Computer Vision — CVWW'03, Ondřej Drbohlav (ed.)
Valtice, Czech Republic, February 3–6, 2003
Czech Pattern Recognition Society

[←]

Extraction of Geometric Features with Functional Models

Jocelyn Marchadier and Walter Kropatsch

Pattern Recognition and Image Processing Group(PRIP) Favoritenstrae 9/1832 A-1040 Wien, AUSTRIA,
mailto:jm@prip.tuwien.ac.at

Abstract *In this contribution, we show how to encode in a convenient and compact way information from images with so called functional graphical models. Those models permit to represent in a coherent mathematical framework features such as interest points , simply connected curves such as line and conic segments, connected regions of homogeneous properties, and compound features such as lines and line pencils. We show also the relevance of functional models for feature extraction, which is realized by minimizing a global model selection function. We apply the framework for encoding/extracting geometrical features thanks to two fast sub-optimal algorithms.*

1 Introduction

A feature is usually defined as some object that can be observed and/or measured, used as a model of certain aspect of reality. The operation of feature extraction consists in retrieving the parameters of features from a data source describing some sought aspect of the data. Feature extraction is an issue of image analysis, as structured features are often needed by complex image analysis tasks, such as image registration and matching, document analysis, camera calibration, and model reconstruction.

The structure of an image may be viewed as a composition of three different kinds of elements, describing the domain of the underlying continuous image. Regions are open sets where the image is continuous (in some adapted space), curve segments are open sets describing the shape of the linear discontinuities of the image in a continuous manner, and vertices must be given in complement for describing points adjacent to continuous curves (junctions, intersections, etc). Those elements "explain" local measurement variation onto an image, such as coordinates (for curves segment and their extremities) and intensity or color (regions). Such elements can also be grouped relating them according to some global constraints. Example of such features are lines (collinear connected line segments), line pencils (vanishing points), circles (composed of circular arcs), concentric circles, etc.

Authors usually propose specialized algorithms for the extraction of one feature type. For example, some authors proposed techniques of interest point detection, such as corners and multiple junctions [6]. Other authors proposed linear feature extraction techniques [4][10], resulting in features, without connection information. Vanishing point detection has also been under consideration [12]. Segmentation algorithms, which partition an image into regions of homogeneous properties, are numerous [9].

Relations such as perceptual and topological relationships are usually established afterward, with a lot of consistency problems. The perceptual grouping paradigm can solve this problem [15][10]. In another context, Fuchs and al. [5] proposed a multi-primitive extraction system where conflicts between features extracted independently are detected. A line drawing vectorization technique [8] consists in extracting line segments from a skeleton, which guarantees the consistency of the adjacency relations between the extracted line segments.

Our approach for feature extraction is based on the notion of functional graphical model that permits to represent functional dependencies between different types of features. Thus, this approach, opposed to the other, is global, as all information is related into a unique model. In section 2, we show how functional models can code measurements from images, and how their construction can be stated as a combinatorial optimization problem. In section 3, we propose two new fast and sub-optimal algorithms that permit the construction of a functional graphical model describing the geometry of contours of an image. In the last section (section 4), we demonstrate the proposed algorithm on an application.

2 Functional Graphical Models (FGM)

In this section, we firstly introduce the notion of functional models (section 2.1). We discuss the construction of models coding images (section 2.2), thanks to topological and functional models, and the construction of complex models enabling to take account of global information on an image (section 2.3). We then define the combinatorial optimization problem involved in its automatic construction from a model selection criterion (section 2.4).

2.1 FGM : definitions

A functional model is an implicit equation system relating measurements and parameters. We have introduced the notion of functional graphical models [14] (FGM for short), which may be used to design and analyze functional models via the functional relations between the involved variables.

Definition 1. *A* functional graphical model *is a couple $M = (V, F)$ with V a set of variables $V_i \in V$, and F a set of implicit equations $F_j \in F$ of the form $F_j = [f_j(V_{j_1}, ..., V_{j_{n_j}}) = 0]$, where $V_{j_i} \in V$ and f_j is a function of dimension d_j (the relations).*

A variable of the model is a vector describing parameters associated to some abstract object. Variables are thus referred as features of the model.

The *domain set* of the relation $F_j = [f_j(V_{j_1}, ..., V_{j_{n_j}}) = 0]$ is the set $Dom(F_j) = \{V_{j_1}, ..., V_{j_{n_j}}\}$. A FGM has an hypergraph structure, where vertices of the deduced hypergraph are the features, and hyperedges are the domain sets of the relations of the model.

The two following definitions define properly a set of models and a compound models. We will use these definitions later in the text.

Definition 2. *The* partial FGM *of a FGM $M = (V, F)$ generated by a family $E \subset F$ is a FGM (V_E, E) where $V_E = \cup_{F_i \in E} Dom(F_i)$.*

A partial FGM is directly related to the induced partial hypergraph of its graphical structure.

Definition 3. *The union of two FGMs $M_1 = (V_1, F_1)$ and $M_2 = (V_2, F_2)$ is the FGM $M = (V_1 \cup V_2, F_1 \cup F_2)$.*

The union of two FGMs produce a FGM. A FGM may be defined thanks to the union of an arbitrary number of FGMs.

FGMs can represent some information of the original image, such as the geometry of the discontinuities, or the variation of intensity on connected domains, and more generally the variation of some measure associated to each point of an image, leading to the definition of different specialized models that can be put together to explain interesting measures deduced from an image.

2.2 FGM of an image

In this section, we will give general hints for coding images with FGMs, and we give a concrete example for encoding the geometry of contours of an image.

A generic topological model in 2D is a topological map[2] defined as the decomposition of an orientable surface into three different disconnected sets :

- A finite set S of points.
- A finite set A of Jordan curves which extremities are elements of S.
- A finite set F of simply connected domains (the faces), which boundaries are unions of elements of S and A.

A coherent topological map embedded in the digital plane (called a contour map in the following text), can be obtained from well-composed gray value images[13]. Jordan arcs are then 4-connected digital curves, and faces are 4-connected components.

The measurements associated to each connected topological element can be described thanks to a model of vertex, a model of curve, or a model of region, and the topological map is attributed with those models. Fig. 1 shows the hypergraph structure of those three partial models. Each big disk represents a feature of the depicted model, and the relational equations are depicted as small disks, connected to the features involved in the relation by lines. The dashed arrows represent topological constraints that are discussed below.

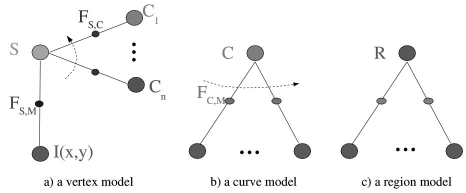

Figure 1: The three connected partial models

In our example, for simplicity, we only code the points belonging to the digital curves of a contour map computed from the modulus of the gradient of the original image. We choose the basic measurements to be coded, which are three dimensional vectors of the form $I(x, y) = (x\ y\ \theta)^t$, where x and y are integers and θ is the computed edge orientation at the considered pixel. This direction is supposed to be close to the normal vector of an object contour, and discriminates the real contours from the spurious ones, i.e. edges that do not compose the contours of smooth regions. Each pixel $I(x, y)$ of a measurement image is a feature of the FGM used for coding the geometry, related to some other features by functional relations.

A model of vertex (Fig. 1a) is composed of a feature $I(x, y)$ which is the measurement associated to an image point, a feature S that codes the information associated to a vertex, and features C_1, ..., C_n which are the parameters of the curve segments adjacent to the vertex. Two types of relations are involved in the model. A relation $F_{S,M}$ enables to deduce from the feature S the associated measurement $I(x, y)$. Other relations F_{S,C_i} link a junction S to the adjacent curves C_i, and do not need to be stored explicitly, as they are redundant with the relations of the topological model. For our application (coding the geometry of contours), we use a vertex feature $S = (x_S\ y_S\ \theta_S)$, and the following relations.

$$F_{S,M} = [S - I(i, j) = 0] \tag{1}$$

$$F_{S,C_i} = [distance(x_S, y_S, x_C, y_C) = 0] \tag{2}$$

where C is the feature associated to a curve adjacent to the vertex, x_C and y_C are the coordinates of the point on the curve C_i the closest to the point (x_S, y_S). Note that three equations are involved in the relation 1.

A curve model (Fig. 1b) is composed of a feature C which encodes the parameters of the continuous curve, and a digital set of points that form a digital curve in the measurement image. Each measurement from the set is linked to the curve with a relation. Ambiguities in the coding may occur when the continuous curve does not describe in a coherent manner an underlying digital curve, i.e. when the order of the points of the digital curve is not the same that the order of their closest points on the associated continuous curve. This constraint is illustrated on Fig. 2, where black figures represent the numbering of the points on the digital curve, whereas gray figures represent the numbering of their closest point on the continuous curve in correspondence.

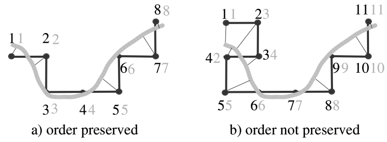

Figure 2: Order constraint on curves

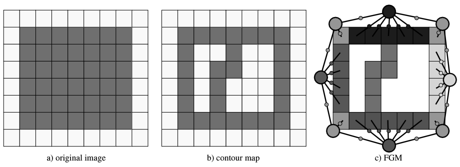

Figure 3: An image FGM example

For our application, the following relation is used :

$$F_{C,M} = \left[\begin{pmatrix} distance(x, y, x_C, y_C) \\ \sin(\theta_C - \theta) \end{pmatrix} = 0 \right] \quad (3)$$

where x_C and y_C are the closest points on the curve C to the point (x, y), and θ_C is the normal to the curve at point (x_C, y_C). For example, a line segment may be represented by a feature $C = (\theta_C \; d_C)$, where θ_C is the angle between the normal of the line and the x axis of the coordinate system, and d_C is the distance of the origin of the coordinate system to the line.

A connected region model (Fig. 1c) links each measurement of a region to the feature R that are parameters coding the function that enables to calculate the variation of the measurements on the region. Usually we are interested in coding intensity or color variations. In our application, we want to be able to detect/code spurious contours present in the contour map. We do not use a region model, and we need an independent point model that consists of all the measurements of spurious contours, without associated relation.

As demonstrated on Fig. 3, those models can be gathered into a FGM coding relevant information for the application. The complementary topological description can be used to describe implicitly domains where a model can be applied, reducing the amount of information necessary for its description.

2.3 Perceptual grouping and hierarchical FGMs

Perceptual grouping can be formalized with FGMs describing some global information, such as alignments of line segments and pencils of lines for geometry encoding. Clearly, parameters of collinear segments can be deduced from parameters of their supporting line. One of the two parameters coding a line can be deduced from the parameters of a line pencil it is participating. The two preceding examples leads to a FGM that has a tree structure (Fig. 4).

The alignment model is composed of relations involving the parameters of the line associated to each segment C, and the parameter of the line $L = (\theta_L \; d_L)$ describing the alignment.

$$F_{C,L} = [C - L = 0] \quad (4)$$

The pencil model involves alignment features and the pencil feature of form $P = (\theta_P \; i_P)^t$, where θ_P and $d_P = \frac{1}{i_P}$ are the polar coordinates of the vertex common to each line. This form is well adapted for parallel lines but fails to represent a pencil of lines crossing near the origin of the coordinate system. Then, the relation involved in the pencil model is :

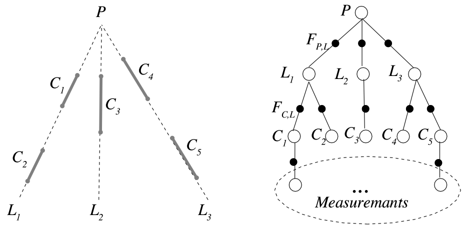

Figure 4: Model of a pencil of aligned line segments

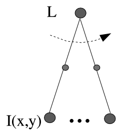

Figure 5: Reduced alignment FGM

$$F_{P,L} = [\cos\theta_P \; \cos\theta_L + \sin\theta_P \; \sin\theta_L - d_P \, d_L = 0] \quad (5)$$

When the relations are treated as constraints, the preceding models (alignment and pencil) can be simplified into compact models that are derived by parameter substitution, replacing simple implicit equations by complex ones, This is done by giving an explicit form of some other equation related by a model variable [14], which has as a consequence to suppressing the variable and the relation used. Substituted models can then be optimized in place of the overall hierarchical model, leading to more reliable parameters.

The reduced alignment model relates the feature L and each measure $I(x, y)$ thanks to the relation (3) :

$$F_{L,M} = \left[\begin{pmatrix} distance(x, y, x_C, y_C) \\ \sin(\theta_C - \theta) \end{pmatrix} = 0 \right] \quad (6)$$

Note that this model can be generalized to any implicit curve.

The reduced pencil model is composed of relations of form :

$$F_{P,L,M} = \left[\begin{pmatrix} \cos\theta_L(d_P x - \cos\theta_P) \\ + \sin\theta_L(d_P y - \sin\theta_P) \\ \sin(\theta_L - \theta) \end{pmatrix} = 0 \right] \quad (7)$$

Some structural constraints can be added in order to achieve a coherent digital/continuous mapping. An example of such a constraint is the order constraint of curve segment

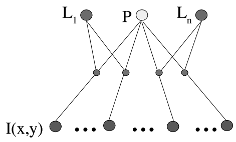

Figure 6: Reduced pencil FGM

models (section 2.2) generalized to curve models. Other generalizations can be achieved. For example, 3 pencils can be generalized by self-polar triangles [3] if they describe groups of orthogonal lines in space. Another example of hierarchical model can be given by the model constituted of circular arcs of concentric circles.

2.4 FGM selection : problem statement

The problem of constructing a model composed of all the previous models that explain an image can be put in a model selection framework by defining a cost function that evaluates the appropriateness of a model instance. Such a cost function may be, for example, the MDL which is widely used in computer vision [9][11]. Assuming that the structure of a FGM is known by both the coder and the decoder, and that all the relations are independent, the description length of a two part code of a standard FGM $M = (V, F)$ has the form :

$$k(M) = -\sum_{F_i \in F} log_2\left(p\left(F_i\right)\right) + g\ sizeof(Real) \tag{8}$$

where $sizeof$ returns the size (in bits) of the given type, $p\left(F_i\right)$ is the likelihood of F_i given the instances v of the variables, and g is the minimum number of reals that enable the calculation of all the variables of the model (the size of its generating set [14]). We will assume for simplicity that the residual of each equation involved in the model follow a normal law of given standard deviation, which enables to handle noise properly. Then the probability p may be computed for each equation from the quantiles of the normal law.

The code of a FGM M embedded on a topological structure contains the codes of all the partial models involves, and the code of the topological model. Its description length, for the previously defined partial model set, takes the following form :

$$k_T(M) = \sum_{M_P} k(M_P) + \sum_{M_L} k(M_L) + \sum_{M_V} k(M_V) + K_t + K_I \tag{9}$$

$k(M_V)$ is the minimum sized code of the vertex model M_V. In order to code a vertex model, we need to know the angle associated to the feature S. Moreover, if only one curve is adjacent to the vertex model, then we need to know another parameter, and $g = 2$. If the vertex is adjacent to more than one curve segment, its coordinates can be retrieved from the functional model and $g = 1$. $k(M_L)$ is the minimum sized code of the reduced curve model M_L, such that M_L is not a partial model of a pencil model. Two real valued parameters define a line ($g = 2$). $k(M_P)$ is the minimum sized code of the reduced pencil model M_P, and $g = 2 + n$, where n is the number of lines involved in the pencil model. K_I is the amount of memory needed to store each independent measurement, for which three real numbers are needed. $k(K_t)$ is the size of the topological model, which is considered fixed here. This size can nevertheless be computed when the topological structure is an attributed combinatorial map.

The most appropriate model assembly (i.e. FGM that is composed of union of partial models from the model set) given a measurement image is the one minimizing the length of the model code. This cost needs at least a two step optimization strategy. The first step consists in constructing a compound model. The second step is a classical estimation procedure that enables to determine the parameters maximizing the likelihoods of the relations.

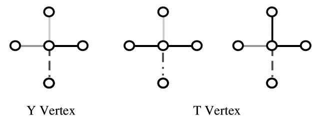

Figure 7: Local configurations of vertices model (up to rotations). The black lines of the T-vertex combinations correspond to a single curve model.

3 Sub-optimal FGM selection algorithms for feature extraction

In this section, we propose two new sub-optimal algorithms that retrieve complementary partial FGMs from a contour map. In the section 3.1, we present an algorithm for simultaneous connected curves and vertex model, based on a vertex relaxation strategy. In the section 3.2, we present a general strategy that can be used for construction independent partial models from a hierarchy, treating the model hierarchy level by level. The construction of the line pencil hierarchical model is illustrated.

3.1 Vertex relaxation for connected model retrieval

An optimization strategy well adapted for connected model optimization is relaxation. Relaxing a vertex model consists in constructing all possible combinations and models of curves adjacent to a vertex (i.e. all the possible vertex models), and calculating for each combination the global cost 9 of section 2.4. The local combination minimizing the global cost is kept for further processing.

Each vertex from the contour map is adjacent to at most 4 digital Jordan arcs, which are labeled by a curve model. Note that the digital curve associated to a curve model is composed by the concatenation of several Jordan arcs from the map. The following algorithm describes one vertex relaxation :

1. If a curve pass threw the considered vertex v, then we split the associated digital curve in two digital curves having v as extremity. In all other cases, the digital curves associated to adjacent Jordan arcs are used. Thus, in further processing, we use at most 4 digital curves having v as extremity.

2. We construct each possible local combination of digital curves (Fig. 7). For constructing T-vertex models, two digital curves from step 1 are concatenated. For each of these combinations :

   - For each digital curve of the combination, we construct all possible curve models, estimate their parameters, and compute their cost using equation 8 of section 2.4. The best local curve model is kept. Higher level models can be used (e.g. lines and line pencils) if hypothesis of such models have been generated.

- The parameters of the vertex are estimated thanks to the previously selected curve models involved in the combination.
- The cost of the constructed model is calculated.

The local model (curve+vertex) minimizing the sum of the cost calculated at the end of the second step is kept for further processing. Note that all the estimations and cost calculations are realized in a local way, although a correct treatment for the minimization would be to optimize the whole model, which is not feasible because of its size.

The initial solution can be constructed by polygonization of each Jordan arc of the computer map. The contour map is then transformed in such a way that its vertices are the extremities of the line segments constructed by the polygonization step. Another solution can be to consider each point of the contour map as a vertex. The relaxation step is done for each vertex of the initial solution, and overall algorithm is iterated until convergence, or until a maximum of iterations has been reached. The constructed curves are longer as the number of global iterations increases, leading to partial results that are already exploitable. Note that the algorithm does not converge in general.

The algorithm can be turned easily into a simulated annealing algorithm, which would converge slowly.

3.2 General layered model optimization

In this section, we propose a new algorithm for model selection that constructs from a set of features of known values $D = \{D_1, ..., D_n\}$ a set of models $M = \{M_1, ..., M_m\}$ such that each feature from D is involved in at most one model of M, and all the model of M have the form $M_i = (V_i, F_{M_i})$, with $V_i = \{D_j, D_j \in D\} \cup \{P_i\}$ and $F_{M_i} = \{[f(D_j, P_i) = 0]$, for all $D_j \in V_i\}$.

Examples of such models are the line and pencil model of section 2.3, which where used for validating the approach.

The proposed algorithm is a 4 steps sub-optimal algorithm minimizing the sum of the costs of the constructed models M_i (see section 2.4 for examples of cost functions) :

1. The *fuzzy clustering* step. This is realized by accumulating each datum from D in the parameter space of the function f by techniques such as the Hough transform [7]. The modes resulting of the accumulation are considered as the approximated parameters of potential models. Then, each datum is associated to models that are close enough to it (if any), and thus, overlapping partial models are build.

2. The *robust estimation* step. During that step, the parameters of each model constructed previously are determined with a robust estimation technique, such as RANSAC [1], which enable to determine more precise parameters for each model, excluding from the calculation outliers that were previously affected to the model.

3. The *relaxation* step. Each datum is being relaxed, in order to associate it to the more interesting model (if any), which is the one optimizing the model selection criterion. When the first datum is associated to a model, the parameters of the model are being taken into account for the cost calculation. Such a strategy enables to merge "close" models built with the same data. Structural constraints, such as the order constraint on curves can be validated in this step.

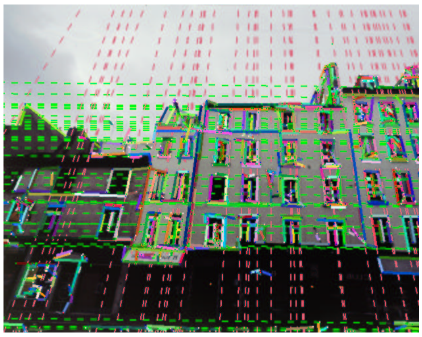

Figure 8: Line pencil extraction

4. The *global adjustment* step. The parameters of each non overlapping model previously generated may be estimated with a standard procedure involving all the data affected to each model. Eventually, the procedure may loop to step 3 for refinement, until the composed models are stable, or a minimum of the global cost is reached.

This method calculates a sub-optimal model. It takes advantages of two robust estimation paradigms, which are the Hough transform and the consensus estimation. It is computationally efficient thanks to the clustering step, which enable to construct quickly coarse groups of data, and enable also to obtain an approximation of the parameters of the model associated to each group. The parameters are refined thanks to the robust estimation step, and then can be used for approximating the selection cost for each partial model in the relaxation step.

4 Application

The parameters of each model is computed thanks to coarse estimator optimizing only, for convenience and efficiency, the geometrical parts of the models. Fig. 8 illustrates the result of the feature extraction framework on a real image. Line segments belonging to the same alignment have the same graylevel, and dashed lines are the lines participating to a pencil model. The algorithm was also evaluated on other images leading to comparable results.

For line models selection, the order constraint is maintained thanks to an interval tree structure. This algorithm enables the correct retrieval of very close lines almost parallel. Vertices extracted are adjacent to lines, lines to region which intensity is varying smoothly. Because we avoid to extract features independently from each other, they are more reliable, their parameters are known with a great precision, and the topological relations between the features are reliable, and can be used when we try to match to structures from two different images for example. Such a result is obtained by a single fast algorithm, as opposition with other methods which would use several step for building a less reliable equivalent structure.

For line pencils extraction, the clustering step has been ignored and we only seek the vanishing points corresponding to the vertical and horizontal lines. The robust estimation step is done thanks to the reduced model, with an adapted sampling technique. n lines are chosen, and p measures on each lines, with $n \geq 2$ and $p \geq 2$. The buckets of lines used are calculated with the moving center algorithm applied directly on the parameters of each line. This one constructs buckets of lines that are well spread on the image. The use of global features such as lines and line pencils restrict even more the space matches of simpler features as line segments and interest points, and give better approximations of the parameters of line segments and their extremities.

5 Conclusion

In this contribution, we have presented an image model based on a generic mathematical framework that can describe simple as well as complicated features and relations. We showed how this framework could be used for extracting features, defining the problem as a combinatorial optimization problem. In opposition to other approaches, our approach is global, involving different types of features and relations in a unique model, which construction involves the optimization of a unique and well established objective function.

We have shown how to apply the framework for geometric feature representation and extraction. Two sub optimal algorithms were proposed to solve different parts of the feature extraction problem. The first one construct a coherent model based on the simultaneous optimization of both curve models and vertex models. It can be easily extended to include also region models, which offer the advantage of taking account in the same algorithm all kind of features, as opposed to existing approaches. By doing so, we obtain different features that are topologically coherent with no further processing. The second algorithm build high level features, enriching the model in a coherent way. The result of the two step construction is a model that is, according to the used model set, the shortest representation of the data.

The presented framework can be extended in several ways: we can add other curve models, region models, or models of complicated features to the used model set; we can apply the framework for modeling images of higher dimensions; other problems than feature extraction can be solved with the framework of functional models such as complex shape/pattern recognition, or 3D reconstruction.

Acknowledgement

This paper has been supported by the Austrian Science Fund under grants P14445-MAT and P14662-INF.

References

[1] Bolles, R. C., Fisher M. A., "Random Sample Consensus : A Paradigm to Model Fitting with Applications to Image Analysis and Automated Cartography, Communication of ACM, **24(6)**, (1981), 381–395"

[2] Braquelaire, J.-P., Brun, L. "Image Segmentation with Topological Maps and Inter-pixel Representation", Journal of Visual Communication and Image Representation, **9(1)** (1998) 62–79

[3] Bhattacharya P., Rosenfeld A., Weiss I., "Geometric and algebraic properties of point-to-line mappings", Pattern Recognition 36 (2003) 483–503

[4] Danuser, G., Stricker, M., "Parametric Model Fitting : From Inlier Characterization to Outlier Detection", IEEE Trans. on PAMI, **20(3)** (1998) 263–280

[5] Fuchs, C., Frostner, W., "Polymorphic Grouping for Image Segmentation", Proc. 5th ICCV, (1995) 175–182

[6] Harris, C., Stephens, M., "Combined Corner and Edge Detector", Alvey88, (1998) 147–152

[7] Illingworth, J., Kittler, J., "A Survey of the Hough Transform", Computer Vision, Graphics and Image Processing, **44** (1988) 87–116

[8] Janssen, R. D. T., Vossepoel, A. M., "Adaptative Vectorization of Line Drawing Images", Computer Vision and Image Understanding, **65(1)** (1997) 38–56

[9] Leclerc, Y. G., "Constructing Simple Stable Descriptions for Image Partitioning", International Journal of Computer Vision, **3** (1989) 73–102

[10] Lee, J.-W., Kweon, I.S., "Extraction of Line Features in a Noisy Image", Pattern Recognition, **30(10)** (1997) 1651–1660

[11] Lindeberg, T., Li, M.-X., "Segmentation and Classification of Edges Using Minimum Description Length Approximation and Complementary Junction Cues", Tech. Rep. ISRN KTH/NA/P–96/01–SE, Jan. 1996.

[12] Lutton, E., Maitre, H., Lopez-Krahe, J., "Contribution to the Determination of Vanishing Points using Hough Transform", IEEE Trans. on PAMI, **16(4)** (1994) 430–438

[13] Jocelyn Marchadier, Didier Arqus, Sylvain Michelin, "Thinning Grayscale Well-Composed Images: A New Approach for Topological Coherent Image Segmentation", Lectures Notes On Computer Science no. 2301, DGCI 2002, 360–371

[14] Jocelyn Marchadier, "Functional Graphical Models", Technical Report 76, Vienna University of Technology (2002), 24 pages

[15] Mohan, R., Nevatia, R., "Using Perceptual Organization to Extract 3-D Structures", IEEE Trans. on PAMI **11(11)** (1989) 1121–1139