Extraction of Curve Segments and Junctions by Graph-based Optimization [1)]

Jocelyn Marchadier and Walter Kropatsch
Pattern Recognition and Image Processing Group(PRIP)
Favoritenstraße 9/1832 A-1040 Wien, AUSTRIA,
jm@prip.tuwien.ac.at

Abstract:
A classical problem in image analysis is the extraction of primitives such as junction points or line segments, which are used in pretreatment for many computer vision tasks. In this contribution, we propose a new method for extracting primitives from raster graphs, which are graphs of digital points typically obtained from skeletonization algorithms. The problem is posed as a combinatorial optimization problem. This problem is solved using a simple and efficient algorithm that builds a continuous description of a raster graph, which relies on the simultaneous use of different kind of features : junctions, and curve segments. The proposed method can be opposed to existing ones which relies generally on ad hoc criteria, and are specialized for the extraction of one type of primitive.

1 Introduction

A structural primitive is usually defined as some object part that can be observed and/or measured as a unit, used as a model of certain aspect of reality. The extraction of structural primitive (feature) consists in retrieving the parameters of primitives from a data source describing some sought aspect of the data. Primitive extraction is an issue of image analysis, as structural primitives are often needed by complex image analysis tasks, such as image registration and matching, document analysis, camera calibration, and model construction.

Authors usually propose specialized algorithms for the extraction of one primitive type. For example, some authors proposed techniques of interest point detection, such as corners and multiple junctions using a measure on each point of the image [6][3]. Other authors proposed curve segments extraction techniques. Line segments are generally considered. Existing techniques range from statistical testing approaches [2], the well-known Hough transform [7], or ad hoc methods, based for example on perceptual grouping [9]. Other methods are adapted to

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

other kind of primitives, such as ellipses [10]. Existing methods usually result in disconnected curve segments. Moreover, they are usually ad hoc, in the sense that there is no optimized criterion, as they use some statistics which modes indicate the presence of the sought feature.

Adjacency relations between primitives as curve segments or junctions are usually established after the specialized extraction processes, with a lot of consistency problems. The perceptual grouping paradigm can solve this problem [9]. In another context, Fuchs and al. [4] proposed a multi-primitive extraction system where conflicts between primitives extracted independently are detected. There again, the used criterion is either ad hoc, or statistically based, which results in topological information which is not reliable, and which is hard to use. 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. Another interesting approach was presented in [1], where edges of a run graph are contracted until stability, the vertices of the resulting minimal graph are the junctions and the line ends connected by edges corresponding to connections in the image.

Our approach for primitive extraction is based on the notion of functional graphical model, which can be viewed as systems of constraints representing functional dependencies between different types of primitives. Thus, this approach, opposed to the other, is global, as all information is related to a unique model. Moreover, our technique can manage to extract simultaneously junctions, lines, ellipses and other implicit curves of higher order. In section 2, we show how functional models can code attributed raster graph describing geometric information, and how their construction can be stated as a combinatorial optimization problem. In section 3, we propose a new fast and sub-optimal algorithm that permits the construction of a continuous geometric model from an attributed raster graph. In the last section (section 4), we demonstrate the relevance of the proposed approach on examples.

2 Functional Models of Raster Graphs

A functional model is a system of implicit equations relating measurements and parameters. We have introduced the notion of functional graphical models [12] (FGM for short), which are constraint systems that may be used to design and analyze functional models via the functional relations between the involved variables.

A *variable* V represents some primitive. It is a symbol describing a real-valued multi-dimensional vector coding parameters of the underlying primitive. An *instance* v of a variable V is a constant real-valued vector specifying a particular set of parameters.

A *functional constraint* is a $n+1$-uple $(f, V_1, ..., V_n)$, where $V_1, ..., V_n$ are variables, and f is a function which domain is $V_1 \times ... \times V_n$, and which takes values in $\mathbb{R}^d$. A functional constraint $F = (f, V_1, ..., V_n)$ is *satisfied* for instances $v_1, ..., v_n$ of the variables $V_1, ..., V_n$ iff $f(v_1, ..., v_n) = 0$.

A *functional graphical model* (FGM) is a couple $M = (\mathcal{V}, \mathcal{F})$ with $\mathcal{V}$ a set of variables and $\mathcal{F}$ a set of functional constraints $F_j \in \mathcal{F}$ involving variables from $\mathcal{V}$. The *union* of two FGMs $M_1 = (\mathcal{V}_1, \mathcal{F}_1)$ and $M_2 = (\mathcal{V}_2, \mathcal{F}_2)$ is the FGM $M = (\mathcal{V}_1 \cup \mathcal{V}_2, \mathcal{F}_1 \cup \mathcal{F}_2)$. 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.

2.1 Geometric FGM of an Attributed Raster Graph

A raster graph is a graph of digital points that can be obtained from a skeleton or a homotopic kernel of gray value images [13, 11]. A raster graph can be built from the modulus of the gradient of a graytone multilevel image, describing contours of the original image. We define an $((n = 4, 8)$-connected) attributed raster graph as a 3-tuple $G = (\mathcal{P}, \mathcal{A}, w)$, where $\mathcal{P} \subset \mathbb{Z}^2$ is a set of digital points, $\mathcal{A}$ is defined as $\forall (P_1, P_2) \in \mathcal{A}, P_1 \in \Gamma_n(P_2)$ with $\Gamma_n(P_2)$ the $(n = 4, 8)$-neighborhood of P_2, and $w(P_i)$ is a function associating to a point $P_i \in \mathcal{P}$ a real-valued measurement vector relevant for encoding underlying geometry of G. In the following text, we use measurements of the form $w(P_i) = (x_i \; y_i \; \theta_i)$, where x_i and y_i are cartesian coordinates associated to P_i, and θ_i is the direction of the gradient of the original image, which is supposed to be perpendicular to the direction of the contour.

FGMs can encode in a convenient computational way the geometric primitives approximating the measurements from an attributed raster graph. This representation may be decomposed by three kind of FGMs : models of curve, models of junction, and a model of outliers, taking into account points not classified in other models. The curves encoded here are implicit curves.

A curve model is an FGM $(\{C, w(P_1), ..., w(P_n)\}, \{F_{C,1}, ..., F_{C,n}\})$ with C being a variable which encodes the parameters of the continuous curve, $(P_1, ..., P_n)$ an ordered sequence of vertices of G, and $F_{C,i}$ the functional constraints :

$$F_{C,i} = \left[\begin{pmatrix} f(x_i, y_i, C) \\ \sin\left(\arctan\left(\frac{\delta f}{\delta y}(x_i, y_i, C) / \frac{\delta f}{\delta x}(x_i, y_i, C)\right) - \theta\right) \end{pmatrix} = 0 \right]$$

where f is an implicit function describing the geometry of the curve. An example is a line segment model. A line would be encoded by the vector $L = (\theta \; d)^t$ where θ is the angle between the normal of the line and the x-axis of the coordinate system, and d is the distance of the

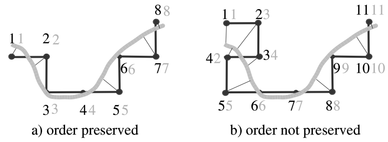


Figure 1: Order constraint on curves

line to the origin. The constraint involves the function $f(x_i, y_i, L) = x_i \cos\theta + y_i \sin\theta - d$.

A model of junction $(\{S, w(P), C_1, ..., C_n\}, \{F_{S,w(P)}, F_{S,C_1}, ..., F_{S,C_n}\})$ with $S = (x_S\ y_S\ \theta_S)$ that encodes information associated to a junction point (θ_S being the edge direction associated to S), $w(P)$ a measurement affected to the model, C_1, ..., C_n the parameters of the curve segments ending at or passing threw P. The two types of relations involved in the model are :

$$F_{S,w(P)} = [S - w(P) = 0]$$
$$F_{S,C_i} = [f(x_S, y_S, C_i) = 0]$$

where f is the implicit function associated to the model of C_i. For an endpoint (involving a unique curve), either x_S or y_S need to be stored. For the other junction models (two or more adjacent curves), neither x_S nor y_S need to be stored.

A curve segment C is encoded by a path $(P'_1, P_1, ..., P_i, P'_k, P_{i+1}, ..., P_n, P'_m)$ of G, where $(P_1, ..., P_n)$ is the ordered sequence associated to the curve model of C, the vertices P'_i are associated to junctions models involving C (with P'_1 and P'_m the extremities of C, and the points P'_k are associated to T junction models). For avoiding certain pitfalls of this mapping, we impose also that $\forall i < n, t_i < t_{i+1}$, where t_i is the parametric coordinate of the point closest point to P_i (the standard euclidean distance where used) on the path associated to C. This constraint is illustrated on Fig. 1, 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.

The outlier model is the FGM $(\{w(P_1), ..., w(P_n)\}, \emptyset)$, where $w(P_i)$ are the measurements not associated to a curve or a vertex model. $P_1, ...P_n$ are isolated vertices.

2.2 Description length of geometric FGM

A description composed of the three models encoding an attributed raster graph can be built by defining a cost function that evaluates the appropriateness of a model to instances of its variables. Such a cost function may be, for example, the MDL ([5]) which is widely used in computer vision (eg. [10]). Assuming that the structure of a FGM is known bythe coder and the decoder, and that all the relations are independent, an approximation of the optimal description length of a two part code of a standard FGM $M = (\mathcal{V}, \mathcal{F})$ has the form :

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

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 [12]. We assume for simplicity that the residual of each equation involved in the model follow a normal distribution 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 distribution.

The code of an FGM M describing the geometry of a contour graph can be constructed by tabulating the codes of junction, curve and noise FGM, and the code of an attributed graph describing the topology of the continuous model (its arcs are labeled by a curve model, and its vertices are labeled by a junction model). An approximation of the optimal description length is then given by the global cost :

$$k_T(M) = \sum_{M_C} k(M_C) + \sum_{M_V} k(M_V) + K_t + k(M_N) \tag{1}$$

$k(M_V)$ is the minimum sized code of the junction model M_V. $k(M_C)$ is the minimum sized code of the curve segment model M_C. $k(M_N)$ is the amount of memory needed to store each outlier. $k(K_t)$ is the size of the labeled graph, which is considered fixed here.

We consider that the union of models respecting the constraints most appropriate to a set of measurements 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.

3 Relaxation for curve segments/junctions extractions

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 junction models), and calculating for each combination the global cost 1. The local combination minimizing the global cost is kept for further processing. As each vertex from a 4-connected raster graph is adjacent to at most 4 curve models, the number of combinations is small (6 for a 4-connected raster graph) and the algorithm is computationally efficient.

For each vertex of the graph, we apply the following sub-iteration. We start from the model computed so far (eg. Fig. 2.a).We construct each possible local combination of digital curves (Fig. 2.b). For each of these combinations :

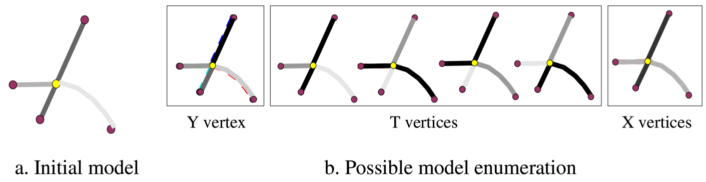


Figure 2: Sub-iteration of the relaxation algorithm.

1. For each digital curve of the combination, we construct all possible curve models, estimate their parameters, and compute their cost using equation 2.2. 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.

2. The parameters of the vertex are estimated thanks to the previously selected curve models involved in the combination.

3. The cost of the vertex 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. 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 relaxation step is done for each vertex of the initial solution, and the 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 initial solution can be constructed by polygonization of each Jordan arc of the raster graph. For efficiency reasons, we used the extremities of the graph obtained by polygonization as the vertices being relaxed.

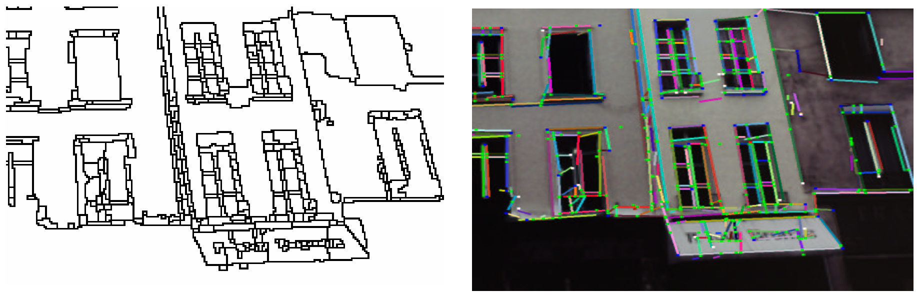
Alternatively, the algorithm can be turned into a simulated annealing algorithm.

4 Application

Several curve primitives (and associated implicit functions) may be used for curve models. Results given here demonstrate the use of line and ellipse segments. The parameters of each model are computed thanks to a coarse estimator optimizing only, for convenience and efficiency, the geometrical part of the models. The equations involving the gradient of the measurements are taken into account for cost computation.

Fig. 3 illustrates the extraction of junctions, line segments, and ellipse segments on a synthetic

Figure 3: Simultaneous extraction of junctions, line and ellipse segments



a) labeled raster graph b) reconstructed junctions and line segments

Figure 4: Junctions/Line segments extraction on an architectural image

image. The five curves are correctly retrieved using the MDL criterion, as shown on the right hand side of the figure. Our experiments where realized using a linear ellipse estimator, which is not suited due to the inclusion of bias at points of high curvature. On more complex graphs, the ellipse model involving orthogonal distances from points to the ellipse was found to be very to slow.

Fig. 4 illustrates the result of the feature extraction framework on a real image particularly well-suited to demonstrate the use of structural features such as lines. The original raster graph is diplayed on Fig. 4.a. Its continuous approximation computed is displayed on Fig. 4.b. We can find number of correct topological relations. The noisy graph and the non preservation of topology by digitalization explains partly some of the remaining problems. For the whole image, the number of real-valued parameters needed to encode the geometry with lines is about ten times smaller than the number of parameters required to encode each measurement.

The algorithm was also evaluated on other images leading to comparable results. The method enables to extract junctions and line segments in a reliable manner. These results are explained by the fact that it uses the figural continuity of contours.The algorithm is efficient as the time order of the computation is of 1-2 minutes without special optimization (for 10 iterations of relaxation, with images of 700x500 pixels, on a pentium III at 900 MHz). We have tried also a second order relaxation scheme, for which two extremities of a curve are being simultaneously relaxed. The results obtained were comparable, but the computing performances decreased.

5 Conclusion

In this contribution, we presented a new multi-feature extraction algorithm based on the optimization of a distance length function of a constraint system based on implicit equations. The algorithm constructs a coherent model by simultaneously optimizing both curve models and junction models. Results including junctions, line segments and ellipses models have been presented. The remarkable fact is that good results have been obtained by compressing the measurements with well-chosen functional models. The algorithm can be 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. Models of other features are under study, as well as other combinatorial algorithms. The framework can also be used for modeling images of higher dimensions.

References

[1] Burge M.., Kropatsch W.G., "A Minimal Line Property Preserving Representation of Line Images", Computing, vol. 62 (1999) 355–368

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

[3] Deschênes F., Ziou D., "Detection of line junctions and line terminations using curvilinear features", Pattern Recognition Letters, vol. 21 (2000) 637–649

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

[5] Grünwald P., "Model Selection Based on Minimum Description Length", J. of Mathematical Psychology 44, (2000) 133–152

[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, vol. 44 (1988) 87–116

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

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

[10] 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.

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

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

[13] Pierrot Deseilligny M., Stamon G., Suen C., Veinerization : A New Shape Description for Flexible Skeletonization, IEEE Trans. on PAMI, vol. 20(5) (1998) 505–521