Semantic Adaptation of Neural Network Classifiers in Image Segmentation

Nikolaos Simou, Thanos Athanasiadis, Stefanos Kollias,
Giorgos Stamou, and Andreas Stafylopatis

Department of Electrical and Computer Engineering,
National Technical University of Athens,
Zographou 15780, Greece
{nsimou,thanos}@image.ntua.gr, stefanos@cs.ntua.gr

Abstract. Semantic analysis of multimedia content is an on going research area that has gained a lot of attention over the last few years. Additionally, machine learning techniques are widely used for multimedia analysis with great success. This work presents a combined approach to semantic adaptation of neural network classifiers in multimedia framework. It is based on a fuzzy reasoning engine which is able to evaluate the outputs and the confidence levels of the neural network classifier, using a knowledge base. Improved image segmentation results are obtained, which are used for adaptation of the network classifier, further increasing its ability to provide accurate classification of the specific content.

1 Introduction

The usage of semantic analysis in multimedia applications is currently a field of extensive research [9] that also forms recent R&D activities of European IST projects, such as Acemedia, Muscle, K-Space, X-Media, Mesh. Moreover, machine learning techniques are also used in the field to handle specific aspects related to learning classification or adaptation. In this paper, we show that both technologies can be interweaved to provide improved performance segmentation of static or moving images.

In the following, we describe the overall architecture used for semantic adaptation of a neural network classifier in image or video segmentation. The architecture of the proposal is illustrated in Figure 1.

An image, or a video frame is initially processed by a segmentation algorithm [1] which partitions it in a number of regions, that may have a symbolic interpretation. Standard MPEG-7 low level visual features are extracted from these regions forming the input of the adaptable neural network classifier, which assigns a semantic label and a confidence value to each segment. The obtained classification results are then processed by the application of a semantic-based segmentation algorithm, which aims to refine the initial labels and the derived segmentation masks. Finally, neighboring regions that share common semantic labels and meet certain criteria are merged to form a more meaningful segmentation of the image.

V. Kůrková et al. (Eds.): ICANN 2008, Part I, LNCS 5163, pp. 907–916, 2008.

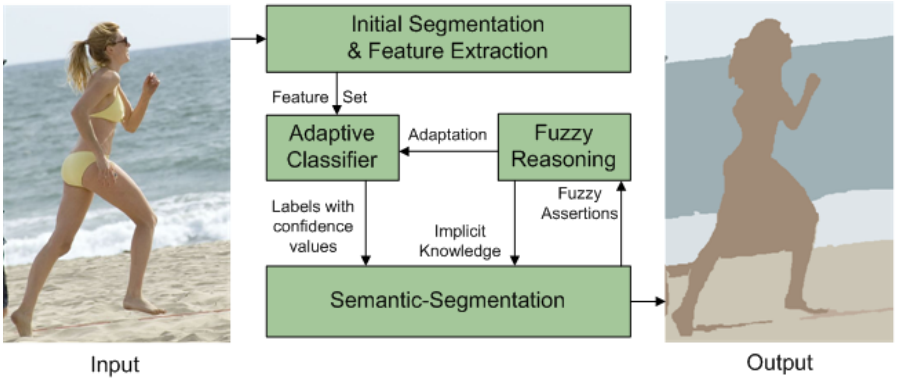

Fig. 1. The semantic adaptation architecture

In particular, the region-associated semantic labels and degrees of confidence are refined by the fuzzy reasoning engine FiRE[1]. FiRE is based on the expressive description logic (DL) f-$\mathcal{SHIN}$ [11]. The segments of the image are represented as DL-individuals, participating in the domain concepts to a given degree, and together with their spatial relations, they comprise the fuzzy assertion component (ABox) of the knowledge base. The terminology (TBox) is defined by using the domain concepts, declaring more general and complex concepts regarding both the segments and the image.

Using such a representation, implicit knowledge about segments can be extracted. This inferred knowledge either assigns them to higher concepts or corrects labels that have been mistakenly assigned by the classifier. These results constitute a source of information that can be used:

- to feed a semantic segmentation algorithm merging the updated segments and producing an improved segmentation mask
- as input to the adaptable neural network classifier

This classifier uses the semantically corrected results from reasoning, for adaptation purposes, so as to improve:

- its knowledge of the specific domain
- its performance over the next videos frames or images of similar content.

This reasoning adaptation cycle can be repeated more than once, depending on the image, or video frame, complexity.

The rest of the paper is organized as follows. In the next section the semantically adaptive neural network classifier is presented. Section 3 introduces the fuzzy knowledge base that was used for the refinement of segments and their confidence values. Section 4 presents the algorithm which performs the semantic image segmentation task. Finally, the last section presents some preliminary

[1] FiRE can be found at http://www.image.ece.ntua.gr/FiRE together with installation instructions and examples.

results of the proposed architecture. Furthermore, conclusions and suggested further work are provided in Section 6.

2 The Semantically Adaptable Classifier

The neural network classifier accepts an input vector $\overline{x}_i$ containing the features extracted from each region which determined by the initial segmentation phase, and categorizes it to one of, say, p available region classes ω_i.

The output vector $\overline{y}(\overline{x}_i)$ is

$$\overline{y}(\overline{x}_i) = \left[p^i_{\omega_1} p^i_{\omega_2} \dots p^i_{\omega_p}\right]^T \tag{1}$$

where $p^i_{\omega_j}$ denotes the probability that the ith region belongs to the jth class.

The neural network is initially trained to perform the classification task using a specific training set, say $S_b = \left\{ \left(\overline{x'}_1,\ \overline{d'}_1\right), \cdots, \left(\overline{x'}_{m_b},\ \overline{d'}_{m_b}\right) \right\}$, where vectors $\overline{x'}_i$ and $\overline{d'}_i$ with $i = 1, 2, \cdots, m_b$ denote the i input training vector and the corresponding desired output vector consisting of p elements. In the present case the input features are the low-level descriptors for every image segment. These are the MPEG-7: Scalable Color, Homogeneous Texture, Edge Histogram and Region Shape. The computed feature vector is employed by the neural network for the generation of the initial hypotheses regarding the segments semantic labels.

Then, the network classifier is applied to a new video frame or image. Whenever the network performance is estimated as non very accurate, or erroneous, a slightly different network weight set should be estimated. This can be established through a network adaptation procedure.

Let $\overline{w}_b$ include all weights of the network before adaptation, and $\overline{w}_a$ the new weight vector, which is obtained after adaptation is performed. To perform the adaptation, a training set S_c is formed including features of say m_c regions the semantic label of which has been refined or modified by the fuzzy reasoning engine; $S_c = \left\{ \left(\overline{x}_1,\ \overline{d}_1\right), \cdots, \left(\overline{x}_{m_c},\ \overline{d}_{m_c}\right) \right\}$ where $\overline{x}_i$ and $\overline{d}_i$ with $i = 1, 2, \cdots, m_c$ correspond to the i input and the desired output data to be used for adaptation. The adaptation algorithm that is activated, whenever such a need is detected, computes the new network weights $\overline{w}_a$, minimizing the following error criteria with respect to weights,

$$E_a = E_{c,a} + \eta E_{f,a}$$

$$E_{c,a} = \frac{1}{2} \sum_{i=1}^{m_c} \left\| \overline{z}_a(\overline{x}_i) - \overline{d}_i \right\|_2$$

$$E_{f,a} = \frac{1}{2} \sum_{i=1}^{m_b} \left\| \overline{z}_a(\overline{x'}_i) - \overline{d'}_i \right\|_2 \tag{2}$$

where $E_{c,a}$ is the error performed over training set S_c ("current" knowledge), $E_{f,a}$ the corresponding error over training set S_b ("former" knowledge); $\overline{z}_a(\overline{x}_i)$

and $\overline{z}_a(\overline{x}'_i)$ are the outputs of the adapted network, corresponding to the input vectors $\overline{x}_i$ and $\overline{x}'_i$ respectively, of the network consisting of weights $\overline{w}_a$. Similarly $\overline{z}_b(\overline{x}_i)$ would represent the output of the network, consisting of weights $\overline{w}_b$, when accepting vector $\overline{x}_i$ at its input. Parameter η is a weighting factor accounting for the significance of the current training set compared to the former one and $\|\cdot\|_2$ denotes the L_2-norm.

The goal of the training procedure is to minimize $E_{f,a}$ and estimate the new network weights $\overline{w}_a$. The adopted algorithm has been proposed by the authors in [5][6] leads and provides an analytical and tractable solution for estimating $\overline{w}_a$, through linearization of the non-linear activation function of the neurons.

Equation (2) indicates that the new network weights are estimated taking into account both the current and the previous network knowledge. To stress, however, the importance of current training data in (2), the first term is replaced by the constraint that the actual network outputs are equal to the desired ones. Assuming that weight adaptation refers to small increments that is

$$z_a(\overline{x}_i) = d_i, i = 1, \cdots, m_c, \forall \overline{x} \in S_c \tag{3}$$

Moreover, minimization of the second term of (2), which expresses the effect of the new network weights over the data set S_b, can be considered as minimization of the absolute difference of the error over the data in S_b with respect to the previous and the current network weights. This means that the weight increments are minimally modified, with respect to the following error criterion

$$E_S = \|E_{f,a} - E_{f,b}\|_2 \tag{4}$$

with $E_{f,b}$ defined similarly to $E_{f,a}$, with $\overline{z}_a$ replaced by $\overline{z}_b$ in (2).

It can be shown [8] that (4) takes the form of

$$E_S = \frac{1}{2}(\Delta\overline{w})^T \cdot K^T \cdot K \cdot \Delta\overline{w} \tag{5}$$

where the elements of matrix K are expressed in terms of the previous network weights w_b and the training data in S_b. The error function defined by (5) is convex since it is of squared form. The gradient projection method has been used to estimate the weight increments. [5] [6]

Detection of the regions in which the output of the neural network classifier is not appropriate and, consequently activation of the adaptation is required, is achieved through a comparison of the semantic label and confidence value produced by the fuzzy reasoning engine with the one estimated by the original neural network classifier. Whenever the difference in these values is significant, adaptation is activated. Following to this, the adapted network can be applied to similar images or consequent video frames contributing to the improvement of the obtained segmentation results.

3 The Fuzzy Reasoning Engine

This section presents the operation of the fuzzy reasoning engine together with the fuzzy knowledge base that have been used for the adaptation of the neural network classifier.

Description Logics (DLs) [3] are a family of logic-based knowledge representation formalisms designed to represent and reason about the knowledge of an application domain in a structured and well-understood way. Recently, DLs have been extended to accommodate imperfect information [12,10].

As pointed out in the fuzzy DL literature, fuzzy extensions of DLs involve only the *assertion* of individuals to concepts and the semantics of the new language. Hence, FiRE which is the reasoner used, supports fuzzy $\mathcal{SHIN}$ using an alphabet of distinct concept names ($\mathbf{C}$), role names ($\mathbf{R}$) and individual names ($\mathbf{I}$). The $\mathcal{SHIN}$ constructors regarding concepts are disjunction ($\mathsf{C}_1 \sqcup \mathsf{C}_2$), conjunction ($\mathsf{C}_1 \sqcap \mathsf{C}_2$), negation ($\neg\mathsf{C}$), full existential quantification ($\exists\mathsf{R}.\mathsf{C}$) and value restrictions ($\forall\mathsf{R}.\mathsf{C}$).Furthermore the $\mathcal{SHIN}$ language permits the hierarchy of roles as well as the use of transitive and inverse roles.

The semantic analysis module evaluates the spatial relations for each region, providing information for their location relatively to their neighboring regions. Additionally, the adaptive classifier estimates a degree of participation for each region in some trained labels.

Hence the alphabet of our fuzzy knowledge base consists of the relations representing the roles in our terminology and forming the following set:

$Roles = \{above-of, below-of, left-of, right-of\}$.

as well as the concepts that the adaptive classifier may estimate, that are :

$Concepts = \{Sky, Building, Person, Rock, Tree, Vegetation, Sea, Grass, Ground, Sand, Trunk, Dried-plant, Pavement, Boat, Wave\}$

The set of individuals consist of the segments, and the images.

Using these sets, we have defined a terminology that refines some concepts with the aid of the regions spatial relations.

For the specific architecture axioms which correct mistaken estimations of analysis are defined for further adaptation purposes. For example Sea is

Table 1. Knowledge Base ($TBox$)

$$
\begin{array}{|l|}
\hline
\mathcal{T} = \{\mathsf{SEA} \equiv \mathsf{Sea} \sqcap ((\exists \mathsf{right-of}.(\mathsf{Sea} \sqcup \mathsf{Wave})) \sqcup (\exists \mathsf{left-of}.(\mathsf{Sea} \sqcup \mathsf{Wave})) \\
\quad \sqcup (\exists \mathsf{above-of}.(\mathsf{Sea} \sqcup \mathsf{Wave})) \sqcup (\exists \mathsf{below-of}.(\mathsf{Sea} \sqcup \mathsf{Wave} \sqcup \mathsf{Sky}))), \\
\quad \mathsf{SAND} \equiv \mathsf{Sand} \sqcap ((\exists \mathsf{right-of}.(\mathsf{Sand} \sqcup \mathsf{Wave})) \sqcup (\exists \mathsf{left-of}.(\mathsf{Sand} \sqcup \mathsf{Wave})) \\
\quad \sqcup (\exists \mathsf{above-of}.(\mathsf{Sand} \sqcup \mathsf{Wave})) \sqcup (\exists \mathsf{below-of}.(\mathsf{Sand} \sqcup \mathsf{Wave} \sqcup \mathsf{Sea}))), \\
\quad \mathsf{WAVE} \equiv \mathsf{Wave} \sqcap (\exists \mathsf{right-of}.(\mathsf{Sea} \sqcup \mathsf{Wave})) \sqcup (\exists \mathsf{left-of}.(\mathsf{Sea} \sqcup \mathsf{Wave})) \\
\quad \sqcup (\exists \mathsf{above-of}.(\mathsf{Sea} \sqcup \mathsf{Wave})) \sqcup (\exists \mathsf{below-of}.(\mathsf{Sea} \sqcup \mathsf{Wave}))), \\
\hline
\mathcal{R} = \{\mathsf{above-of}, \mathsf{below-of}, \mathsf{left-of}, \mathsf{right-of}, \\
\quad \mathsf{below-of}^- = \mathsf{above-of}, \mathsf{left-of}^- = \mathsf{right-of}\} \\
\hline
\end{array}
$$

re-defined as SEA and is specified by the concept Sea assigned by the classifier and by a neighboring criterion concept which requires neighbors to be either one of Wave, Sea or Sky.

The main reasoning services provided by crisp reasoners are *entailment* and *subsumption*. These services are also available in FiRE together with greatest lower bound queries which take the advantage of the fuzzy element. Since a fuzzy $ABox$ might contain many positive assertions for the same individual, without forming a contradiction, it is of interest to compute what is the best lower and upper truth-value bounds of a fuzzy assertion. The term of *greatest lower bound* (GLB) of a fuzzy assertion w.r.t. a knowledge base has been defined in [12].

In this case, a variation of greatest lower bound reasoning service is used for the semantic refinement of the labels provided by the neural network classifier. Since the classifier is trained, we assume a correct estimation of the region label but with a mistaken confidence value. Hence, we first compute the GLB of the region of interest to the concept of interest (i.e. SEA). We then evaluate the GLB of the region of interest to the neighbor criterion concept of the concept of interest (if SEA is the concept of interest then neighbor criterion concept is $((\exists \mathsf{right-of}.(\mathsf{Sand} \sqcup \mathsf{Wave})) \sqcup (\exists \mathsf{left-of}.(\mathsf{Sand} \sqcup \mathsf{Wave})) \sqcup ...)$. If this bound is greater that the value that was originally assigned to that concept then the region value is refined, differently it remains as assigned. For example, if a region has been assigned by the classifier as Sea to degree 0.8, and it is also "below-of" a region assigned as Sky to a degree 0.9, then due to the SEA axiom defined in the terminology (Table 1), the Sea value will be refined to 0.9.(Note that if the Sky value was 0.7 then the Sea value would have remained as assigned) This value will form the desired input for the adaptation of the neural network classifier.

4 Semantically Adaptive Image Segmentation

In this section we examine how a variation of a traditional segmentation technique, the Recursive Shortest Spanning Tree, also known as RSST [7], can be used to integrate and apply the results provided by the adaptive reasoning mechanism. RSST is a bottom-up segmentation algorithm that begins from the pixel level and iteratively merges similar neighboring regions until certain termination criteria are satisfied. It uses an internal graph representation of image regions, like the Attributed Relation Graph (ARG) [4]. In the beginning, all edges of the graph are sorted according to a criterion, e.g. color dissimilarity of the two connected regions using Euclidean distance of the color components. The edge with the least weight is found and the two regions connected by that edge are merged. After each step, the merged region's attributes (e.g. region's mean color) is re-calculated. RSST will also re-calculate weights of related edges and resort them, so that in every step the edge with the least weight will be selected. This process goes on recursively, until termination criteria are met. Such criteria may vary, but they usually are either the number of regions, or a threshold on the distance.

We modify this algorithm to operate on the fuzzy sets in a similar way as if they worked on low-level features (such as color, texture, etc.). This variation

follows in principle the algorithmic definition of the traditional RSST, though a few adjustments were considered necessary and were added. S-RSST aims to improve the usual oversegmentation results by incorporating region labeling in the segmentation process [2]. The modification of the traditional algorithm to S-RSST lies on the definition of the two criteria: (a) The dissimilarity criterion between two adjacent regions a and b (vertices v_a and v_b in the graph), based on which the graph's edges are sorted and (b) the termination criterion.

For the calculation of the similarity between two regions, two approaches have been examined. The first one is based on the definition of a metric between two fuzzy sets, those that correspond to the candidate concepts of the two regions. This dissimilarity value is computed according to the following formula and is assigned as the weight of the respective graph's edge e_{ab}:

$$w(e_{ab}) = 1 - \sup_{c_k \in C} (t - norm(\mu_a(c_k), \mu_b(c_k))) \tag{6}$$

where a and b are two neighboring regions and $\mu_a(c_k)$ is the degree of membership of the concept $c_k \in C$ in the fuzzy set L_a.

Let us now examine one iteration of the S-RSST algorithm. Firstly, the edge e_{ab} with the least weight is selected, then regions a and b are merged. Vertex v_b is removed completely from the ARG, whereas v_a is updated appropriately. This update procedure consists of the following two actions:

1. Re-evaluation of the degrees of membership of the labels fuzzy set in a weighted average (w.r.t. the regions' size) fashion.
2. Re-adjustment of the ARG edges by removing edge e_{ab} and re-evaluating the weight of the affected edges.

This procedure continues until the edge e^* with the least weight in the ARG is bigger than a threshold: $w(e^*) > T_w$. This threshold is calculated in the beginning of the algorithm, based on the histogram of all weights of the set of all edges.

5 Results

In this section, certain results of the semantically adaptive architecture evaluated on real images are presented. As described in Section 1, an image is initially processed by the low-level segmentation algorithm that produces the segmented mask together with the input features for the adaptive neural network classifier. The classifier produces region-associated labels and degrees of confidence. These values pass through the semantic segmentation module and form the input for the fuzzy reasoning engine. Fuzzy reasoning provides refinement of some regions values according to which classifier adaptation is performed.

Figure 2 presents for some images, the initial output of the classifier and the semantically adaptive segmentation results.

It can be seen that based on the implicit knowledge provided by the fuzzy reasoner, semantic adaptation of the neural network classifier is achieved and used in improving the performance of the image segmentation module. The neural

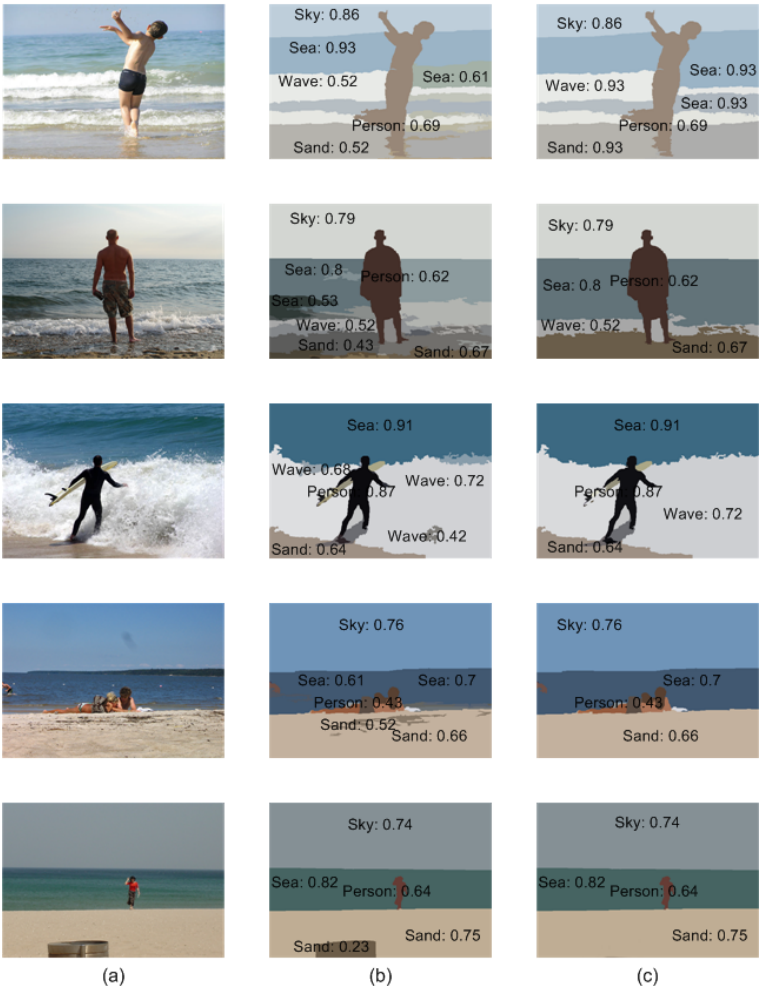

Fig. 2. (a) Original Image (b)Segmentation based on the original neural network classifier (c)Semantic segmentation using the adapted NN classifier

network accepts an input vector of 5 elements composed of the MPEG7 Scalable Color, Homogeneous Texture, Edge Histogram and Region Shape features and provides 15 outputs, corresponding to the fifteen concepts which form the Concepts alphabet of the fuzzy reasoning engine mentioned in Section 3. Based on pruning a two hidden layer architecture was formed composed of ten and six neurons respectively. Segmentation of a data set about 200 image results in a training set of 4000 regions (i.e feature vectors) which were used for training, while 50 more images were used for testing.

As indicatively shown in Figure 2 the results are very promising. The fuzzy reasoning engine propagates the confidence values of region labels, which have

"correct" spatial relations according to the fuzzy knowledge base, to the neighboring regions. These semantically corrected values are used for adaptation of the classifier in order to improve its knowledge of the specific domain and also its performance.

6 Conclusions

In this paper we have presented an architecture used for semantic adaptation of a neural network classifier in image or video segmentation. The proposed architecture combines techniques used for semantic multimedia analysis together with an adaptive classifier. A semantic segmentation algorithm and a fuzzy reasoning engine provide semantically corrected results that are used by the classifier for adaptation.

An evaluation of our architecture was made using images, presenting very promising results and a strong potential. The improved performance of adapted classifier on segments estimation could be also successfully used for segment indexing. Future work includes evaluation of the architecture using video frames and various domains.

Acknowledgment

This research was supported by the European Commission under contract FP6-027026 K-SPACE.

References

1. Adamek, T., O'Connor, N., Murphy, N.: Region-based segmentation of images using syntactic visual features. In: Proc. Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS 2005, Montreux, Switzerland, April 13-15 (2005)
2. Athanasiadis, T., Mylonas, P., Avrithis, Y., Kollias, S.: Semantic image segmentation and object labeling. IEEE Trans. on Circuits and Systems for Video Technology 17(3), 298–312
3. Baader, F., McGuinness, D., Nardi, D., Patel-Schneider, P.F.: The Description Logic Handbook: Theory, implementation and applications. Cambridge University Press, Cambridge (2002)
4. Berretti, S., Del Bimbo, A., Vicario, E.: Efficient matching and indexing of graph models in content-based retrieval. IEEE Trans. on Circuits and Systems for Video Technology 11(12), 1089–1105 (2001)
5. Doulamis, N., Doulamis, A., Kollias, S.: On-line retrainable neural networks: Improving performance of neural networks in image analysis problems. IEEE Transactions on Neural Networks 11, 1–20 (2000)
6. Ioannou, S., Kessous, L., Caridakis, G., Karpouzis, K., Aharonson, V., Kollias, S.: Adaptive on-line neural network retraining for real life multimodal emotion recognition. In: Kollias, S.D., Stafylopatis, A., Duch, W., Oja, E. (eds.) ICANN 2006. LNCS, vol. 4131, pp. 81–92. Springer, Heidelberg (2006)

7. Morris, O.J., Lee, M.J., Constantinides, A.G.: Graph theory for image analysis: An approach based on the shortest spanning tree. Inst. Elect. Eng. 133, 146–152 (1986)
8. Park, D., EL-Sharkawi, M.A., Marks II., R.J.: An adaptively trained neural network. IEEE Transactions on Neural Networks 2, 334–345 (1991)
9. Stamou, G., Kollias, S.: Multimedia Content and the Semantic Web: Methods, Standards and Tools. John Wiley & Sons Ltd, Chichester (2005)
10. Stoilos, G., Stamou, G., Pan, J.Z., Tzouvaras, V., Horrocks, I.: Reasoning with very expressive fuzzy description logics (2007)
11. Stoilos, G., Stamou, G., Tzouvaras, V., Pan, J.Z., Horrocks, I.: The fuzzy description logic f-shin. In: A International Workshop on Uncertainty Reasoning For the Semantic Web, 2005 (2005)
12. Straccia, U.: Reasoning within fuzzy description logics. Journal of Artificial Intelligence Research 14, 137–166 (2001)