C_i)$ if the reliability $P(C_i)$ of the object class C_i is defined, or $c_r(m_k) = 1$ otherwise, $f(t_l)$ is defined as (3).

Note that only the recognition tree with the root node π_0^2 directly corresponds to the small part of -KRFPN depicted in Fig. 9. The leaf nodes of all the other trees are obtained based on the whole -KRFPN, which contains all the places that correspond to the scene classes.



The procedure of scene recognition using the fuzzy-recognition algorithm that corresponds to the recognition trees in Fig. 10 is described below. Due to the simplicity of the example, only one level of the recognition tree is generated.

Each leaf node π_i^k in the recognition tree k is represented by a vector of dimension $|P|$, where P is the set of places, so that the index of a node in the recognition tree corresponds to the index of the vector component and the value of a node is assigned to a value of the vector component. For example, a node $\pi_1^2 = (p_{45}, 0.595)$ is represented by the vector $\pi_1^2 = (0, 0 \dots 0, 0.595, 0, \dots, 0)$ so that all the vector components are assigned a value 0, except the 45th vector component, to which a node value of 0.595 is assigned. Accordingly, the total sum Z of all the nodes represented by the distribution vectors $\pi_i^k, i = 1, 2, \dots, o^k$ is computed:

$$Z = \sum_{k=1}^b \sum_{i=1}^{o^k} \pi_i^k, \quad (5)$$

where i is the index of the leaf nodes in the recognition tree k , $o^k \leq |P|$ is the total number of leaves in the recognition tree k , and k is the index of the recognition tree, $k = 1, 2, \dots, b \leq |M|$. The obtained vector Z represents the ranking of scene classes according to confidence values obtained by the fuzzy recognition algorithm.

In this example $b = 4, o^1 = 12, o^2 = 2, o^3 = 9, o^4 = 11$ and the total sum is:

$$Z = \sum_{k=1}^4 \sum_{i=1}^{o^k} \pi_i^k = \sum_{i=1}^{12} \pi_i^1 + \sum_{i=1}^2 \pi_i^2 + \sum_{i=1}^9 \pi_i^3 + \sum_{i=1}^{11} \pi_i^4 = (0 \dots 0, 0.36, 0.44, 0.09, 0.05, 0, 0.03, 0, 0.04, 0, 0, 0.05, 0.03, 0.03, 0.04, 0, 0.16, 1.80, 0.05, 0.05, 1.11, 0, \dots 0).$$

For example, the 30th component of the vector Z with the value 0.44 is obtained by summing all the values of the nodes in all the recognition trees that correspond to the place p_{30} (i.e. $\pi_7^1, \pi_2^2, \pi_8^3, \pi_9^4$): $0.115 + 0.175 + 0.052 + 0.102 = 0.44$

Then, a set of indices of elements with the highest sum $Z = (Z_1, Z_2, \dots, Z_{|P|})$ among all of the nodes in all the recognition trees is selected as:

$$i^* = \arg \max_{i=1, \dots, |P|} \{Z_i\}. \quad (6)$$

In the case that there are several i for which the same maximum value of $\{Z_i\}$ is obtained, the set I^* is created:

$$I^* = \{i_1^*, i_2^*, \dots\}. \quad (7)$$

A scene class assigned to a place with the max argument $p_i: i \in I^*$ is chosen as the best match for a given set of object classes obtained during image annotation the first tier. In this example, the 45th component of the vector Z has the maximum value 1.80. Therefore, a set of max arguments consists of only one element $i_1^* = 45$, so only one scene class is chosen as the best match, i.e., the one that is assigned to a place with that max argument, $\alpha(p_{45}) = Seaside$. The next scene candidate is “*Inland*” with a value of 1.11.

By merging the labels that are so far associated with the image, the two-tiered annotation of the image is formed. For example image e in the Figure 6 above, the annotation is $\varphi''(e) = \{sky, rock, sand, water\} \cup \{Seaside\}$.

7. EVALUATION OF THE PROPOSED TWO-TIER MODEL

We have compared the results of automatic image annotation using different subsets of features at the first tier of our system with previously published results and with the results of the second tier. The obtained labels on object level are refined using inconsistency checking, and the impact of that process is analysed on the scene level. The achieved results are averaged over 3 runs, since 3-fold cross validation was used. The classification performance on the object level was measured in terms instance-based and label-based accuracy, precision, recall and F1 score [Tsoumakas and Katakis, 2007].

7.1 Evaluation measures

The instance-based evaluation measures are based on the average differences of the actual and the predicted sets of labels over all examples in the test dataset. The label-based evaluation measures assess the predictive performance for each label separately and then average the performance over all labels [Tsoumakas and Katakis, 2007]. These measures are used due to the fact that an instance may not only be correctly or incorrectly annotated, but also partially correctly in case of multi-label classification. For example, if an image should be annotated with grass, sky, wolf, and is automatically annotated with tree, sky, dog, cloud, then the evaluation measure should reflect the insertion of wrong labels (tree, dog, cloud), missing labels (wolf, grass) and correct labels (sky).

To define the evaluation measures, we assume that an instance $e_j \in E$, $j = 1..N$ should be classified into the set of true object labels $Y_j = \{C_l, C_m, \dots, C_r\}$, $Y_j \subseteq C$ where E is a set of images, C is a set of all class labels and $N = |E|$ corresponds to the number of images in the set E . For an example e_j , the set of labels that are predicted by a classifier is denoted as Z_j .

Instance based accuracy is defined as the average ratio of correctly assigned and all labels assigned to each example by the classifier and the true labels:

$$Accuracy_{ins} = \frac{1}{N} \sum_{i=1}^N \frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}$$

Instance based precision is defined as the average ratio of correctly assigned and all labels assigned to each example by the classifier:

$$Precision_{ins} = \frac{1}{N} \sum_{i=1}^N \frac{|Y_i \cap Z_i|}{|Z_i|}$$

Instance based recall is defined as the average ratio of labels correctly assigned by the classifier and all labels in the ground truth for each example:

$$Recall_{ins} = \frac{1}{N} \sum_{i=1}^N \frac{|Y_i \cap Z_i|}{|Y_i|}$$

Instance based F-Measure is the harmonic mean of precision and recall:

$$F1_{ins} = \frac{1}{N} \sum_{i=1}^N \frac{2|Y_i \cap Z_i|}{|Z_i| + |Y_i|}$$

These measures reach their best value at 1 and the worst at 0.

Label based measures are computed firstly by computing the instance-based measure and then averaging over all labels. In the case of scene classification, which is treated as single-label classification, $|Y_j| = |Z_j| = 1$ and label-based measures are used.

7.2 Data

To evaluate the proposed two-tier model for image annotation a part of the Corel image dataset related to outdoor scenes [Barnard et al. 2003] was used.

The features were extracted from images that were sized 128 x 192 pixels or 192 x 128 pixels. We have used images labelled with one or more labels from the set C of 54 object classes related to natural and artificial objects such as *Airplane, Bird, Lion, Train* etc. and background objects like *Ground, Sky, Water* etc, provided by [Duygulu et al. 2002]. Additionally, we have labelled the images with one of the 20 elements form the set SC related to outdoor scenes such as *SceneTrain, SceneLion, Inland*, etc.

Some labels were too rare to effectively train the classifier and images that correspond to those labels were excluded from data. The resulting data were more suitable for learning of classification models. The details of the data set before and after simplification are presented in Table 1.

Table 1. Statistic of original and simplified data sets

Statistic	Original data		Simplified data	
	Objects	Scenes	Objects	Scenes
No. of labels	54	20	22	12
Max images per label	248	81	220	77
Min images per label	1	1	9	15
Mean images per label	26	25.2	50	32
Median images per label	7.5	19	28	25.5
Std. dev. per label	50	22	56	21

7.3 Annotation results

The label and instance based evaluation results for annotation on object level are presented in Table 2 considering different feature subsets and classifiers (RAKEL, ML-kNN and NB along with the data transformation). Overall the best results considering the label-based F1 score are obtained using the NB classifier independently of the used feature subset, due to significantly better achieved recall than with other methods. However, on instance-based measures NB obtained the worst results with both precision and recall. The RAKEL-C4.5 performed best on instance-level but performed worse than NB on label-based measures. For both instance-based and label-based measures, the RAKEL-kNN achieved slightly lower results than the best classifier. The RAKEL-C4.5 also performed well with all subsets of features, although their dimensions varied between 48 in case of dominant colours and 740 when all features are used. The achieved results are better than published results for 28 object classes, on the same data set [Carbonetto et al. 2004, Ivašić-Kos et al. 2010]. In [Carbonetto et al. 2004] the authors have reported average precision for the task of automatic image annotation on object level achieved with the dMRF model based on Markov random fields defined in [Carbonetto et al. 2004] and the dInd translation model from [Duygulu et al. 2002]. The dMRF model achieves

a label-based precision of 21%, while the dInd model achieves the label-based precision of 20%. With that feature set and the Naïve Bayes classifier using image segments, label-based precision of 32.6% and recall of 27.5% was achieved[Ivašić-Kos et al. 2010].

Table 2. Evaluation results for object annotation level.

<i>Feature subset</i>	<i>Classification method</i>	<i>Label-based results for object level annotation</i>			<i>Instance-based results for object level annotation</i>		
		<i>Precision</i>	<i>Recall</i>	<i>F1 score</i>	<i>Precision</i>	<i>Recall</i>	<i>F1 score</i>
All	RAKEL-C4.5	0.39	0.28	0.30	0.61	0.54	0.54
	RAKEL-kNN	0.49	0.41	0.40	0.57	0.53	0.53
	ML-kNN	0.32	0.2	0.23	0.66	0.42	0.48
	NB	0.38	0.64	0.46	0.22	0.34	0.25
GIST	RAKEL-C4.5	0.35	0.26	0.27	0.58	0.50	0.50
	RAKEL-kNN	0.48	0.45	0.43	0.57	0.55	0.54
	ML-kNN	0.26	0.17	0.19	0.60	0.38	0.44
	NB	0.31	0.65	0.40	0.21	0.38	0.26
DC+CM	RAKEL-C4.5	0.40	0.27	0.29	0.60	0.50	0.52
	RAKEL-kNN	0.31	0.32	0.29	0.50	0.48	0.47
	ML-kNN	0.16	0.10	0.11	0.63	0.34	0.41
	NB	0.28	0.65	0.36	0.22	0.44	0.27
DC1..DC 5 + CM	RAKEL-C4.5	0.37	0.27	0.29	0.61	0.53	0.54
	RAKEL-kNN	0.37	0.31	0.30	0.51	0.46	0.47
	ML-kNN	0.19	0.11	0.12	0.63	0.33	0.41
	NB	0.32	0.67	0.41	0.22	0.39	0.26
DC+ DC1..DC 5 + CM	RAKEL-C4.5	0.39	0.27	0.30	0.59	0.52	0.52
	RAKEL-kNN	0.39	0.33	0.31	0.51	0.46	0.47
	ML-kNN	0.20	0.11	0.12	0.64	0.34	0.41
	NB	0.31	0.66	0.41	0.22	0.4	0.26

Often, the results of automatic annotation can include labels that do not correspond to the context of an image. By using the facts from the knowledge base and the co-occurrence relationships between object classes, the obtained results of the image annotation at the first tier can be refined using fuzzy inheritance algorithms for inconsistency checking. In our model those object classes that are obtained as a result of the image annotation in the first tier and did not fit the likely context are discarded. As a consequence, the average precision of the image annotation can be increased if there is only one intruder class among the object level annotation, Fig. 12(a). If the majority of labels for an instance are wrong, the inconsistency checking can discard the true labels and the precision falls, Fig. 12(b).



Image example:	 (a)	 (b)
Object labels before inconsistency checking	<i>coral, sky, wolf, trees, grass</i>	<i>shuttle, train, building</i>
Object labels after inconsistency checking	<i>sky, wolf, trees, grass</i>	<i>train, building</i>

Figure 12. A positive (a) and negative (b) example of inconsistency checking of object labels.

Automatic image annotation at the second tier of the proposed model is performed by the fuzzy-recognition algorithm of the KRFPN scheme, and the object classes from the annotation of the first tier obtained using RAKEL-kNN and all features. The obtained label-based precision is 61% and the recall is 55%. The results of the second tier are dependent on the results of the first tier that are used as input. For those scenes for which there is one main object class which is highly discriminant for that scene (e.g. *train* for *SceneTrain*), it is crucial to detect that object at the first tier. In this kind of scenes background objects that are common to most scenes do not play an important role, but in scenes without one prominent object (e.g. *Sea, Inland*) they are important. For example, in case of object-level annotation, the best F1 score is obtained for *train* (0.8), *tracks* (0.77) and *polarbear* (0.68), and the worst for *wolf* (0.07) classes. This is reflected on the scene level classification, where *SceneTrain* has the best precision (0.86) and *SceneWolf* among the worst (0.30). For background objects, the best F1 scores are for *sky* (0.65) and *grass* (0.66) and the worst F1 for *mountain* (0.11) and *clouds* (0.13). Differences in results on object level may be due to imbalanced number of examples per class and the fact that in case of multi-label classification, partially correct annotation is possible. In case of similar classes, e.g. *lion* and *tiger*, *cloud* and *sky* in multi-label classification, both labels can be assigned.

In Table 3, some examples of image annotation obtained by the proposed model are shown.

Table 3. Examples of two-tier image annotation.

Image example:				
First tier	<i>shuttle</i>	<i>train, tracks, sky</i>	<i>grass, tiger</i>	<i>water, sand, sky, road</i>
Second tier	<i>SceneShuttle</i>	<i>SceneTrain</i>	<i>SceneTiger</i>	<i>Seaside</i>

8. CONCLUSION

The aim of this paper is to present a two-tier annotation model where the first tier corresponds to object level and the second tier to scene level annotation. In the first tier, images are annotated with labels of objects present in them, using multi-label classification methods on low level features extracted from images. In the second tier the fuzzy knowledge-representation scheme based on the Fuzzy Petri Net (KRFPN) is incorporated. The places of the KRFPN are used to represent the concepts related to scene and object classes. The transitions of the KRFPN are used to represent the relationships between concepts. The KRFPN scheme is

compatible with various classification methods, and different types of classification methods were tested for image classification in the first tier. The inference engine of the KRFPN scheme is used for image annotation on the scene level as well as for inconsistency checking of object labels obtained at the first tier. To be more specific, the fuzzy-recognition algorithm provides inference about relationships between scene classes and their components (object classes) for scene classification, and the fuzzy-inheritance algorithm provides reasoning about co-occurrence relations between object classes for inconsistency checking. The algorithms of the inference engine of the KRFPN scheme are presented with finite-inference trees. Their complexity is $O(nm)$, where n is the number of places (concepts) and m is the number of transitions (relations). In practise usually the whole net is not used for inference, but only parts that can be reached from the initially marked places. If the token values represent the reliability of concepts, they decay quickly after passing through few levels of the tree.

The proposed model was experimentally tested for annotation of a dataset of outdoor images. Different subsets of features composed of dominant colours, image moments, and GIST descriptors as well as different classification methods (RAKEL, ML-kNN and Naïve Bayes) were tested in the first tier. Obtained results at the first tier are better than already published on the same set of images and so are more useful as inputs to the KRFPN scheme.

This research was focused on the domain of outdoor images for which the KRFPN scheme and inference engine provide knowledge for image annotation, but it can be further expanded with knowledge from different domains. Furthermore, the proposed two-tier annotation model is adaptive and each tier can be independently used and improved.

REFERENCES

- [Athanasiadis et al. 2009], Athanasiadis, T. et al. 2009. "Integrating Image Segmentation and Classification for Fuzzy Knowledge-based Multimedia", Proc. MMM2009, France, 2009.
- [Barnard et al. 2003, 14] Barnard K, Duygulu P, Forsyth D, Freitas N, Blei DM, Jordan MI. 2003. "Matching words and pictures," Journal of Machine Learning Research vol. 3: 1107–1135.
- [Binder et al. 2013.] Binder, A., W. Samek, K.-R. Müller, M. Kawanabe. 2013. "Enhanced representation and multi-task learning for image annotation." Computer Vision And Image Understanding 117, no. 5: 466-478, May 2013.
- [Carbonetto et al. 2004] Carbonetto, P., Freitas, N. de, Barnard, K., 2004. "A Statistical Model for General Contextual Object Recognition", Proc. ECCV 2004, Czech Republic, May 2004, pp. 350-362.
- [Chen et al. 1990] Chen, S. M., Ke, J. S., & Chang, J. F. Knowledge representation using fuzzy Petri nets. Knowledge and Data Engineering, IEEE Transactions on, 1990, vol. 2(3), 311-319.
- [Datta et al. 2008] Datta R, Joshi D, Li J. 2008. "Image Retrieval: Ideas, Influences, and Trends of the New Age", ACM Transactions on Computing Surveys, vol. 20, pp. 1-60, April 2008.
- [Deng et al. 2009] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. "ImageNet: A Large-Scale Hierarchical Image Database". IEEE Conference on Computer Vision and Pattern Recognition CVPR 2009. pp. 248-255.
- [Deruyver et al. 2009] Deruyver, A., Hodé, Y., Brun, L. "Image interpretation with a conceptual graph: Labeling over-segmented images and detection of unexpected objects", Artificial Intelligence, vol. 173(14), pp. 1245-1265, September 2009.

- [Duygulu et al. 2002] Duygulu P, Barnard K, de Freitas JFG, Forsyth DA. 2002. Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. *Proceedings of European Conference on Computer Vision*: 97-112.
- [Hare et al. 2006] Hare JS, Lewis PH, Enser PGB Sandom CJ. 2006 January 17-19. Mind the Gap: Another look at the problem of the semantic gap in image retrieval. *Multimedia Content Analysis, Management and Retrieval*, San Jose, California, USA.
- [Ivašić-Kos et al. 2010] Ivašić-Kos, M. Ribarić, S.; Ipšić, I. "Image Annotation using Fuzzy Knowledge Representation Scheme", *IEEE Proceedings of the 2010 International Conference of Soft Computing and Pattern Recognition*. Paris, France, 2010. 218-223
- [Li and Wang, 2008] J. Li and J. Z. Wang, "Real-Time Computerized Annotation of Pictures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, 2008, pp. 985-1002.
- [Liu et al. 2008] Jing Liu, Bin Wang, Hanqing Lu, Songde Ma, "A graph-based image annotation framework", *Pattern Recognition Letters*, Volume 29, Issue 4, March 2008, pp407-415
- [Madjarov et al., 2012] Madjarov, G., Kocev, D., Gjorgjevikj, D., Džeroski, S.: An extensive experimental comparison of methods for multi-label learning. *Pattern Recognition*, vol. 45, no. 9. (2012)
- [Mezaris et al. 2009] Mezaris V, Papadopoulos GT, Briassouli A, Kompatsiaris I, Strintzis MG. 2009. "Semantic Video Analysis and Understanding", chapter in "Encyclopedia of Information Science and Technology", Second Edition, Mehdi Khosrow-Pour, ebook.
- [Monay and Gatica-Perez, 2003] Monay F. and Gatica-Perez D., "On image auto-annotation with Latent Space Models", *Proc. ACM Multimedia*, Berkeley, CA, 2003, pp. 275–278.
- [Oliva and Torralba, 2001] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.
- [Pan et al., 2004] Pan, J. Y., Yang, H. J., Faloutsos, C., Duygulu, P. Gcap. "Graph-based automatic image captioning", *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04.* (pp. 146-146).
- [Ribarić and Pavešić, 2009] Ribarić, S., Pavešić, N., 2009. "Inference Procedures for Fuzzy Knowledge Representation Scheme", *Applied Artificial Intelligence*, vol. 23, January 2009, pp. 16-43.
- [Smeulders et al. 2000] Smeulders AWM, Worring M, Santini S, Gupta A, Jain R. 2000. Content-based image retrieval at the end of the early years. *IEEE Transaction on Pattern Analysis and Machine Intelligence*; 22 (12): 1349–1380.
- [Tsoumakas and Vlahavas, 2007] Tsoumakas, G., Vlahavas, I.: Random k-label sets: An ensemble method for multi-label classification. In: *Machine Learning: ECML*. Springer, pp. 406–417. (2007)
- [Tsoumakas and Katakis, 2007] Tsoumakas, G., Katakis, I.: Multi-Label Classification: An Overview. *International Journal of Data Warehousing & Mining*, vol. 3, no. 3, (2007).
- [Tousch et al., 2012] Tousch, A. M., Herbin, S., and Audibert, J. Y. (2012). Semantic hierarchies for image annotation: A survey. *Pattern Recognition*, 45(1), 333-345.
- [Zhang and Zhou, 2007] Zhang, M.L. and Zhou, Z.H. ML-KNN: A lazy learning approach to multi-label learning, *Pattern recognition*, Elsevier, 40(7), 2038-2048.2007.
- [Zhang et al. 2012] Zhang, D., Islam, M. M., and Lu, G. (2012). A review on automatic image annotation techniques. *Pattern Recognition*, 45(1), 346-362.
- [Zhang et al. 2014] Zhang, S, Tian, Q, Hua, G, Huang, Q, Gao, W, 2014. "ObjectPatchNet: Towards scalable and semantic image annotation and retrieval", *Computer Vision And Image Understanding*, 118, pp. 16-29