World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:1, No:7, 2007

# Integrating Low and High Level Object Recognition Steps by Probabilistic Networks

András Barta, and István Vajk



***Abstract*****—**In pattern recognition applications the low level segmentation and the high level object recognition are generally considered as two separate steps. The paper presents a method that bridges the gap between the low and the high level object recognition. It is based on a Bayesian network representation and network propagation algorithm. At the low level it uses hierarchical structure of quadratic spline wavelet image bases. The method is demonstrated for a simple circuit diagram component identification problem.

***Keywords*****—**Object recognition, Bayesian network, Wavelets, Document processing

## I. 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 object recognition systems. In this research we try to address these problems by combining the low level segmentation problem with the high level object recognition process. 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.

In pattern recognition applications the low level segmentation and the high level object recognition are generally considered as two separate steps. The main contribution of the paper is that it presents a method that bridges the gap between the low and the high level object recognition. The system consists of a hierarchical data structure, a network calculation method that is capable of processing top-down and bottom-up information flow and a set of image elements. In many object recognition systems adaptivity is an important requirement, since most images contain significantly more information that is necessary for recognizing the objects. In an adaptive system only the part of the image is processed that contains the required object descriptions. This can be achieved by applying both top-down and bottom-up processing. The top-down information flow is created by an object hypotheses generating process. 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.

In the next section we overview the related work. Section III briefly investigates the architecture of object recognition systems. This part is included in the paper to clarify the role of the individual components of the system. In section IV the Bayesian network definition is given for object representation. The network is defined by the node description and the conditional probabilities. In this section an object recognition algorithm is also defined for model based object recognition. In section IV and V low level image segmentation is addressed. The given low level processing method can be integrated into the previously defined graph representation and Bayesian network framework. At the lowest level we represent the image by the disk transform. The disk transform is based on continuous wavelet transform. In the simulation section low level segmentation and a high level object recognition problem is investigated. In order to demonstrate the method a simple circuit diagram document image processing example is chosen for the high level processing. In this example the image element dictionary creation is also addressed. The method is also tested on a real image. In the final section we conclude our work.

Manuscript received March 12, 2006. This work was supported by the fund of the Hungarian Academy of Sciences for control research and by part by the OTKA fund TO42741. The supports are kindly acknowledged.

András Barta is with the Department of Automation and Applied Informatics, Budapest University of Technology and Economics, Budapest, H-1111 Budapest, Goldmann Gy. tér 3., Hungary (phone: 36-1-463-2870; fax: 36-1-463-2871; e-mail: barta@ aut.bme.hu).

István Vajk is with the Department of Automation and Applied Informatics, Budapest University of Technology and Economics

### A. Related work

In this section we briefly review the literature related to this work. Since the presented method applies several theories the review reflects this fragmentation.

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 image features. Wavelets and digital filters are used in the literature many ways. Freeman and Adelson provided a framework for steerable filters [13]. They presented an architecture to synthesize filters of arbitrary orientations from linear combinations of basis filters. They, however not addressed the problem of joining the edge pixels. Sung, Bang and Choi constructed hierarchical network of wavelets for handwritten numeral recognition [14]. They used only the imaginary components of Gabor wavelets for the identifications. Deng, Lati and Regentova used cubic splines for document processing [15]. They applied a classification method for segmenting documents. They applied three-means or two-means classification for classifying pixels with similar characteristics after feature estimation of spline wavelet transforms.

In image processing the selection of data structure is important and an open question. Generally the structural relationships of the object components can be captured by a graph [11]. In many vision applications, however, simpler data structure is sufficient to represent the image components. The tree structures that are applied for image processing tasks can be classified as fixed or dynamic. In a fixed architecture the tree structure does not change during the processing. In case of dynamic model the tree structures changes depending on the image content. A frequently used fixed model is the quadtree structure. Quadtree models are frequently used because they have simple structure and able to represent the image at different scales in a pyramid structure. The disadvantage of using fixed tree structure is that it produces blocky images, since the adjacent pixels may belong to different branches of the tree [10]. Since the pixels have different neighbors at a different three branches, this causes discontinuity at the region boundary in case the boundary moves across different branches. To eliminate the problem dynamic tree structures are proposed. Bouman and Shapiro modify the quadtree structure to allow children to connect to multiple parents (coarse level). In their augmented pyramidal structure an overlapping is introduced between neighboring parents [10]. Adams proposed a dynamic structure by allowing edges to connect to not just to their natural parents [12]. He introduces a probability on the different tree structures. His method was further improved by Storkey, Christopher and Williams [6]. Their algorithm assigns position information and an object class label to every node. In this paper a dynamic structure is presented. The tree is of the object descriptions are built up incrementally.

## II. Object recognition System Architecture

In this section we briefly look at the motivation of creating the system. An object recognition system has the following functional components: object representation, object recognition process and object learning.

The representation type of the objects should be chosen based on the structure of the real world objects. In this research we start from the premise that the objects are hierarchically structured. That means that any object is built up from lower level components. This is true for both natural and man made objects. In object recognition systems this leads to the introduction of image elements (image components, image bases, features) and a hierarchical data structure. The image elements are the frequently occurring structures of the images. The relationships of the image elements should be stored in a hierarchical data structure, usually in a hierarchical graph structure.

The object recognition process means the identification of the image elements and their structural relationships. Generally this is a highly complex problem and it can be solved only for the simplest objects. The problem is simplified if we assume statistical independence among some image elements. Due to this independence every image element will be influenced by only a few neighboring image elements or objects. Some of these will be higher and some of them will be lower in the object hierarchy. In order to calculate their influence both up-down and bottom up network processing is necessary. There are two more advantages of the independence assumptions. It is possible to process different parts of the image parallel and an adaptive processing method can be implemented. Adaptive processing is an important requirement because it is enough to process just a few important object features for the recognition. This can be done by moving the visual attention of the system [16].

The capabilities of the object recognition systems are significantly influenced by the stored object descriptions. It is an open question whether this 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 unsupervised 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. Still, even with this step there is no guarantee, that the actual dictionary structure follows the real object structure. In supervised learning procedures the dictionary can be learned from images, but the images should be selected and pre-processed to contain the visual vocabulary. This is an incremental process, first the lower level image element are learned and then based on the learned image bases more and more complex images are processed. This is a similar way as humans learn in schools.

Based on these ideas we can set a few requirements for an object recognition system.

- A hierarchical data structure.
- A network calculation method, that is capable of processing top-down and bottom-up information flow.
- A set of image elements, which can be taught by preprocessed image sequence.

Based on these requirements we have selected a hierarchical graph structure, a Bayesian network and a supervised learning method for the system components.

## III. 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], [2]. An image feature is represented by lower level image bases in a recursive way.

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

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 the dictionary. 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

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

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 $\varphi_i^r$ 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

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

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].

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

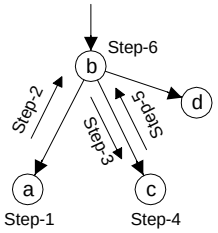


Fig. 1 Steps of the algorithm

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 the dictionary 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:

$$s_{pi} = \frac{s_i}{s_k^r} \tag{4}$$

$$\varphi_{pi} = \varphi_i - \varphi_k^r \tag{5}$$

$$\mathbf{x}_{pi} = \mathbf{x}_i - \mathbf{x}_k^r s_{pi} \begin{bmatrix} \cos\varphi_{pi} & \sin\varphi_{pi} \\ -\sin\varphi_{pi} & \cos\varphi_{pi} \end{bmatrix} = \mathbf{x}_i - \mathbf{x}_k^r s_{pi} \mathbf{R}_\varphi \tag{6}$$

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 dictionary 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 lower level 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 dictionary 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 dictionary index.

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.

## IV. Wavelets for Object Recognition

Wavelets are frequently used for image processing and object recognition applications because they provide position and frequency information and with them local processing is possible. However there is significant difference in using wavelet for image representation and object recognition. In case of image representation the main task is to code the image in denser form and then reconstruct the original image. In object recognition it is not necessary to represent every detail of the image and perfect reconstruction is also not needed. In case of image representation the purpose of the image coding is tile the position-frequency plane as much as possible. In object recognition applications the object descriptions are advantageous either in the frequency or in the position domain, therefore exact tiling is not necessary. A texture pattern may be identified better by the frequency response and therefore frequency domain description is more advantageous. For localized features however position domain description is simpler.

In object recognition applications the selection of wavelets for low level image base representation is a critical step. The following issues should be considered:

- Translation invariant representation
- Support size
- Frequency domain resolution
- Real or complex
- Orthogonal representation

Translation invariant wavelet representation is a critical requirement for object recognition. In case of translation invariant representation if the objects are translated then the wavelet coefficients are also translated without any change in values. If the representation is not translation invariant, even small position variation may result significant coefficient changes. The coefficient variations would make the higher level object recognition difficult.

The support size of the wavelets is an important factor in practical implementations. Compact support is necessary for fast calculations.

The frequency resolution of the wavelets depends on the detectable image feature. For texture identification high frequency resolution wavelets are necessary

Complex wavelets have some advantages to real wavelets. They can provide linear phase filter representation and they are easier to use in translation invariant applications. They provide, however very redundant representation, because the coefficients have imaginary and real components.

For multilevel representation the image element are represented by lower level image bases. In case of wavelet basis the wavelet scaling function can be constructed with higher resolution scaling functions, where $h(n)$ is a linear

filter [11][12].

$$\phi(x) = \sum_n h(n)\sqrt{2}\phi(2x-n) \tag{7}$$

Similarly, the wavelet function can be constructed with higher resolution scaling functions,

$$\psi(x) = \sum_n g(n)\sqrt{2}\phi(2x-n) \tag{8}$$

In this paper spline wavelets are used for low level image base representation, because they have several properties that are advantageous for object recognition.

- They are symmetric.
- Exact reconstruction is possible.
- They provide a bi-orthogonal system.
- The wavelet decomposition can be done by mirror filters.
- They have compact support.

The one-dimensional quadratic spline wavelet can be calculated by the following mirror filters (in Matlab *rbio3.1*).

| | | | | |
|---|---|---|---|---|
| low fr. decomp. = | 0.1250 | 0.3750 | 0.3750 | 0.1250 |
| hi fr. decomp. = | 0.2500 | 0.7500 | -0.7500 | -0.2 |
| low fr. reconstr. = | -0.2500 | 0.7500 | 0.7500 | -0.2500 |
| hi fr. reconstr. = | 0.1250 | -0.3750 | 0.3750 | -0.1250 |

Fig. 2 shows the scaling function and Fig. 3 the wavelet function decomposition. In case of hierarchical object structure this recursive decomposition is important.

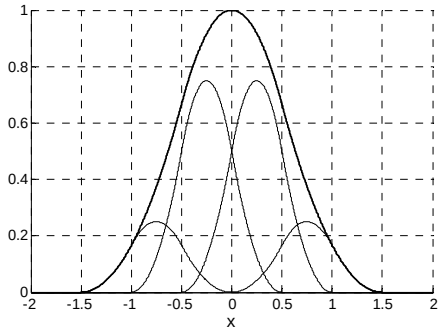


Fig. 2 Quadratic spline scaling function decomposition

Quadratic spline wavelets provide good frequency localization. Its scaling function frequency response can be given in analytical form. Its Fourier transform is,

$$\Phi(\omega) = \left(\frac{\sin(\omega/2)}{\omega/2}\right)^3 e^{-i\omega/2} \tag{9}$$

The fastest method of calculating the wavelet coefficients is the *fast wavelet transform* (FWT). Unfortunately the FWT is not translation invariant because of dyadic sampling of the position parameter. At higher scale due to this fixed re-sampling the sampling grid and the features positions are not aligned. The continuous wavelet and the dyadic wavelet transform (à trous algorithm) is translation invariant, but they provide highly redundant representations [11]. Adaptive re-sampling is a way to solve this problem. A translation invariant representation can be achieved by optimization. The image base location is optimized in a continuous position parameter then the surrounding of the image feature is re-sampled relative to this maxima position.

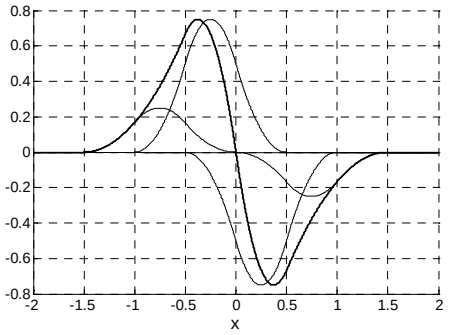


Fig. 3 Quadratic spline wavelet function decomposition

For image processing two-dimensional wavelets should be used. From going to two-dimensional wavelets from one dimensional is an not obvious problem. The low level image feature detection requires directional sensitivity. Directional sensitivity can be achieved by selecting different scaling parameters for the two main axes. Fig. 4 shows some of the low level spline image bases.

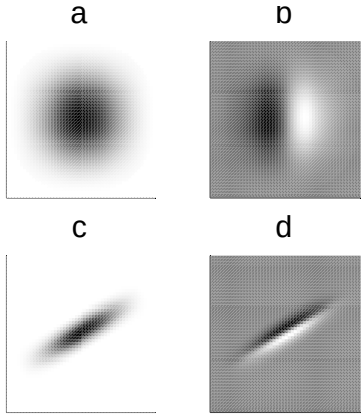


Fig. 4 Quadratic spline wavelet image bases. a. Scaling function b. Wavelet function c. Directional ridge filter d. Directional edge filter

The edge detecting two-dimensional wavelet (Fig. 4-b) uses the scaling function for one axis and the wavelet function for the other in order to detect the intensity variation of the image,

$$\xi(x,y) = \psi(x)\phi(y) \tag{10}$$

The image bases can be rotated by applying the transformation equations (6) with the $\mathbf{R}_{\varphi}$ rotation operator. Directional sensitive wavelet is shown on Fig. 4-c and Fig. 4-d.

The wavelets have several good features in case of image processing applications, but their integration to a complete object recognition system should be solved. In order to able to apply the wavelet transform in our integrated system the disk transform is introduced.

## V. Low Level Representation by Disks Transform

The presented Bayesian network representation and algorithm can be used for low level image element detection. The recognition process is based on local continuous wavelet decomposition. The purpose of the wavelet decomposition in case of object recognition is to represent the image with as few as possible wavelet coefficients. This can be achieved by selecting wavelet bases which are very similar in shape to the detectable image elements. We introduce a new object, the disk. In the hierarchical structure of image bases the disk is between the pixel and the low level image features, edges and lines. The pixel resolution of an image depends on the image creation process and it is independent of the content of the image. On the other hand the resolution of the image elements depends on the image content. The disk can bridge this gap, because it is constructed from pixels and its size is selected to reflect the resolution of the image content. With the introduction of the disk the picture can be represented independently from the pixel resolution.

There is significant difference in the working of the human visual system and the digital image processing systems at the lowest level. The human brain is capable of directing the visual attention to any area of the image by moving the head and the eye. This movement determines a continuous position parameter. The eye can be also focused to adjust the processing scale to the image content. Due to the continuous nature of the visual attention position and scale parameter the human brain is capable of creating a continuous image from the discrete photoreceptors. A similar effect can be produced by introducing the disk transform.

The disk is determined by a continuous position parameter a discrete scale parameter and a disk function. The $\mathsf{D}$ disk transform creates a continuous signal from the discrete $\boldsymbol{I}\left[x_i, y_j\right]$ pixel data:

$$D(x,y) = \mathsf{D}\left\{I\left[x,y\right]\right\}. \tag{11}$$

The task of the object recognition is to reconstruct the real world objects. The object recognition algorithm constructs the objects from the discrete pixel data. With the application of the disk the object recognition task and the object recognition algorithm becomes almost identical. Fig. 5 shows the role of the disk in the object recognition process. The ideal $D'(x,y)$ and the real $D(x,y)$ transform differ in an error term:

$$\begin{aligned} D'(x,y) &= \mathsf{D}'\left\{I\left(x,y\right)\right\} = \\ &= \mathsf{D}\left\{I\left[x_i,y_i\right]\right\} + e(x,y) = D(x,y) + e(x,y) \end{aligned} \tag{12}$$

The $e$ error term is the results of the continuous-discrete and discrete-continuous transformations and the inaccurate disk representation. It can be reduced by the selecting the right magnification factor and the pixel resolution during the image creation process.

The $\mathsf{A}$ disk area can be considered as a small window through which the real image is viewed. The shape of the window can be chosen to any form. A circular window is chosen. In case of circular shape the effect of window rotation on the disk transform is minimized. The disk area is determined by the center position and radius of disk, $\mathsf{A}(x,y,r)$. The radius of the disk, i.e. the scale parameter is a free parameter that should be adjusted to the image content. The behavior of the disk is determined by the $\gamma(x,y)$ disk function. The disk function is a wavelet that is defined on the disk area.

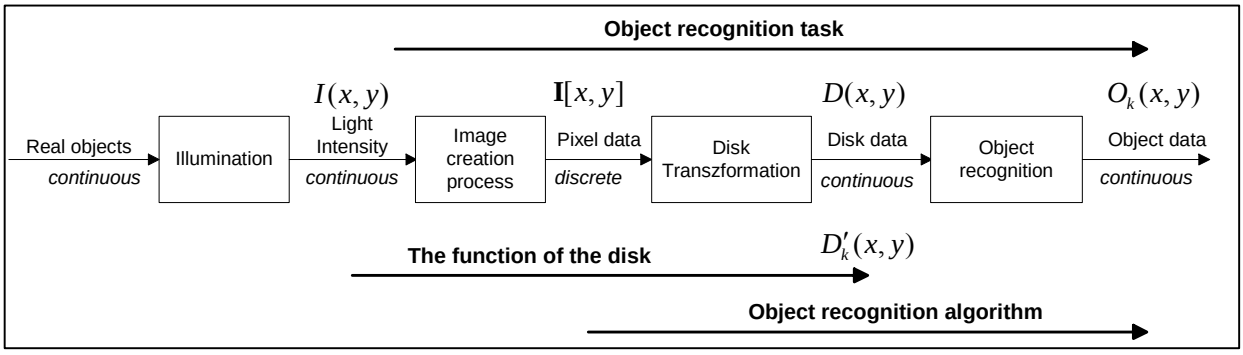


Fig. 5 The role of the disk in the object recognition process

The value of the disk transform is calculated by a continuous convolution,

$$D(x,y,r) = \mathsf{D}\left\{I[i,j]\right\} = \\ = \int\limits_{x',y' \in \mathsf{A}(x,y,r)} I(x+x', y+y')\gamma(x',y')\,dx'dy' \tag{13}$$

where $\mathbf{I}(x, y)$ is the value of the image at the continuous $x, y$ position and $r$ is the radius of the disk.

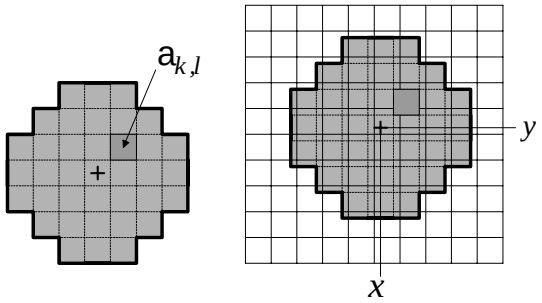


Fig. 6 The $\mathsf{A}$ disk window

The continuous integral generally can not be calculated therefore an approximation is necessary. The integral is approximated by the trapezoid rule. The disk area is constructed from small $\mathsf{a}$ square whose size is equal to the pixel size (Fig. 7). We assume that the disk function $\gamma_{k,l}$ value is constant on the $\mathsf{a}_{k,l}$ square. Fig. 6 shows the disk window for *r=4.* With this selection the continuous convolution can be calculated by the calculation of four discrete convolutions.

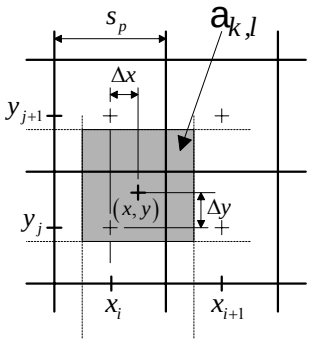


Fig. 7 The calculation of the disk transform

The difference of the continuous and discrete position is

$$\Delta x = mod\left(x, s_p\right) \\ \Delta y = mod\left(y, s_p\right), \tag{14}$$

where $s_p$ is the size of the pixel. In relative units they are,

$$\nu_x = \frac{\Delta x}{s_p}\ \nu_y = \frac{\Delta y}{s_p}. \tag{15}$$

The continuous integral on the disk is the sum of the $d_{k,l}$ integrals calculated on the $\mathsf{a}_{k,l}$ squares.

$$D(x,y) = \sum_{k,l} d_{k,l}(x,y) \tag{16}$$

$$d_{k,l}(x,y) = \int\limits_{x',y' \in \mathsf{a}_{k,l}} I(x',y')\gamma_{k,l}\,dx'dy' = \\ = \gamma_{k,l} \int\limits_{x',y' \in \mathsf{a}_{k,l}} I(x-x', y-y')\,dx'dy' \tag{17}$$

$$d_{k,l}[i,j] = \gamma_{k,l}\big(I[i+k, j+l](1-\nu_x)(1-\nu_y) + \\ I[i+k+1, j+l]\nu_x(1-\nu_y) + I[i+k, j+l+1](1-\nu_x)\nu_y + \\ + I[i+k+1, j+l+1]\nu_x\nu_y\big)$$

$I(x,y)$ is a continuous function whose value is constant on the pixel the areas, $I\left(x_i, y_j\right) = I[i,j]$.

From the above equations the disk transform is

$$D(x,y) = D[i,j](1-\nu_x)(1-\nu_y) + \\ + D[i+1,j]\nu_x(1-\nu_y) + D[i,j+1](1-\nu_x)\nu_y + \\ + D[i+1,j+1]\nu_x\nu_y \tag{18}$$

$D[i,j]$ is a discrete transform which can be calculated by discrete convolution,

$$D[i,j] = conv\left(\gamma[i,j], I[i,j]\right) = \\ = \sum_{k,l \in \mathsf{A}} \gamma[x_k, y_l] I[i-k, j-l] = \\ = \left\langle \gamma[i,j], I[i-k, j-l] \right\rangle \tag{19}$$

The discrete convolution can be calculated by scalar multiplication, which is important for fast calculations. The disk transform assigns a $D(x,y)$ value to every point of the image area. One advantage of the disk transform is that it can be applied adaptively. The size and the resolution can be

adjusted to the image content. Any continuous function that is capable of representing the image elements can be used for the $\gamma$ disk function. In this paper two-dimensional wavelets are used.

The image element is detected by placing the disk on the image in different positions and orientation angles, and calculating its cost value. The position scale and orientation is described by the coordinate system transformation. The higher level image elements are constructed by the combination of these disks. They can be identified by the same network calculations. The method has an advantage to other conventional edge detection algorithms because both bottom-up and top-down calculation can be used. The neighborhood of the found image element is searched again for new elements. The relationships of these new disks are calculated and they can be used for higher level image component detection. In case of linear type feature detection the curve segments are detected and joined together to form curves. After the first segment is found, the location of the next one can be guessed from the position and orientation of the previous disk. This representation is simpler than a general graph structure, since the points are calculated sequentially. From the coordinate system of a disk the position and the first derivative of the curve can be gained directly. Edge detection can be performed the same way as line detection, except the disk function has to be chosen differently.

With this method lines, circles and parameterized curves can be detected. The method can be used for detecting any other line type features the same way, but in this paper we focus on line detection Disks can also be defined to identify special types of picture elements, for example special types of edge profiles or textures. This can be achieved by defining mask functions that identifies that type of edge profiles. The disk definition can also be extended to use statistical properties of the disk area.

## VI. Simulation Results

The integrated method is tested for both low and high level image component detection. The system was built in Matlab object oriented environment. A flexible, general object graph structure is created. A user defined objects are assigned to every node and edge. These objects definitions determine the hierarchical object descriptions. In this work for the node and edge objects definitions (1) and (2) are used. At the lowest level a disc object is assigned to every node. A simulation runs on regular PC and is capable of segmenting the image and creating the object structure in a few seconds.

### *A. Low level processing*

The low level segmentation is carried out by the disk object and the disk functions derived from spline wavelet. At every disk location two wavelet functions are used: the quadratic spline wavelet scaling function and its half scale version.

$$\gamma_1(x,y) = \phi_s(x,y)$$
$$\gamma_2(x,y) = c_1\phi_{s/2}(x,y) - c_2\phi_s(x,y) \quad (20)$$

With the $\gamma_1$, $\gamma_2$ functions the $D_1(x,y,s)$, $D_2(x,y,s)$ disk transforms are calculated by (18). The $D_1$ can be used to detect region and $D_2$ to detect color or intensity variation. The $c_1$ and $c_2$ normalizing constants are adjusted that the $D_2$ disk transform is zero on a constant color region. Fig 8 shows the two disk objects. Based on the $D_1$ and $D_2$ values the node state can be set to *constant* or *variation* values. Depending on this state the graph branches to perform region or edge detection.

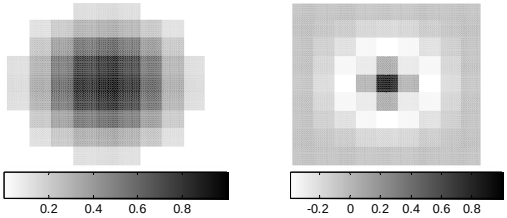


Fig. 8 Spline disk wavelets for segmentation, left: $D_1$, right: $D_2$

In order to detect line type features and regions two special graph structures are defined (Fig. 9). Bidirectional information flow is used for the central node, and unidirectional for all the others. If the previously defined network propagation algorithm (in section II) is applied to these sub-graphs then the graph expands horizontally. The result of this horizontal expansion is the segmentation of the whole image.

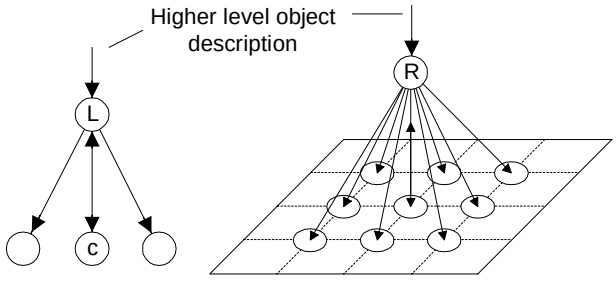


Fig. 9 Graph structures for line and region detection

This low level segmentation was tested on synthetic images containing different types of lines. Fig. 10 shows the disk behavior near a line if the position and the rotation parameter vary.

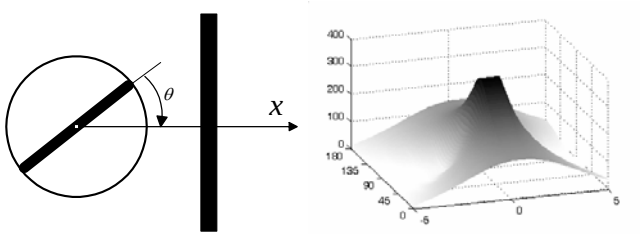


Fig. 10 Disk cost function dependence on position and rotation parameter

Since it is a monotone continuous function it is easy to find its local maxima. If one line segment is found, then the recognition proceeds further along the line. Similarly edge can be detected by a directional edge filter. Several types of edge or line profiles can be generated to detect different type of image components.

### B. High level processing

A simple document processing example is used to demonstrate the method. The task was to identify the components of circuit diagram images. Fig. 11 shows a computer generated simple circuit diagram.

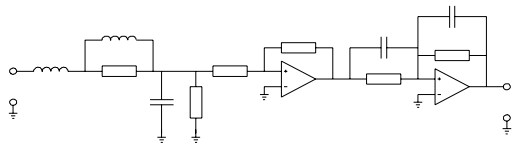

Fig. 11 Sample circuit diagram

A dictionary of the image elements is constructed by supervised learning. A sequence of images is created in a hierarchical way. First the lower level image elements are learned, and then based on the learned image bases more and more complex images are processed. Fig. 12 shows the hierarchically structured image element dictionary. The conditional probabilities are initially calculated from the object dictionary. With this initial probabilities circuit diagrams are processed and the values are updated. With the component dictionary and the conditional probabilities the graph of the circuit diagram can be generated. The identification is carried out by the Bayesian network belief propagation algorithm presented in section IV.

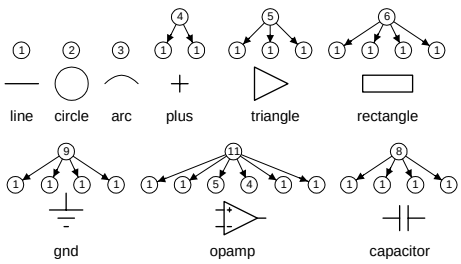


Fig. 12 The dictionary components for the sample circuit diagram

The graph of the circuit diagram is built up incrementally. Based on the node probabilities more and more component-graphs added to the graph of the circuit diagram. At the end of the simulation the highest probability nodes are identified as the components of the circuit diagram. Fig. 13 shows the number of unidentified nodes during the identification process.

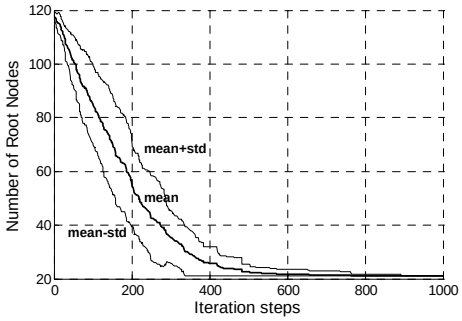


Fig. 13 The state of the network during identification

This demonstration shows that the presented method can be applied to integrate the low and high level object recognition steps.

### C. Real images test

The integrated method is also tested on real images. A wavelet ridge is used for the disk mask function. This disk object can be applied to detect the edges (Fig. 14, Fig. 15). The disk objects are optimized and a chain code of edge segments is generated. With higher level object description this chain code can be converted to higher level object description. This requires three-dimensional object definitions.

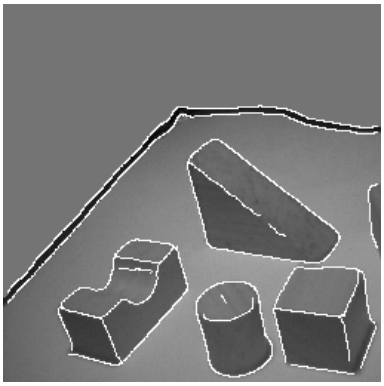

Fig. 14 The integrated method on a real image

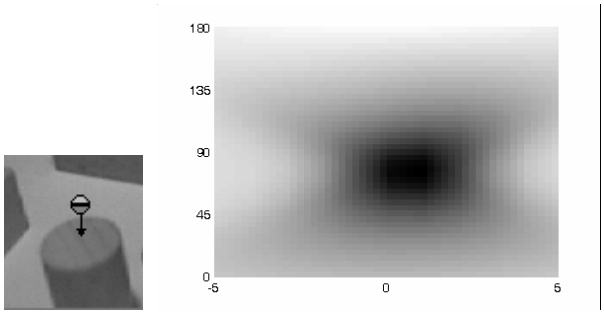

Fig. 15 The behavior of the disk transform near an edge

The method works well, since edge pieces can be identified and connected easier if a line or curve hypothesis exists. This demonstration shows the flexibility of the method. The same system works for the line detection and edge detection of real images with very little modification.

## VII. Discussion and Conclusion

The presented method is capable of performing both low and high level object recognition. The simulation shows that the same framework can be used for integrating low and high level object recognition. The individual steps of the object recognition can be performed properly if higher level object hypothesis are available. This can be achieved by storing the different levels of object descriptions in a hierarchical graph structure. The graph structure can be fixed or dynamically modified during the recognition. The advantage of the dynamic structures, that it can be adaptively adjusted to the input image. There are several ways to create a dynamic graph structure. In case of the human visual system the dynamic structure is created by moving the visual attention and focusing the eye. In our method the graph is built up adaptively, by adding small sub-graphs.

In the presented simulation only spatial relationships are used for object description. This is only sufficient to recognize simple objects. More complex object recognition requires symbolic object descriptions. The method with little modification can be also implemented for symbolic description. In the human brain higher level reasoning is the result of the combined effects of the neurons. This kind of behavior can be achieved by an upward and downward propagation on a hierarchical graph structure. The presented method is far from achieving this kind of system, but we believe that this type of research will reveal the true *theory of object recognition.*

## VIII. Acknowledgment

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*, 29, pp. 291-301, 2005

[2] Barta A., Vajk I, Processing Circuit Diagrams with Belief Network and Intelligent Agents., *Transactions on Information Science and Applications*, Issue 9, Vol. 2, September, pp. 1321-1329, 2005

[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] Mallat S., *A Wavelet Tour of Signal Processing*, Academic Press, 1999

[12] Burrus C. S, *Introduction to Wavelets and Wavelet Transforms*, Prentice Hall, 1998

[13] Freeman T.W., Adelson E.H., The Design and Use of Steerable Filters, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 13, No. 9, September, pp 891-906, 1991

[14] Sung J., Bang S.J., Choi S., A Bayesian network classifier and hierarchical Gabor features for Handwritten Numeral Recognition, *Pattern Recognition Letters*, 27. pp 66-75, 2006

[15] Deng S., Lati S., Regentova E., Document segmentation using polynomial spline wavelets, *Pattern Recognition*, 34. pp. 2533-2545, 2001

[16] Olshausen B.A, Anderson C.H., Van Essen D.C., A neurobiological model of visual attention and invariant pattern recognition based dynamic routing information, *The Journal of Neuroscience*, 13(11), 4700-4719, 1993