Document Processing with Bayesian Network and Agent-based Programming

ANDRÁS BARTA, ISTVÁN VAJK
Department of Automation and Applied Informatics
Budapest University of Technology and Economics
H-1111, Budapest, Goldmann Gy. tér 3.
HUNGARY

Abstract: - This paper presents a document processing algorithm that is based on a Bayes network and a recursive orthogonal linear transformation. The system is extended to include agent-based programming in order to simplify the control procedures. The research also investigates the teaching procedure of the object library. The method is demonstrated on a document image processing problem, extracting the components of a circuit diagram.

Key-Words: - Document processing, Object recognition, Bayesian network, Intelligent agents

1 Introduction

Computer vision has gone through significant advancement during the past few decades. Though several real world applications are created, it is still behind its capabilities. Due to the fast technical development the main limitation is not memory and speed any more. More emphasis should be put on the control and teaching procedures of the object recognition systems. In this research we try to address these problems by combining Bayesian network and agent-based programming.

General object recognition and part based structural description has a long history. Several object recognition system with structural object description have been created: VISION (Hanson, Riseman, 1978), SIGMA (Hwang at al., 1986) , SPAM (McKeon at al., 1985), ACRONYM (Brooks, Binford, 1981), SCHEMA (Draper et al., 1989). These systems, their successes and failures are investigated by Draper [3]. He finds that knowledge-directed vision systems typically failed because the control problem for vision procedures was never properly addressed as an independent problem. He also argues that problems are created because adding new features or new object classes solves many problems initially but as the system grows they make the system intractable. This paper searches solutions for these problems. The control problem of the object recognition system is treated in the framework of Bayesian networks. The system learning is addressed by an incremental learning procedure. The Bayesian framework is extended by agent-based programming in order to simplify the control procedures in case of complicated object descriptions. A similar approach is used by Takatsuka et. al [7]. They used Hopfield neural network to solve the sub-graph matching problem.

The algorithm is based on a Bayes network and a recursive orthogonal linear transformation. The method is demonstrated on a document image processing problem, extracting the components of a circuit diagram. Many old blue-prints of electrical equipment are sitting on shelves. Converting them to a meaningful digital representation would make it possible to search and retrieve them by content.

Probabilistic Bayesian networks are investigated extensively in the literature. Perl presented a tree based belief network inference with linear complexity [5]. Dynamic tree structures are gaining popularity, because of their better object representation capabilities [9]. Okazaki at al. proposes a method for processing VLSI-CAD data input [6]. His system is implemented for digital circuitry where the components are mainly loop-structured symbols. Symbol identification is achieved by a hybrid method, which uses heuristics to mediate between template matching and feature extraction. The entire symbol recognition process is carried out under a decision-tree control strategy. Siddiqi at al. present a Bayesian inference for part based representation [8]. The object subcomponents are represented by fourth order polynomials. The recognition is based on geometric invariants, but it does not provide a data-structure for representing the components. A similar approach to our research was taken by Cho and Kim [2]. They modelled strokes and their relationships to on-line handwriting recognition. Their system also used Bayesian networks for inference, but only for fixed models. They also assumed Gaussian distributions which can not be applied for circuit diagram analysis.

2 Creating the Object Library

In object recognition systems the creation and management of the object library is an important issue. The object library of the object recognition system should be structured hierarchically. This hierarchical structure reflects the underlying hierarchical structure of natural and man made objects. The 1/f power law of natural images is an indication of this hierarchy [10]. The question is what the hierarchical structure of image elements is and how it can be learned from images. It is an open question whether the visual vocabulary should accurately reflect the hierarchical structure of real objects. In speech processing the vocabulary is defined quite clearly: phonemes, words, phrases and sentences. This visual dictionary for object recognition either can be learned from images or manually programmed. Both methods have their disadvantages and generally they are used in a combination. In automatic learning it is difficult to maintain a suitably structured dictionary. One possible way is define an MDL like measure which forces the learning procedure to create simple objects. Even with this step there is still no guarantee that the actual library structure follows the real object structure. The manual coding involves not just the case when the object geometries are coded directly, but when the image processing algorithms are designed to extract a special type of image elements. In this work a learning procedure is used that tries to combine the two methods. The dictionary is learned from images, but the images are selected and pre-processed to contain the visual vocabulary. This is an incremental process, first the lower level image element are learned then based on the learned image bases more and more complex images are processed. This is similar way as humans learn in schools. There are several advantages of an incremental learning procedure.

1. There is no need to program any object description into the object recognition software
2. The structure of the object library can be controlled
3. Different object recognition systems can be taught with the same sequence of images, therefore their performance can be compared.

3 Model Based Bayesian Network

Bayesian networks are well suited for image processing applications and it is used for this research because of the following advantages:

- provides probabilistic representation
- provides a hierarchical data structure
- provides an inference algorithm
- separates the operating code from the data representation
- it is capable of processing both predictive and diagnostic evidence
- provides and inhibiting mechanism that decreases the probabilities of the unused image bases

Bayesian network is defined for this application as follows [1]. An image feature is represented by lower level image bases in a recursive way.

$$\xi_j = \sum_{i=1}^{n} T(\xi_i(\mathbf{a}_i), \mathbf{r}_i) \qquad (1)$$

T is an operator that performs an orthogonal linear transformation on the image bases. The parameters of the transformation are stored in the $\mathbf{r}_i$ parameter vector. The image bases may be parameterized by an $\mathbf{a}_i$ attribute vector. Since features belong to parameterized feature classes the $\mathbf{a}_i$ vector is necessary to identify their parameters. This description defines a tree structure. The tree is constructed from its nodes and a library. The T transformation has three components, displacement, rotation and scaling. The four parameters of the transformation of node i are placed in a reference vector

$$\mathbf{r}_i = \begin{bmatrix} \mathbf{x}_i^r & s_i^r & \varphi_i^r \end{bmatrix} \qquad (2)$$

where $\mathbf{x}_i^r = \begin{bmatrix} x_i^r & y_i^r \end{bmatrix}$ is the position of the image element in the coordinate system of its parent node, s_i^r is the scaling parameter and φ_i^r is the rotation angle. The conditional probability parameters $\theta_{i,j}$ are learned as relative frequencies. It can be shown that the distribution of the $\theta_{i,j}$ parameters is a Dirichlet distribution [4]. The conditional probabilities of the network can be described

$$p(\theta_1, \theta_2, ..., \theta_{L-1}) = \frac{\Gamma(n)}{\prod_{k=1}^{L} \Gamma(n_k)} \theta_1^{N_1 - 1} \theta_2^{N_2 - 1} ... \theta_K^{N_L - 1} = \\ = Dir(\theta_1, \theta_2, ..., \theta_{L-1}; n_1, n_2, ..., n_L) \qquad (3)$$

where n_k is the number of time node k occurs in the sample data and $n = \sum_{k=1}^{L} n_k$ is the sample size. The

$\Gamma(x)$ function for integer values is the factorial function, $\Gamma(x)=(x-1)!$. These conditional probabilities are learned from the training data. In a typical Bayesian network the direction of the edge shows the casual relationships. In image processing applications it can not be said whether the object is causing the feature or the feature is causing the object; the edges of the tree may go in either direction. The direction depends on whether we are using a generative or descriptive model [10].

3.1 The Algorithm

The recognition process starts by selecting a new image component. This single node tree is expanded by adding a structure shown on Fig. 1. By adding more and more nodes the whole image tree is created.

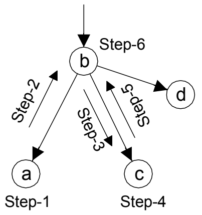

Fig. 1. The steps of the algorithm

This node expansion is performed in the following steps:

Step 1: A new image component (a) is selected randomly based on the node probability distribution. The selection is performed by the roulette-wheel algorithm. In case of new node the prior probability is used.

Step 2: This new evidence starts the belief propagation of the network. Based on the conditional probabilities several parent node hypotheses are created (b, upward hypothesis). These object hypotheses are described by library index and coordinate system of the node. The coordinate system of the object hypothesis (a) $\mathbf{i}_{pi}=\begin{bmatrix}\mathbf{x}_{pi} & s_{pi} & \varphi_{pi}\end{bmatrix}$ can be calculated by the following coordinate transformation:

$$
\begin{aligned}
s_{pi} &= \frac{s_i}{s_k^r} \\
\varphi_{pi} &= \varphi_i - \varphi_k^r \\
\mathbf{x}_{pi} &= \mathbf{x}_i - \mathbf{x}_k^r s_{pi} \begin{bmatrix} \cos\varphi_{pi} & \sin\varphi_{pi} \\ -\sin\varphi_{pi} & \cos\varphi_{pi} \end{bmatrix}
\end{aligned}
\tag{4}
$$

where $\mathbf{i}_i=\begin{bmatrix}\mathbf{x}_i & s_i & \varphi_i\end{bmatrix}$ is the coordinate system of the image component (b) and $\mathbf{r}_k=\begin{bmatrix}\mathbf{x}_k^r & s_k^r & \varphi_k^r\end{bmatrix}$ is the reference vector of child node of the library tree.

Step 3: This parent node hypothesis can be projected back to the image. This projection creates child hypotheses not only for node c, but all of the child nodes of b (for example d).

Step 4: A search is performed to match this projected child node hypotheses. If this object hypothesis matches one of the already identified subtrees then they are combined. If no match has been found then a new hypothesis are created (downward hypothesis) for the child node. If the child hypothesis is one of the lowest level image components then it is compared against the image, based on a distance measure. This distance measure can be, for example, the Euclidean distance. It should be defined for every basic image element independently; in our case for lines, circles and arcs. The results of the child node comparisons are converted to probability by an arbitrarily chosen function.

Step 5: The probability of the child modes propagates upward as new evidence. The upward probabilities are combined to calculate the probability of root b.

Step 6: Only the high probability nodes are processed, the others are neglected. This is true for both the upward and downward object hypotheses. This process creates a structure with several root nodes. These root nodes can be an input to a next level of recognition step. The root nodes are either sub-components that the algorithm will grow further or they are the final solutions.

The search method is adaptive and local; only certain area of the image is processed at a time. This is advantageous for images with noise or clutter. The calculation complexity of the algorithm can be described by the following dependencies:

- The complexity is linear with the number of nodes.
- The complexity does not depend on the size of the library but only on the number of nonzero upward conditional probability values. The complexity is lower if these probability values are concentrated in few high probability entries.
- The complexity is lower if the average object size is higher.
- The complexity is higher if the objects have symmetries.
- The complexity is higher if a node has several child nodes with identical library index.

4 Agent Oriented Processing

In order to take advantage of the feature based recognition methods the Bayesian network is extended by agent technology. This allows more flexible symbolic representation.

Agent based computing is a popular research area. An agent is a piece of software that can complete a task independently without user interaction. A common task can be solved by the cooperation of autonomous entities. Agents are almost an extension of object-oriented computer design principles. There is confusion in the literature what makes a system truly agent based. They are used quite broadly since many computer vision algorithms can be viewed as agent based methods. Agents are frequently used components in pattern recognition and object recognition applications. The SPAM, SIGMA and SCHEMA systems can be considered as agent based systems. Mackenzie used agents for sketch interpretation task [11].

The critical issue of agent based computing is the definition of communication among agents. One frequently used way of communication is to use a medium described as a Blackboard. The Blackboard method has been originally introduced by Hayes-Roth [12]. The blackboard is simply a central repository for all shared information. A typical session begins by a problem being written onto the blackboard, and all known information and assumptions. As individual agents realise they can contribute to the solution with their own expertise they approach the facilitator and are allowed to add new information to the blackboard [11].

In our research every image agent communicates directly to its neighbours, to its children and to its parent. It is very similar to the network propagation of the Bayesian network. The main difference is that object description information also flows in the network. Every agent or node collects information about its neighbours and sends it to its parent. The information is coded in symbolic form. For the circuit diagram processing line feature description is used. Every line object searches its surrounding and calculates the symbolic relationship of its neighbours. Parallelism (P), collinearity (C), junction type (V,L,T), line angle relationships are used. The object sends this symbolic massage to its parent. For example the message can be "PLT". The parent collects these messages and compares to the library description and sends a message back. This return message contains a probability value that has the same role as in the Bayesian network and a symbolic message to tell the children the type of relationships to search for. The information flows the same way as described in section 3.1 for the model based recognition process. Based on the probability values child parent relationships are created or terminated. This is similar to the "cut and merge" region segmentation methods. The parent hypotheses are created based on the conditional and prior probabilities the same way as for the Bayesian network.

5 Demonstration of the Method

In order to demonstrate the method a circuit diagram extraction is carried out for computer generated images (Fig. 2. shows a sample circuit diagram). The identification of the image components is performed by calculating the probabilities or beliefs of the Bayesian network modes.

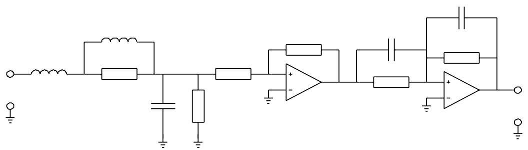

Fig. 2. Sample circuit diagram

First, the image library is created. For the demonstration the lower level image bases are line, circuit and arc. From these image bases images of new complex image bases are created. These images are processed and the object trees are placed into the library. Fig. 3 shows the sequence of images and the created library.

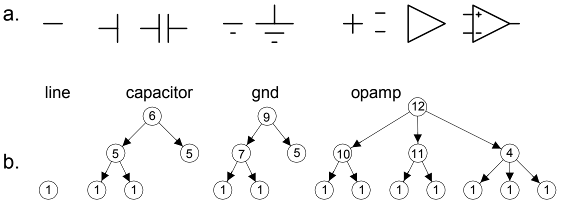

Fig. 3. A sequence of images is used to teach the object library. (a) shows an image sequence to teach the components of the sample circuit diagram. (b) shows the library trees created by the process

In this research the conditional probabilities are calculated for both directions. Since the algorithm uses upward and downward processing this simplifies the calculations. The upward $(x \leftarrow y)$ conditional probability is calculated from the relative frequencies. For the downward $(x \rightarrow y)$ conditional probability calculation the library object definitions are used. Unity distribution is assumed on each downward edge of the nodes.

Several simulations have been performed to evaluate the method. The algorithm is programmed in Matlab object oriented environment. The circuit diagram is processed and the components are extracted in a few seconds on a regular PC. The circuit processing is repeated 500 times and a normalized histogram of the necessary iteration steps is generated. The result is shown on Fig. 4.

The presented method can be integrated with other image processing and object recognition problems. It is demonstrated only for circuit diagram extraction but it can be used for other type of object recognition problems. In this paper only a simple object description is presented, but the method with little modification can be used for more complex objects also.

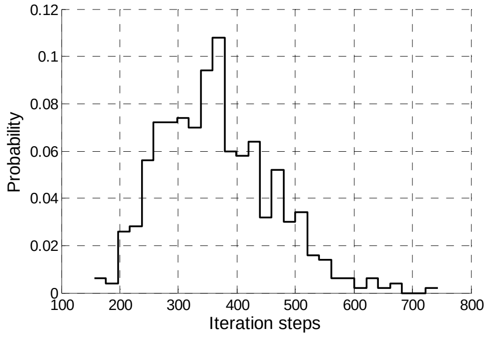

Fig. 4. The histogram of the necessary iterations to process the sample circuit diagram

5 Conclusion

This research shows that the combination of Bayesian network and agent programming can simplify the control and teaching processes of object recognition systems. However, more research is needed to find to optimal agent communication and behaviour for a given computer vision problem.

Acknowledgements

The work was supported by the fund of the Hungarian Academy of Sciences for control research and partly by the OTKA fund T042741. The supports are kindly acknowledged.

References:

[1] Barta A, Vajk I., Document Image Analysis by Probabilistic Network and Circuit Diagram Extraction, *Informatica - An International Journal of Computing and Informatics*, Special Issue on Multimedia Mining in Soft Computing, June 2005,

[2] Cho S.J., Kim J.H., Bayesian Network Modeling of Strokes and Their Relationships for On-line Handwriting Recognition, *Proceedings of the Sixth International Conference on Document Analysis and Recognition* (ICDAR'01), pp. 86-91, 2001

[3] Draper B., Hanson H., Riseman E., Knowledge-Directed Vision: Control, Learning and Integration, http://www.cs.colostate.edu/~draper/publications /draper_ieee96.pdf *Proceedings of the IEEE*, 84(11), pp. 1625-1637, 1996

[4] Neopolitan R. E., *Learning Bayesian networks*, Pearson Prentice Hall, 2004

[5] Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Interference*, Morgan Kauffmann Publishers, 1988

[6] Okazaki A., Kondo T., Mori K., Tsunekawa S., Kawamoto E., An Automatic Circuit Diagram Reader With Loop-Structure-Based Symbol Recognition , *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 10, No. 3, pp. 331-341, May 1988

[7] Takatsuka M., Caelli T. M., West G. A. W., Venkatesh S., An application of ''agent-oriented'' techniques to symbolic matching and object recognition, *Pattern Recognition Letters* 23, pp. 419–429, 2002

[8] Siddiqi K., Subrahmonia J., Cooper D., Kimia B.B., Part-Based Bayesian Recognition Using Implicit Polynomial Invariants, *Proceedings of the 1995 International Conference on Image Processing* (ICIP), pp. 360-363, 1995

[9] Storkey A.J., Williams C.K.I., Image Modeling with Position-Encoding Dynamic Trees, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 25, No. 7, July pp. 859-871, 2003

[10] Zou Song-Chun, Statistical Modeling and Conceptualization of Visual Patterns, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 25, No. 6, June, pp 691-712, 2003

[11] Mackenzie G., Agent-Based Sketch Recognition, *PhD Thesis*, http://www.cs.nott. ac.uk/IPI/publication42.html, 2003

[12] Hayes-Roth, B. A blackboard architecture for control. *Artificial Intelligence*, 26, pp. 251-321, 1985