

Labelling Image Regions Using Wavelet Features and Spatial Prototypes^{*}

Carsten Saathoff, Marcin Grzegorzek, and Steffen Staab

ISWeb – Information Systems and Semantic Web Research Group
Institute for Computer Science, University of Koblenz – Landau
<http://isweb.uni-koblenz.de>
{saathoff,marcin,staab}@uni-koblenz.de

Abstract. In this paper we present an approach for image region classification that combines low-level processing with high-level scene understanding. For the low-level training, predefined image concepts are statistically modelled using wavelet features extracted directly from image pixels. For classification, features of a given test region compared with these statistical models provide probabilistic evaluations for all possible image concepts. Maximising these values themselves already leads to a classification result (label). However, in our paper they are used as an input for the high-level approach exploiting explicitly represented spatial arrangements of labels, so called spatial prototypes. We formalise the problem using Fuzzy Constraint Satisfaction Problems and Linear Programming. They provide a model with explicit knowledge that is suitable to aid the task of region labelling. Experiments performed for nearly 6000 test image regions show that combining low-level and high-level image analysis increases the labelling accuracy significantly.

1 Introduction

It has been shown in various studies [1] that semantic access to multimedia content is desired by most users, regardless of whether they are professional or private users. An important field of research is automatic annotation of images, and specifically the automatic labelling of image regions [2]. Region-level annotations provide more detailed information about the image contents, allow for answering complex queries, and can be used to improve global classification accuracy [3].

Since exploiting solely low-level features often leads to unsatisfactory results, research towards using contextual and spatial features is a prominent research topic recently. A comprehensive study of using context for improving object recognition was carried out in [4, 5], showing the importance of contextual information. In [6] a survey of using spatial features for image region labelling based on graph models was performed and showed that spatial features improve

^{*} The research activity leading to this work has been supported by the European Commission under the contract FP6-027026-K-SPACE.

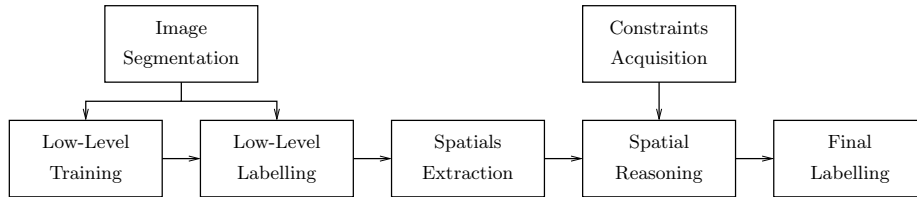


Fig. 1. Overall workflow of the approach for image region labelling.

the labelling accuracy. In [7] an approach based on explicitly defined spatial constraints was introduced that employed genetic algorithms to compute a final labelling. We have also published results on exploiting explicitly represented spatial constraints for improving image labelling accuracy in [8].

The experiments in [8] indicate that our approach based on explicit representation of spatial context requires only a low amount of labelled examples for acquiring the explicit model. In this paper we conduct a new study focusing on the performance of our approach with different training set sizes. However, we combined the spatial reasoning part with a new low-level classification technique based on wavelet features, and provide a new formalisation of spatial constraints using binary integer programs. As we will show, the combination provides much better labelling accuracy with only few training examples.

The overall workflow of our method is depicted in Figure 1. The algorithm starts with the low-level training (Section 2.1). Here, all image concepts considered in our labelling task are modelled using feature vectors computed directly from image pixels. Instead of using MPEG-7 descriptors [9], we represent the image contents by wavelet features [10] and statistically model the concepts (e. g., sky, road, building, etc.) by density functions [11]. Subsequently, the low-level labelling is performed (Section 2.2). Both the training and labelling take advantage of automatic image segmentation algorithms. The low-level labelling results are used for further high-level processing, namely the extraction of spatial relations (Section 3) and spatial reasoning (Section 4). Here, we formalise the problem using Fuzzy Constraint Satisfaction Problems and Linear Programming. They provide a model with explicit knowledge that is suitable to aid the task of region labelling. Finally, the image region labels are provided by our algorithm. Results of experiments performed for nearly 6000 test image regions show that using the combination of low-level and high-level image analysis increases the labelling accuracy significantly (Section 5). This leads to some interesting conclusions presented in Section 6.

2 Content-Based Image Region Classification

In this section the low-level algorithm for content-based image region classification (labelling) is described, whereas the set of image concepts $\Omega = \{\Omega_1, \Omega_2, \dots,$

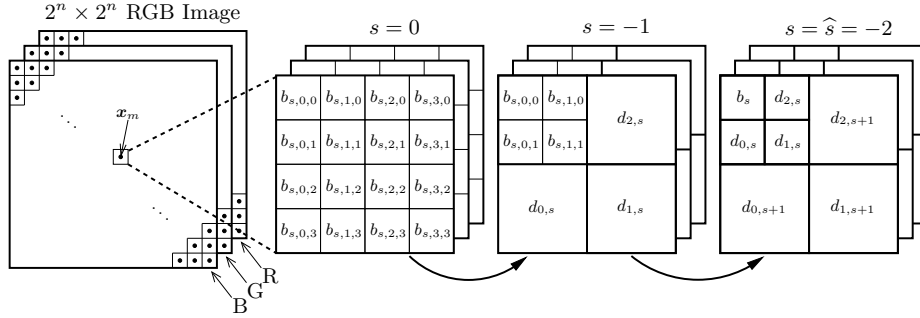


Fig. 2. Wavelet decomposition for a local neighbourhood of size 4×4 pixels done separately for the green, the red, and the blue channel. The final coefficients for the blue channel result from $b_{0,k,l}$ and have the following meaning: b_{-2} : low-pass horizontal and low-pass vertical, $d_{0,-2}$: low-pass horizontal and high-pass vertical, $d_{1,-2}$: high-pass horizontal and high-pass vertical, $d_{2,-2}$: high-pass horizontal and low-pass vertical.

$\Omega_\kappa, \dots, \Omega_{N_\Omega}$ is assumed to be a-priori known and constant. First, the statistical learning process is explained (Section 2.1). Second, the automatic labelling of image regions based on their contents is presented (Section 2.2).

2.1 Training of Image Concepts

In order to statistically train concepts (e.g., sky, road, building, etc.) based on image contents (pixel values), representative sets of example images for those concepts are required. The size of the training sets may vary, however, as you can see in Section 5, the performance of our image region classification algorithm depends on the number of training images. In order to reduce the amount of resources required to describe a large set of training images and to simplify the description, image contents are represented by feature vectors.

In our case RGB colour images are used for feature extraction. In order to calculate the vectors, a two-dimensional discrete signal decomposition with the wavelet transform [10] is applied for local neighbourhoods, whereas the Johnston 8-TAB wavelet is used as the basis function. A grid with size $\Delta r = 2^{\lceil \hat{s} \rceil}$, where \hat{s} is the minimum multiresolution scale parameter¹ s , is overlaid on the image [12]. Figure 2 depicts this procedure for the case of colour scenes divided into local neighbourhoods of size 4×4 pixels. Further, the results of the low-pass filtering for all three colour channels (b_s^R , b_s^G , and b_s^B) in Figure 2 represented as b_s are taken into consideration for feature computation. Although the wavelet analysis is done for local neighbourhoods (see Figure 2), a training image should rather be described by a single global feature vector independent of the location in the image. For this reason the results of the local wavelet analysis are put together and their mean values are used for image description. Finally, each

¹ Further decomposition of the signal with the wavelet transform is not possible.

training image $\mathbf{f}_{\kappa,i}$ obtains a global four-dimensional feature vector

$$\mathbf{c}_{\kappa,i} = (c_{\kappa,i,1}, c_{\kappa,i,2}, c_{\kappa,i,3}, c_{\kappa,i,4})^T . \quad (1)$$

The first component $c_{\kappa,i,1}$ of this feature vector is simple a mean pixel value in the image $\mathbf{f}_{\kappa,i}$

$$c_{\kappa,i,1} = \frac{1}{3 \cdot N_{\kappa,i}} \sum_{n=1}^{N_{\kappa,i}} (f_{\kappa,i,n}^R + f_{\kappa,i,n}^G + f_{\kappa,i,n}^B) , \quad (2)$$

where $N_{\kappa,i}$ is the number of all pixels representing the concept Ω_{κ} in the training image $\mathbf{f}_{\kappa,i}$. The remaining three components of the global feature vector $\mathbf{c}_{\kappa,i}$ (1) result from the low-level wavelet coefficients b_s^R , b_s^G , and b_s^B for the red, green, and blue channel respectively. They are computed as simple mean values of those coefficients for all local neighbourhoods defined according to Figure 2

$$c_{\kappa,i,2} = \frac{1}{M_{\kappa,i}} \sum_{n=1}^{M_{\kappa,i}} b_{s,n}^R , \quad (3)$$

$$c_{\kappa,i,3} = \frac{1}{M_{\kappa,i}} \sum_{n=1}^{M_{\kappa,i}} b_{s,n}^G , \quad (4)$$

and

$$c_{\kappa,i,4} = \frac{1}{M_{\kappa,i}} \sum_{n=1}^{M_{\kappa,i}} b_{s,n}^B , \quad (5)$$

where $M_{\kappa,i}$ is the number of all local neighbourhoods defined as in Figure 2 representing the concept Ω_{κ} in the training image $\mathbf{f}_{\kappa,i}$.

Since the number T_{κ} of training images $\mathbf{f}_{\kappa,i}$ for each concept Ω_{κ} is usually quite high², statistical modelling can be applied for training. It has been observed that the values of the feature vector components $c_{\kappa,i,n=1,\dots,4}$ behave regularly and can perfectly be modelled by normal density functions [13]. In order to do so, the mean values $\mu_{\kappa,n=1,\dots,4}$ and the standard deviations $\sigma_{\kappa,n=1,\dots,4}$ for the feature vector components $c_{\kappa,i,n=1,\dots,4}$ are computed in accordance to the well-known formulas

$$\mu_{\kappa,n} = \frac{1}{T_{\kappa}} \sum_{i=1}^{T_{\kappa}} c_{\kappa,i,n} , \quad (6)$$

and

$$\sigma_{\kappa,n}^2 = \frac{1}{T_{\kappa}} \sum_{i=1}^{T_{\kappa}} (c_{\kappa,i,n} - \mu_{\kappa,n})^2 . \quad (7)$$

Therefore, all concepts Ω_{κ} considered in the image region classification task are represented by a mean value vector

$$\boldsymbol{\mu}_{\kappa} = (\mu_{\kappa,1}, \mu_{\kappa,2}, \mu_{\kappa,3}, \mu_{\kappa,4})^T , \quad (8)$$

² In our experiments presented in Section 5 it varies from 50 to 400 in 7 steps.

and a standard deviation vector

$$\boldsymbol{\sigma}_\kappa = (\sigma_{\kappa,1}, \sigma_{\kappa,2}, \sigma_{\kappa,3}, \sigma_{\kappa,4})^\top \quad (9)$$

after the training phase.

2.2 Labelling of Image Regions

In order to classify image regions, first, a test image \mathbf{f} is automatically segmented into test regions \mathbf{f}_r . Then, each region found in the image \mathbf{f}_r is described by a four-dimensional feature vector

$$\mathbf{c}_r = (c_{r,1}, c_{r,2}, c_{r,3}, c_{r,4})^\top \quad (10)$$

This global feature vector is computed in exactly the same way as in the training phase (2, 3, 4, 5). The first component $c_{r,1}$ is a mean pixel value in the test region, while the remaining components $c_{r,n=2,\dots,4}$ result from the wavelet analysis performed separately for the red, green, and blue channel of the image (see Figure 2). Now, for all possible concepts $\Omega = \{\Omega_1, \Omega_2, \dots, \Omega_\kappa, \dots, \Omega_{N_\Omega}\}$ trained as shown in Section 2.1, the comparison with the test region is performed. For this, density values $p_{\kappa,r,n=1,\dots,4}$ for all feature vector elements $c_{r,n=1,\dots,4}$ are computed using the trained mean (8) and standard deviation vectors (9) according to the definition of the Gaussian density function [11]

$$p_{\kappa,r,n} = p(c_{r,n} | \mu_{\kappa,n}, \sigma_{\kappa,n}) = \frac{1}{\sigma_{\kappa,n} \sqrt{2\pi}} \exp\left(\frac{(c_{r,n} - \mu_{\kappa,n})^2}{-2\sigma_{\kappa,n}^2}\right) \quad (11)$$

Assuming the statistical independency between the feature vector elements, the final evaluation of the test region represented by \mathbf{c}_r and a hypothesis concept Ω_κ is computed with

$$p_{\kappa,r} = p(\mathbf{c}_r | \boldsymbol{\mu}_\kappa, \boldsymbol{\sigma}_\kappa) = \prod_{n=1}^4 p(c_{r,n} | \mu_{\kappa,n}, \sigma_{\kappa,n}) \quad (12)$$

Finally, the classification result $\hat{\Omega}_\kappa$ (region label) is found by maximisation of the density value (12) over all possible concepts represented by their index κ

$$\hat{\kappa} = \underset{\kappa}{\operatorname{argmax}} p_{\kappa,r} = \underset{\kappa}{\operatorname{argmax}} p(\mathbf{c}_r | \boldsymbol{\mu}_\kappa, \boldsymbol{\sigma}_\kappa) \quad (13)$$

The correspondence between the training results and the image region to be classified is presented in Figure 3.

3 Spatial relations extraction

Within our region labelling procedure we consider four relative and two absolute spatial relations to model the spatial arrangements of the regions within an

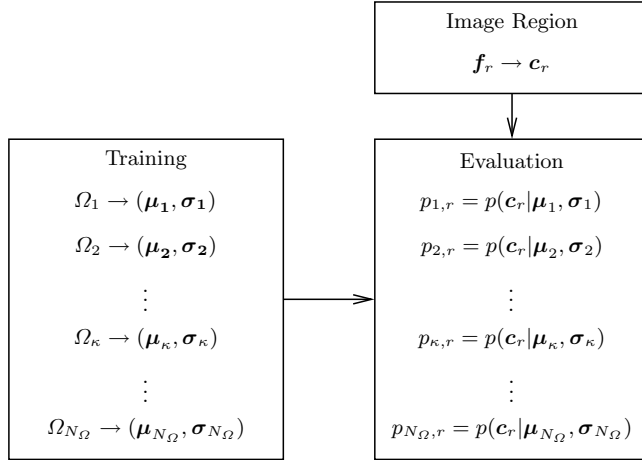


Fig. 3. The results of the training phase in form of mean vectors μ_κ and standard deviation vectors σ_κ for all concepts Ω_κ are compared with the image region to be classified f_r represented by c_r . As evaluation of this comparison density values $p_{\kappa,r}$ for all pairs “image region – hypothesis concept” are achieved.

image. The relative spatial relations are *above-of*, *below-of*, *left-of*, *right-of*, and the absolute spatial relations are *above-all* and *below-all*.

The directional relations are computed based on the centres of the minimal bounding box containing a region. We have illustrated the definition of the directional relations in Fig. 4a. Based on the angle α we determine the relation between two regions.

Computing whether a region is *above-all* or *below-all* is done with three different approaches. First of all, we use the centre of the bounding box and check whether it is above (below) a certain threshold. Second, we take the region with the highest (lowest) bounding box centre. Finally, we also check for regions that “touch” *either* the top *or* the bottom edge of the image. If a region touches both the top and the bottom edge of the image, it is assigned neither of the two absolute regions, in order to not produce any contradictory constraints.

4 Spatial Reasoning Based on Constraints

The goal of the spatial reasoning step is to exploit background knowledge about the typical spatial arrangements of objects in images in order to improve the labelling accuracy compared to pure local, low-level feature-based approaches. As we will discuss in the following, the spatial background knowledge is automatically extracted from labelled examples, which we call the spatial prototypes. The knowledge consists of spatial constraint templates, which are explicitly represented spatial arrangements of concepts, possibly associated with a degree of

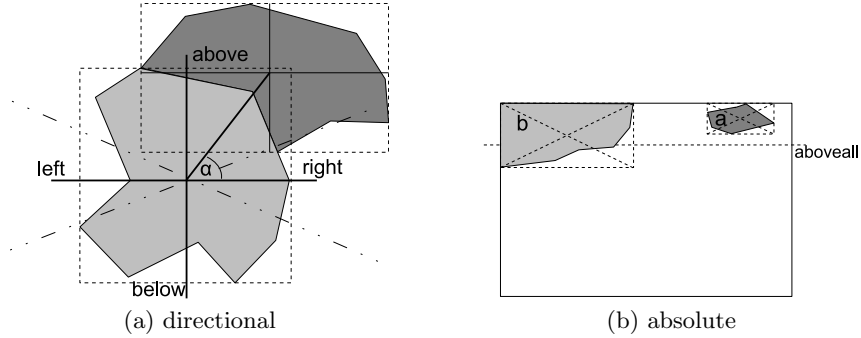


Fig. 4. Definition of the a) directional and b) absolute spatial relations.

confidence. We provide two formalisations of the problem, one based on Fuzzy Constraint Satisfaction Problems, which was already discussed in [8], and a new formalisation based on Linear Programming.

Our following discussions will be based on some fundamental formal definitions. Let L be the set of labels, and T the set of supported spatial relation types. An image is a tuple $I = \{S, R\}$, and S is the set of regions created by the segmentation. Each region $s_i \in S$ is associated with a membership function $\theta_i : L \rightarrow [0, 1]$ with $\theta_i = \{(l_k, p_{k,i})\}$, which associates each label with the probability provided by the low-level classification. Further, let $R = \{r_1, \dots, r_k\}$ be the set of extracted spatial relations. An absolute spatial relation r is a region itself, i.e. $r \in S$, while relative spatial relations are pairs of regions, i.e. $r \in S^2$. Each spatial relation is associated with a type $t \in T$.

We will first discuss the acquisition of constraint templates from a set of spatial prototypes, and then describe the formalisation of the problem both using Fuzzy Constraint Satisfaction Problems and Binary Integer Programs.

4.1 Constraint Acquisition

Spatial constraint templates constitute the background knowledge in our approach. Manually defining these templates is a tedious task, specifically if the number of supported concepts and spatial relations becomes larger. We derive these templates from spatial prototypes, which are manually labelled images. We mine the prototypes using support and confidence as selection criteria, and come up with a set of templates representing typical spatial arrangements.

Let the set of prototypes be a set of images $P = \{I_1, \dots, I_q\}$. For each region s_i of the images we assume that only one label exists, i.e. there is only one $l_i \in L$, such that $\theta_i(l_i) = 1$, and $\forall l_j \in L, l_j \neq l_i : \theta_i(l_j) = 0$. In the following we will say that l_i is the label associated with region s_i . We want to acquire one template for each spatial relation type $t \in T$. We denote the template as \mathcal{T}_t , and interpret each template as a fuzzy relation on the set of labels, i.e. $\mathcal{T}_t : L^n \rightarrow [0, 1]$. n equals 1 in the case of an absolute relation, and 2 for relative spatial relations.

In order to determine the membership degrees for each tuple of labels, we use the information present in our set of prototypes. For each label l we have to determine in what spatial relation to other labels it might be found. Therefore, for each spatial relation type $t \in T$, we consider the set of relations

$$R_{t \downarrow l} := \{r | \exists I = \{S, R\} \in P : \exists s_i \in S : \theta_i(l) = 1 \wedge r \in R_t\}, \quad (14)$$

where $R_t \subseteq R$ denotes the set of relations with type t .

This set only contains relations from images depicting l , i.e. we limit both support and confidence to the *context* of label l . We then define $R_t^{l, l'} \subseteq R_{t \downarrow l}$ to be the set of relations between segments s, s' depicting l and l' , respectively. Finally, we write $R_{t \downarrow l}^{*, l'} \subseteq R_{t \downarrow l}$ to denote all relations between an arbitrary region and a region depicting l' . The confidence of a label arrangement is then defined as

$$\gamma_t(l, l') = \frac{|R_t^{l, l'}|}{|R_{t \downarrow l}^{*, l'}|}, \quad (15)$$

and the support as

$$\sigma_t(l, l') = \frac{|R_t^{l, l'}|}{|R_{t \downarrow l}|}. \quad (16)$$

This definition can easily be modified for unary relations. In that case we are interested in the set R_t^l , which contains all absolute spatial relations on some region s that depicts label l . Further, we denote with $|l|$ the number of regions associated with label l . Support and confidence for absolute spatial relations can then be defined as

$$\gamma_t(l) = \frac{|R_t^l|}{|l|}, \quad (17)$$

and

$$\sigma_t(l) = \frac{|R_t^l|}{|R_{t \downarrow l}|}. \quad (18)$$

Finally, we have to define the template \mathcal{T}_t for the spatial relation type t . We consider only two degrees of confidence for templates. We define $\mathcal{T}_t(l, l') = 1$ if we accept the pair, or $\mathcal{T}_t(l, l') = 0$ if we reject it. In order to determine whether we want to accept or reject a certain pair of labels for the relation t , we use two thresholds th_σ and th_γ , and define a template as

$$\mathcal{T}_t(l, l') = \begin{cases} 1 & \text{if } \sigma_t(l, l') > th_\sigma \text{ and } \gamma_t(l, l') > th_\gamma \\ 0 & \text{else} \end{cases} \quad (19)$$

For absolute spatial relations, the template is defined accordingly.

4.2 Spatial Reasoning with Fuzzy Constraint Satisfaction Problems

We transform the segmented and labelled image along with the spatial prototypes into a *Fuzzy Constraint Satisfaction Problem*. In the following, we will first introduce Fuzzy Constraint Satisfaction Problems as a formal model and then

discuss the transformation. Our definition is based on [14] extended with *fuzzy domains*.

A Fuzzy Constraint Satisfaction Problem consists of an ordered set of fuzzy variables $V = \{v_1, \dots, v_k\}$, each associated with the crisp domain $L = \{l_1, \dots, l_n\}$ and the membership function $\mu_i : L \rightarrow [0, 1]$. The value $\mu_i(l), l \in L$ is called the degree of satisfaction of the variable for the assignment $v_i = l$. Further, we define a set of fuzzy constraints $C = \{c_1, \dots, c_m\}$. Each constraint c_j is defined on a set of variables $v_1, \dots, v_q \in V$, and we interpret a constraint as a fuzzy relation $c_j : L^q \rightarrow [0, 1]$. The value $c(l_1, \dots, l_q)$, with $v_i = l_i$ is called the degree of satisfaction of the variable assignment l_1, \dots, l_q for the constraint c . In case that $c(l_1, \dots, l_q) = 1$, we say that the constraint is fully satisfied, and if $c(l_1, \dots, l_q) = 0$ we say it is fully violated. The purpose of fuzzy constraint reasoning is to obtain a variable assignment that is optimal with respect to the degrees of satisfaction of the variables and constraints. The quality of a solution is measured using a global evaluation function, which is called the *joint degree of satisfaction*.

We first define the joint degree of satisfaction of a variable, which determines the local satisfaction degree of the problem. Let $P = \{l_1, \dots, l_k\}, k \leq |V|$ be a partial solution of the problem, with $v_i = l_i$. Let $C_i^+ \subseteq C$ be the set of the fully instantiated constraints on v_i . Further, let \hat{c} stand for the degree of satisfaction of c given the current partial solution. Finally, let $C_i^- \subseteq C$ be the set of partially instantiated constraints on v_i . We then define the joint degree of satisfaction as $\text{dos}(v_i) := \frac{1}{\omega+1} \left(\frac{1}{|C_i^+|+|C_i^-|} (\sum_{c \in C_i^+} \hat{c} + |C_i^-|) + \omega \mu_i(l_i) \right)$, in which ω is a weight used to control the influence of the degree of satisfaction of the variable assignment on the joint degree. In this definition we overestimate the degree of satisfaction of partially instantiated constraints to 1.

We now define the joint degree of satisfaction for a complete Fuzzy Constraint Satisfaction Problem. Let $J := \{\text{dos}(v_{i_1}), \dots, \text{dos}(v_{i_n})\}$ be an ordered multiset of joint degrees of satisfaction for each variable in V , with $\forall v_{i_k}, v_{i_l} \in V, k < l : \text{dos}(v_{i_k}) \leq \text{dos}(v_{i_l})$. The joint degree of satisfaction of a variable that is not yet assigned a value is overestimated to 1. We can now define a lexicographic order $>_L$ on the multisets. Let $J = \{\gamma_1, \dots, \gamma_k\}, J' = \{\delta_1, \dots, \delta_k\}$ be multisets. Then $J >_L J'$, iff $\exists i \leq k : \forall j < i : \gamma_j = \delta_j$ and $\gamma_i > \delta_i$. If we have two (partial) solutions P, Q to a Fuzzy Constraint Satisfaction Problem with according joint degree of satisfactions J_P, J_Q , solution P is better than Q , iff $J_P >_L J_Q$.

Now, we can transform an initially labelled image into a Fuzzy Constraint Satisfaction Problem using the following algorithm.

1. For each region $s_i \in S$ create a variable v_i on L with $\mu_i := \theta_i$.
2. For each region $s_i \in S$ and for each spatial relation r of type *type* defined on s_i and further segments s_1, \dots, s_k create a constraint c on v_i, v_1, \dots, v_k with $c := p$, where $p \in P$ is a spatial prototype of type *type*.

The resulting Fuzzy Constraint Satisfaction Problem can efficiently be solved using algorithms like branch and bound, as was also discussed in [14].

4.3 Spatial Reasoning with Linear Programming

We will show in the following how to formalise image labelling with spatial constraints as a linear program. We will first introduce Linear Programming as a formal model, and then discuss how to represent the image labelling problem as a linear program.

A linear program has the standard form

$$\begin{aligned} & \text{minimize} && Z = \mathbf{c}^T \mathbf{x} \\ & \text{subject to} && \mathbf{A}\mathbf{x} = \mathbf{b} \\ & && \mathbf{x} \geq 0 \end{aligned} \tag{20}$$

where \mathbf{c}^T is a row vector of so-called objective coefficients, \mathbf{x} is a vector of control variables, \mathbf{A} is a matrix of constraint coefficients, and \mathbf{b} a vector of row bounds. Efficient solving techniques exist for linear programs, e.g. the *Simplex Method*. Goal of the solving process is to find a set of assignments to the variables in \mathbf{x} with a minimal evaluation score Z that satisfy all the constraints. In general, most non-standard representations of a linear program can be transformed into this standard representation. In this paper we will consider binary integer programs of the form

$$\begin{aligned} & \text{maximize} && Z = \mathbf{c}^T \mathbf{x} \\ & \text{subject to} && \mathbf{A}\mathbf{x} = \mathbf{b} \\ & && \mathbf{x} \in \{0, 1\} \end{aligned} \tag{21}$$

where $\mathbf{x} \in \{0, 1\}$ means that each element of \mathbf{x} can either be 0 or 1.

In order to represent the image labelling problem as a linear program, we create a set of linear constraints from each spatial relation in the image, and determine the objective coefficients based on the hypotheses sets and the constraint templates.

Given an image $I = \{S, R\}$, let $O_i \subseteq R$ be the set of outgoing relations for region $s_i \in S$, i.e. $O_i = \{r \in R \mid \exists s \in S, s \neq s_i : r = (s_i, s)\}$. Accordingly, $E_i \subseteq R$ is the set of incoming spatial relations for a region s_i , i.e. $E_i = \{r \in R \mid \exists s \in S, s \neq s_i : r = (s, s_i)\}$. For each spatial relation we need to create a set of control variables according to the following scheme. Let $r = (s_i, s_j)$ be the relation. Then, for each possible pair of label assignments to the regions, we create a variable c_{itj}^{ko} , representing the possible assignment of l_k to s_i and l_o to s_j with respect to the relation r with type $t \in T$. Each c_{itj}^{ko} is an integer variable and $c_{itj}^{ko} = 1$ represents the assignments $s_i = l_k$ and $s_j = l_o$, while $c_{itj}^{ko} = 0$ means that these assignments are not made. Since every such variable represents exactly one assignment of labels to the involved regions, and only one label might be assigned to a region in the final solution, we have to add this restriction as linear constraints. The constraints are formalised as

$$\forall r \in R : r = (s_i, s_j) \in R \rightarrow \sum_{l_k \in L} \sum_{l_o \in L} c_{itj}^{ko} = 1. \tag{22}$$

These constraints assure that there is only one pair of labels assigned to a pair of regions per spatial relation, but it does not guarantee, that for all relations

involving a specific region the same label is chosen for the region. In effect, this means that there could be two variables c_{itj}^{ko} and $c_{it'j'}^{k'o'}$ both being set to 1, which would result in both k and k' assigned to s_i . Since our solution requires that there is only one label assigned to a region, we have to add constraints that “link” the variables accordingly.

We basically require that either all variables assign a label l_k to a region s_i , or none. This can be accomplished by linking pairs of relations. We first have to link the outgoing relations O_i , then the incoming ones E_i , and finally link one of the outgoing relations to one of the incoming ones. This system of linear constraints will ensure that only one label is assigned to the region in the final solution.

We will start by defining the constraints for the outgoing relations. We take one arbitrary relation $r_O \in O_i$ and then create constraints for all $r \in O_i \setminus r_O$. Let $r_O = (s_i, s_j)$ with type t_O , and $r = (s_i, s_{j'})$ with type t be the two relations to be linked. Then, the constraints are

$$\forall l_k \in L : \sum_{l_o \in L} c_{it_Oj}^{ko} - \sum_{l'_o \in L} c_{it'j'}^{k'o'} = 0. \quad (23)$$

The first sum can either take the value 0 if l_k is not assigned to s_i by the relation r , or one if it is assigned. Equation (22) ensures that only one of the c_{itj}^{ko} is set to 1. Basically, the same applies for the second sum. Since both are subtracted and the whole expression has to evaluate to 0, either both equal 1 or both equal 0 and subsequently, if one of the relations assigns l_k to s_i , the others have to do the same. We can define the constraints for the incoming relations accordingly. Let $r_E \in E_i$, $r_E = (s_j, s_i)$ with type t_E be an arbitrarily chosen incoming relation. For each $r \in E_i \setminus r_E$ with $r = (s_{j'}, s_i)$ and type t create constraints

$$\forall l_k \in L : \sum_{l_o \in L} c_{jt_Ei}^{ok} - \sum_{l'_o \in L} c_{j't_i}^{o'k} = 0. \quad (24)$$

Finally we have to link the outgoing to the incoming relations. Since the same label assignment is already enforced within those two types of relations, we only have to link r_O and r_E , using the following set of constraints:

$$\forall l_k \in L : \sum_{l_o \in L} c_{it_Oj}^{ko} - \sum_{l'_o \in L} c_{j't_Ei}^{o'k} = 0 \quad (25)$$

Absolute relations are formalised accordingly. Let $A_i \subseteq R$ be the set of absolute relations defined on s_i . For each $r \in A$ of type t we define a set of control variables $c_{it}^k, \forall l_k \in L$. The constraint enforcing only one label assignment is defined as

$$\sum_{l_k \in L} c_{it}^k = 1, \quad (26)$$

and we link it to all remaining absolute relations r' on the region s_i with

$$\forall l_k \in L : c_{it}^k - c_{it'}^k = 0. \quad (27)$$

Further we have to link the absolute relation to the relative relations. Again, linking one of the relations is sufficient, and therefore we choose either the relation r_O or r_E and an arbitrary absolute relation a :

$$\forall l_k \in L : c_{it_a}^k - \sum_{l_o \in L} c_{it_oj}^{k_o} = 1 \quad (28)$$

Eventually, let t_r and t_a refer to the type of the relative relation r and the absolute relation a , respectively, then the objective function is defined as

$$\sum_{r=(s_i,s_j)} \sum_{l_k \in L} \sum_{l_o \in L} \min(\theta_i(l_k), \theta_j(l_o)) * \mathcal{I}_{t_r}(l_k, l_o) * c_{it_rj}^{k_o} + \sum_{a=s_i} \sum_{l_k \in L} \theta_i(l_k) * \mathcal{I}_{t_a}(l_k) * c_{it_a}^k. \quad (29)$$

For label pairs that violate our background knowledge, the product in the sum will evaluate to 0, while for pairs satisfying our background knowledge, we take the minimum degree of confidence from the hypotheses sets. Therefore we reward label assignments that satisfy the background knowledge and that involve labels with a high confidence score provided during the classification step.

5 Experiments and Results

Since we are interested in providing good labelling performance with only few training examples, we conducted an evaluation of our approach with varying training set sizes on a dataset of over 900 images with region-level annotations. The dataset was divided into a test and a number of training sets of different sizes. We then carried out a number of experiments on the largest training set to determine a set of spatial relation types and acquisition parameters used for the final evaluation. The differences achieved with variations in the parameters or spatial relations are only minor, so that we do not expect a large impact on the final results if the parameters are changed. We will continue by describing the image database first, and then give the results of our experiments both for the pure low-level approach, and the combination with the spatial reasoning. In the end we will discuss the lessons learned.

5.1 Image Database

The dataset consists of 923 images depicting outdoor scenes ranging from beach images over mountain views and cityscape images. An overview is provided in Figure 5. We chose a set of 10 labels that were prominent in the images and where a correct segmentation was feasible. The labels are *building*, *foliage*, *mountain*, *person*, *road*, *sailing-boat*, *sand*, *sea*, *sky*, *snow*. Additionally, we used the label *unknown* for regions where we could not decide on a definite label.

For preparing the groundtruth, all images were segmented by an automatic segmentation algorithm that was available from a research project [15], and the resulting regions were labelled using exactly one of the labels. We always used the dominant concept depicted in a region, i.e. the concept covering the largest

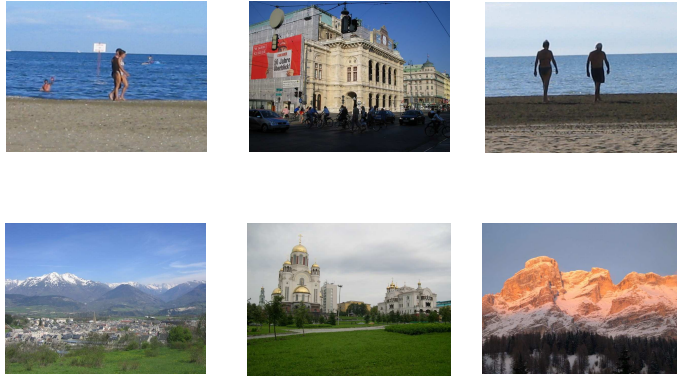


Fig. 5. Overview of the image database used for the experiments.

part of the region, and labelled the region with the according label. Regions without a dominant concept, or regions depicting an unsupported concept, were assigned the *unknown* label.

In total the dataset contained 6258 regions, of which 568 were labelled with the unknown label. This resulted in a dataset of 5690 regions labelled with a supported concept. 3312 were used for evaluation, and in the largest data set we used 2778 for training.

5.2 Experimental Results

The goal of our experiments was to determine the performance of our approach with varying training-set sizes. For that a series of experiments was performed in order to determine the influence of different parameters and features on the overall performance using the largest training set. We fixed those parameters and then performed the experiments using different training set sizes. The final setup consisted of the spatial relations discussed in Section 3, and using the thresholds $\sigma = 0.001$ and $\gamma = 0.2$ for both relative and absolute spatial relations. As one can see from the final thresholds, filtering on support is not feasible, but confidence provides a good quality estimation for spatial constraint templates.

For each approach, i.e. pure low-level classification, spatial reasoning using Fuzzy Constraint Satisfaction Problems, and spatial reasoning using Binary Integer Programs, we measured *precision* (p), *recall* (r) and the *classification rate* (c). Further we computed the *F-Measure* (f). In Table 1 the average for each of these measures is given.

One can clearly see, that both spatial reasoning approaches improve the pure low-level classification results. This observation is fully consistent with earlier findings [8], and also other studies that were performed [6, 4, 7]. It is also obvious that the binary linear programming approach outperforms the fuzzy constraint satisfaction approach. This is probably due to the different objective functions used. The Lexicographic order is still a rather coarse estimation of the overall

set size	Low-Level				FCSP				BIP			
	p	r	f	c	p	r	f	c	p	r	f	c
50	.63	.65	.57	.60	.65	.64	.62	.67	.77	.75	.73	.75
100	.70	.67	.65	.69	.67	.67	.65	.70	.78	.77	.75	.80
150	.67	.63	.61	.66	.66	.64	.63	.69	.74	.71	.70	.75
200	.69	.65	.63	.67	.67	.64	.64	.68	.80	.75	.76	.80
250	.69	.64	.60	.66	.69	.66	.65	.70	.78	.73	.72	.77
300	.68	.63	.61	.66	.68	.65	.64	.69	.82	.77	.78	.82
350	.63	.68	.61	.66	.70	.66	.66	.70	.80	.75	.76	.80
400	.68	.63	.61	.66	.69	.66	.65	.70	.80	.75	.75	.79

Table 1. Overall results for the three approaches.

labelling accuracy, while the objective function for the binary integer program well integrates the two important properties, i.e. satisfying spatial constraints and a high probability score from the classifier.

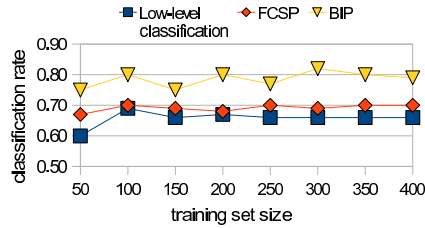


Fig. 6. Development of the classification rate with different training-set sizes.

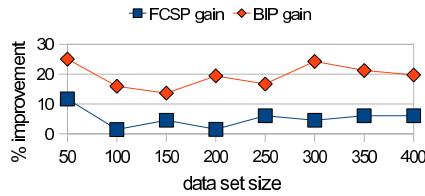


Fig. 7. The improvement over the low-level classification achieved with increasing number of training examples.

In Figure 6 we have visualised the performance development with increasing data set sizes. Against our initial assumptions, we do not see a steady increase in performance, but already with 100 training examples nearly the best performance, which stays pretty stable for the rest of the experiments. One can also see that incorporating spatial context provides the largest performance increase

for the smallest training set, which is an indicator that a good set of constraint templates is already acquired with only 50 prototype images. We have summarised the classification rate improvement achieved with the spatial reasoning in Figure 7.

5.3 Lessons Learned

The improvement is most significant for the training set with only 50 images. For this set, the low-level classification rate is worst, but the large improvement indicates that the background knowledge already provides a good model after acquisition from only 50 examples. The best overall classification rate is achieved with the binary integer programming approach on the data set with 300 training images. However, the classification rate with 100 training examples is nearly the same, which indicates that 100 training examples are a good size for training a well performing classifier.

In general, the experiments show that the combination of the statistical training of low-level classifiers with an explicit spatial context model based on binary integer programming provides a good foundation for labelling of image regions with only few training examples.

Further, our experiments also revealed that solving this kind of problem is much more efficient using binary integer programs. In average, the binary integer programming approach requires 1.1 seconds for one image, with a maximum value of 41 seconds and a minimum of only 6 ms. The fuzzy constraint reasoning, however, takes several hours for a few images, while in average it takes around 40 seconds. So, for the FCSP the runtime is much less predictable and also much higher in the average case. So, especially for real applications the binary integer programming approach clearly seems preferable.

6 Conclusions

In this paper we have introduced a novel combination of a statistical method for training and recognising concepts in image regions, integrated with an explicit model of spatial context. We have proposed two ways of formalising explicit knowledge about spatial context, one based on Fuzzy Constraint Satisfaction Problems, that was already presented in [8], and a new one based on Binary Integer Programming.

Our results show that the combination of both approaches results in a good classification rate compared to results in the literature. We have further evaluated how the classification rate develops with an increasing number of training examples. Surprisingly, nearly the best performance was already reached with only 100 training images. But also with 50 training images (approx. 344 regions) the combined approach provided a reasonable classification rate.

We are going to continue this work in the future introducing some improvements. For instance, new low-level features combining the texture and shape information will be applied for image content description.

References

1. Hollink, L., Schreiber, T.A., Wielinga, B.J., Worring, M.: Classification of user image descriptions. *International Journal of Human-Computer Studies* **61**(5) (2004)
2. Barnard, K., Fan, Q., Swaminathan, R., Hoogs, A., Collins, R., Rondot, P., Kaufhold, J.: Evaluation of localized semantics: data, methodology, and experiments. *International Journal of Computer Vision* **77** (2008) 199–127
3. Fan, J., Gao, Y., Luo, H.: Multi-level annotation of natural scenes using dominant image components and semantic concepts. In: *Proc. of ACM Multimedia 2004*, New York, NY, USA, ACM (2004) 540–547
4. Torralba, A.: Contextual priming for object detection. *Int. J. Comput. Vision* **53**(2) (July 2003) 169–191
5. Grzegorzec, M., Izquierdo, E.: Statistical 3d object classification and localization with context modeling. In Domanski, M., Stasinski, R., Bartkowiak, M., eds.: *15th European Signal Processing Conference*, Poznan, Poland, PTETiS, Poznan (September 2007) 1585–1589
6. Yuan, J., Li, J., Zhang, B.: Exploiting spatial context constraints for automatic image region annotation. In: *Proc. of ACM Multimedia 2007*, New York, NY, USA, ACM (2007) 595–604
7. Panagi, P., Dasiopoulou, S., Papadopoulos, T.G., Kompatsiaris, Strintzis, M.G.: A genetic algorithm approach to ontology-driven semantic image analysis. In: *Proc. of VIE 2006*. (2006) 132–137
8. Saathoff, C., Staab, S.: Exploiting spatial context in image region labelling using fuzzy constraint reasoning. In: *WIAMIS: Ninth International Workshop on Image Analysis for Multimedia Interactive Services*. (2008)
9. Manjunath, B.S., Salembier, P., Sikora, T.: *Introduction to MPEG-7 - Multimedia Content Description Interface*. John Wiley & Sons Ltd, Chichester, UK (2002)
10. Mallat, S.: A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11**(7) (July 1989) 674–693
11. Webb, A.R.: *Statistical Pattern Recognition*. John Wiley & Sons Ltd, Chichester, UK (2002)
12. Grzegorzec, M., Reinhold, M., Niemann, H.: Feature extraction with wavelet transformation for statistical object recognition. In Kurzynski, M., Puchala, E., Wozniak, M., Zolnierek, A., eds.: *4th International Conference on Computer Recognition Systems*, Rydzyna, Poland, Springer-Verlag, Berlin, Heidelberg (May 2005) 161–168
13. Grzegorzec, M.: *Appearance-Based Statistical Object Recognition Including Color and Context Modeling*. Logos Verlag, Berlin, Germany (2007)
14. Ruttkay, Z.: Fuzzy constraint satisfaction. In: *Proc. of Fuzzy Systems 1994*. Volume 2. (1994) 1263–1268
15. Dasiopoulou, S., Heinecke, J., Saathoff, C., Strintzis, M.G.: Multimedia reasoning with natural language support. In: *Proc. of ICSC 2007*. (2007) 413–420