RECOGNITION OF ROAD AND RIVER PATTERNS BY RELATIONAL MATCHING

Norbert Haala and George Vosselman
Institute of Photogrammetry, Stuttgart University
Keplerstraße 11, D-7000 Stuttgart 1, Germany
ISPRS Commission III

ABSTRACT:
This paper discusses a procedure aiming at the automatic exterior orientation of images. To this purpose the relational matching method is used to match relational descriptions of images and maps. Because roads, rivers and land parcels often constitute unique structures, these topographic features are taken as the basic elements of the descriptions and are used to identify and locate landmarks like road crossings, waterway junctions and specific parcel structures. The structural descriptions of the images are obtained by thresholding selected channels of colour images and subsequent thinning of the linear structures. Tree search methods are used to match the derived relational image descriptions with hand made descriptions of the landmarks.

KEYWORDS: Artificial Intelligence, Image Interpretation, Feature Extraction, Pattern Recognition

1 INTRODUCTION

In recent years it has been shown that several important photogrammetric tasks, like the relative orientation of images, the aerial triangulation and the derivation of digital terrain models, can be automated with digital image processing techniques [Schenk *et al.* 1991, Tsingas 1991, Ackermann and Krzystek 1991]. The main concern in solving these tasks is to establish correspondences between (patches of) the overlapping images. Using the area based or feature based correspondence algorithms that have been developed over the last decade, homologous points indeed can be found.

Another group of tasks, including the exterior orientation of images and mapping of images, also is a research topic, but the progress in the automation of them is much slower. Like the ones mentioned above, these tasks also have to be solved by determining a correspondence. However, this is not a correspondence between two images, but a correspondence between an image and a model of the contents of this image. E.g., for the exterior orientation of an image one has to determine a match between image patches and models describing the control points. The automation of the mapping process involves a comparison of the image with generic models that define the expected appearances of the roads and houses in the image. The need to model the image contents makes these tasks relatively hard to automate.

This paper deals with the automatic exterior orientation of images by matching images to descriptions of natural control points (landmarks). Compared to the mapping task, this problem has the advantage that one can select the landmarks one wants to measure. I.e., those landmarks can be utilized that are relatively easy to model and easy to recognize.

The landmarks we use for the orientation are described by relational descriptions. The reason for this choice is twofold. First, the description has to be feature based, because area based descriptions (i.e. grey values) would depend on the season and the weather conditions which are difficult to model. Second, a description by features only usually does not contain enough information to recognize a landmark, because the approximate values of position and rotation that can be obtained from the flight plan are very inaccurate. I.e., the image patch in which a landmark has to be found will contain many features that do not belong to that landmark. The risk of matching the wrong features is therefore relatively high. This risk can be reduced by using the structural information that is contained in the relationships between the features. This structural information can well be represented in relational descriptions.

In aerial photographs such structural information is present in landmarks like road crossings, river junctions and land parcels. The relational descriptions we use in the matching step therefore consist of roads, rivers and parcel boundaries and their topological and geometrical relations. So, the problem of recognizing a landmark is defined as the problem of matching a relational description of an image patch to the relational description of the model of the landmark. This problem can be solved with the relational matching method [Shapiro and Haralick 1981]. In contrast to the usual least squares methods, this method does not require approximate values for the position or the orientation.

The next section describes the extraction of the relational descriptions from the colour images. The relational descriptions of the landmark models were obtained by digitizing maps. Section 3 deals with the evaluation of the correspondences. The task of the matching algorithm is to find the best mapping between the features of the image and the features of the model. To this purpose one needs a quantitative evaluation measure that describes the quality of the mappings. With the tree search methods, that will be described in section 4, one then can select the best mapping. Throughout the paper the different processing steps are illustrated by an example of the location of a road junction. Section 5 shows and discusses the results on the location of this and five other landmarks.

2 STRUCTURAL DESCRIPTION

In order to get a comparable representation of the image object and the landmark, structural descriptions were extracted from the image and the landmark. They describe the selected image objects and the landmark model in terms of geometric primitives (points, lines, and regions) and their relations. To obtain an expressive description, objects containing sufficient structure like roads, rivers and cornfields were used. The relational image description was derived automatically from a colour image in two steps. First, an appropriate band of a colour image was selected to compute a binary image by classifying pixels that belong to the objects of interest. Secondly, the binary image was vectorized by a contour tracing, respectively a line tracing algorithm, and the relational description was extracted.

The structural descriptions of the landmarks were obtained by digitizing maps, but, in principle they could also have been derived from a geographic information system (GIS).

Figure 1: Example street1: intensity image (top left), hue image (top middel), binary image (top right), extracted image lines (bottom left), map (bottom middle), landmark model (bottom right)

2.1 Classification

Two different image types were used: true colour images in scale 1:8000, in which we classified grain fields, and near infrared images in scale 1:6000, in which we extracted streets in rural areas and water surfaces. The images were digitized by scanner resulting in red, green and blue images (RGB). These RGB images were transformed into a hue, saturation and intensity (HSI) representation. The hue channel was used for classification. Figure 1 top middle shows an example of a hue image, which was calculated from a RGB image using the equation below [Frey 1990].

$$H = \arccos\left[\frac{0.5\cdot((R-G)+(R-B))}{\sqrt{(R-G)\,(R-G)+(R-B)\,(G-B)}}\right]$$

The only exception was the water surface, which was classified by thresholding the red channel.

A region growing was performed on the used band before classifying the pixels in order to stabilize the results and to avoid small regions. Pixels belonging to the object surface were extracted by thresholding the average hue values of the extracted regions. The thresholds were fixed in advance, according to the object and image type. Although the images had poor spectral quality, the results of the classification showed to be robust to changes in the thresholds.

2.2 Line extraction

The thresholding was followed by a line and node extraction to transform the raster image (figure 1 top right) into a vector image. In the cornfield example this was done by tracing the contours of the selected regions. For images with linear objects, like the roads and rivers, a thinning algorithm [Arcelli and Baja 1985] was used to obtain a skeleton of the selected objects. Figure 1 bottom left shows a line image, derived from the skeleton image by a line following algorithm. In this step lines, nodes and enclosed regions were extracted. Features caused by image noise like nearby nodes representing the same points, close parallel lines, and short lines were eliminated afterwards.

The relational description of the landmark (figure 1 bottom middle) was obtained by digitizing a map (figure 1 bottom right).

2.3 Primitives and relations

A structural description consists of a primitive part and a relational part. The primitive part contains geometric primitives like points, lines, and regions, which represent the object parts. The primitives are characterized by a set of attribute values like line length, line type or region size. E.g., a straight line of length 30.2 is represented by the primitive p_1 : {(length 30.2) (shape straight)}. The second part describes the interrelationships between these primitives. Possible relations are angles between lines, connections between points and lines and between lines and regions. These can be characterized by attribute values, too. E.g., an angle between lines p_1 and p_3 is represented by the relation tuple r_1 : $\{p_1\, p_3\, (\text{angle}7°)\}$.

Following [Shapiro and Haralick 1981], we use the symbols $D_1(P,R)$ and $D_2(Q,S)$ for the image and the landmark description, respectively . P represents the set of primitives p_i, Q represents the set of primitives q_j. R and S are the set of relation tuples r_i and s_j.

3 EVALUATION OF THE MAPPING

In order to match the image description $D_1(P, R)$ to the landmark description $D_2(Q, S)$, it is necessary to find a mapping h between the primitives of set P and Q. Because we want to select the best mapping from a large set of possible mappings $h : P \rightarrow Q$, a measure has to be defined, which evaluates the quality of a mapping between the two descriptions D_1 and D_2. Intuitively, the evaluation of a mapping should depend on the similarity of the attribute values of the corresponding primitives and relations. Beside that, the frequency of the attribute values in the descriptions should also be taken into account. The fit of primitives and relations with rare attribute values should be more important to the overall measure of similarity, than the fit of primitives and relations with very frequent attribute values.

A measure, that satisfies these demands and has some other nice properties, too, is provided by information theory.

3.1 Mutual information

The mutual information $I(a(p_i); (a(q_j))$ between the attributes a of two primitives p_i and q_j is defined as the difference of the self information $I(a(q_j))$ and the conditional information $I\left(a\left(q_j\right)|a\left(p_i\right)\right)$:

$$I(a(p_i); (a(q_j)) = I(a(q_j)) - I\left(a\left(q_j\right)|a\left(p_i\right)\right)$$

The self information is used to measure the rareness of an attribute. The self information of a (discrete) attribute a depends on the frequency of the value v of that attribute in the description. If it is very frequent, there is a high probability $P(a = v)$ of an attribute taking this value. According to [Shannon and Weaver 1949] the self information of an attribute value is defined by

$$I(a = v) = -\log P(a = v)$$

So, attributes with frequent values contain only little self information, attributes with rare values contain much self information.

The conditional information is a measure of similarity between attributes of primitives or relations. It depends on the probability that the corresponding primitive of a primitive with attribute value v_1 in description D_1 will take a certain value v_2 in description D_2. The conditional information of the attributes $a(q_j)$ and $a(p_i)$ is defined by

$$I\left(a\left(q_j\right)|a\left(p_i\right)\right) = -\log P\left(a\left(q_j\right) = v_2|a\left(p_i\right) = v_1\right)$$

The same equation holds for the conditional information between attributes of relation tuples.

These so-called transition probabilities $P\left(a\left(q_j\right) = v_2|a\left(p_i\right) = v_1\right)$ between the attribute values have to be estimated, computed or otherwise supplied before a mapping can be evaluated.

As the evaluation measure we use the mutual information of a mapping between two relational descriptions $I_h(D_1; D_2)$. This is computed by summing up the mutual information between the attributes of the corresponding primitives and relations. Because the mapping h defines which primitives and relations correspond, this mutual information depends on the mapping h.

3.2 Properties of mutual information

The amount of mutual information between primitives ranges from positive values (rare and similar attribute values) to negative values (unlikely correspondences). Two primitives can not be matched if it is impossible that one of their attribute values correspond. In that case, the conditional information between the primitives is infinite, and therefore the mutual information of that mapping becomes minus infinite.

Beside the fact that the mutual information is an intuitively satisfying measure, it also can be shown that it has some nice properties (cf. [Vosselman 1992])

- Not all primitives of the image description can be matched to primitives of the landmark model. This is the case if image primitives represent objects or object parts which are not contained in the description of the landmark model. It may also be due to errors in the image segmentation. These unmatched primitives are then mapped to a so-called wildcard (or nil-label). In contrast to other similarity measures our method can evaluate a wildcard mapping very easy. A wildcard assignment has a mutual information of zero, because a wildcard provides no information about the primitive it is assigned to.
- Mutual information is a symmetric measure. Matching D_1 to D_2 therefore gives the same result as matching from D_2 to D_1.
- Under the assumption of attribute independence, the mapping with the highest amount of mutual information is the maximum likelihood estimate.

Using the concept of mutual information, one can define the best match between the landmark description and the image description as the mapping with the highest amount of mutual information between the attributes of the corresponding primitives and relations.

4 SEARCHING THE BEST MAPPING

Without any a priori knowledge about the spatial transformation between the landmark model and the image, all mappings between the primitives of set P and the primitives of set Q have to be considered. The first part of this section describes the representation of the search space in a so-called search tree. The second part describes strategies to find the best mapping in the defined search space. Rules of thumb, that help to reduce the search time are discussed in the last part of this section. The methods used here have been developed in the field of Artificial Intelligence (see e.g. [Nilsson 1982]).

4.1 Representation of the possible mappings

Figure 2 shows a search tree, which consists of nodes (the black dots) and arcs (the lines connecting the nodes). Each level of the tree is associated with a single primitive p_i. Each node represents an assignment of a primitive of set Q to a primitive of set P. Primitives p_i are called units, primitives q_j are called labels. The nodes are connected by arcs to the upper and lower level of the tree. Each node can have only one predecessor, the parent node, but several successor nodes.

A sequence of connected nodes is called branch and represents a partial mapping between subsets of P and Q. E.g., the branch of figure 2, shown in thick lines, represents the partial mapping $p_1 \rightarrow q_3$, $p_2 \rightarrow q_5$, $p_3 \rightarrow q_2$. The numbers at the arcs represent the merit (i.e. the mutual information) of the assignment at the following node. This information depends on the primitive attributes as well as on the compatibility of the assignment at the following node to the assignments on the path from the root node to that node. Summing up the information along the arcs at a branch gives the overall mutual information of that branch. Each path from the root node to a node at the lowest level of the tree (a leaf node) is a possible mapping $P \rightarrow Q$.

4.2 Tree search methods

Due to the exponential complexity of the correspondence problem, it is impracticable to search the best mapping by simply

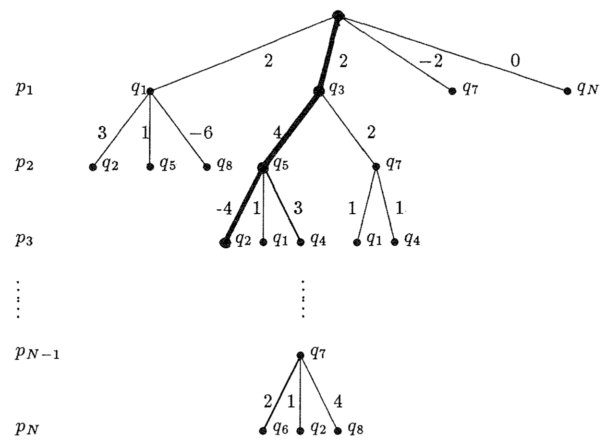

Figure 2: Search tree representation of a correspondence problem

trying all possible combinations between the primitive sets. Several search algorithms have been developed, performing a more efficient scanning of the tree [Nilsson 1982].

A tree search always starts at the root node of the tree. At this point no assignments have been made. The root node is expanded by generating the successor nodes. These nodes represent the possible assignments between the first unit primitive and admissible label primitives. A label is admissible if it has not been used before and if the mutual information between the unit and that label is not minus infinite. One of the generated successor nodes is inspected and expanded afterwards. The search space (defined by the search tree) is examined by visiting and expanding the nodes of the tree until a solution has been found.

A simple method to determine the order of visiting the nodes in the tree is called the depth first method. The depth first method tries to move to the bottom of the tree as fast as possible by always expanding the node on the deepest level of the tree. Scanning the tree of figure 2 in a depth first manner one would start by selecting the first successor node of the root node. This node represents the assignment of label q_1 to unit p_1. Because this node now is the deepest one, the next step is to add the node which e.g. represents the assignment of q_2 to p_2 to this branch. If there is no admissible label primitive q_j left, a node can not be expanded any further. The search then moves up to the parent node and moves down again along one of the other branches of that node. This procedure is called backtracking.

The depth first method has some serious disadvantages. An assignment, leading the search into a part of the tree where no solution will be found, may appear at a very high level. Nevertheless, all nodes below this assignment will be inspected until a new assignment at that level is tried. E.g., if the depth first search expands the node, which assigns label q_1 to unit p_1 in figure 2 all nodes in the sub-tree below this node will be visited until the search reaches this level again. Beside that, one also has to search the complete tree, because one does not know whether the first path to a leaf node that has been found represents the best solution. A more intelligent strategy has to be used.

Always expanding the most promising node would immediately lead to the optimal solution. To this purpose the function $f(n)$ is defined representing the mutual information of the best path through node n [Pearl 1984]. The merit $f(n)$ depends on the merit $g(n)$, collected on the branch from the root node to node n and the future merit $h(n)$, which will be collected if this branch is continued to a leaf node. Of course, it is impossible to compute the future merit $h(n)$ exactly. However, it can be estimated roughly by calculating the mutual information of all unit to label assignments which are left, without checking these assignments for consistency. This value $h^*(n)$ is used to calculate an estimate of the possible mutual information $f^*(n) = g(n) + h^*(n)$ at every node n. Leaving out the consistency check will always cause an overestimation of the future merit and therefore $f^*(n)$ is always greater than the true value $f(n)$. During the so-called A^* search the estimated possible merit is used to decide which node is expanded next. If the possible merit $f(n)$ is always underestimated, the A^* strategy will always find the best solution first (see e.g. [Nilsson 1982]).

4.3 Heuristics

Several heuristics can be used to reduce the number of nodes which have to be visited in the tree. Because the future merit is always overestimated, the A^* algorithm tends to focus the search on the nodes at the higher levels of the tree. If we use a lower estimate of the total merit, calculated with the formula $f^*(n) = g(n)+(1-\epsilon)\,h^*(n)$, the search algorithm will reach the leaf nodes faster. There is, however, a risk of loosing the optimal solution, but the loss of optimality is limited to $\frac{\epsilon}{1-\epsilon}$ percent of the best solution [Pearl 1984].

The number of nodes visited during a search depends on the total number of image and landmark primitives. Another important factor influencing the size of the search tree is the number of possible assignments between label and unit primitives determined by possible correspondences of attribute values. It can be shown, that the size of a search tree is reduced significantly, if units with only few corresponding labels are used at the very first levels of the tree [Haralick and Elliott 1980].

The search time can be further reduced, if branches of the tree which do not lead to a solution are identified as early as possible. Since a correspondence of three points determines the transformation parameters, wrong branches can be found by transforming the landmark model into the image description after the assignment of three points. The coordinate differences of image and landmark points after the transformation are a good indicator whether or not the assumed correspondence is possible. The search at a wrong branch often can be terminated after the assignment of three points.

In section 3.2 we mentioned the necessity of using wildcard assignments, if primitives can not be matched. For this reason, a consistent result can always be achieved by just adding wildcards to a path. Of course, this is not desired. Therefore, the number of possible wildcard assignments is limited and the search algorithm will not expand a node, if this number is exceeded.

5 RESULTS

Figure 3 right shows the landmark model projected into the hue image using the parameters which were calculated by the matching algorithm. The algorithm had to find a match between 25 label or model primitives (6 points, 12 lines, 7 regions) and 67 image or unit primitives (19 points, 33 lines, 15 regions). In all examples we calculated, the scale between the landmark model and the image was assumed to be within a range of ±15% of a given value. In example street1, a match was found after examining 52 nodes of the search tree (figure 3 middle). As mentioned in section 3.2, the algorithm projects the model into the image after three image points have been mapped to model points. Six

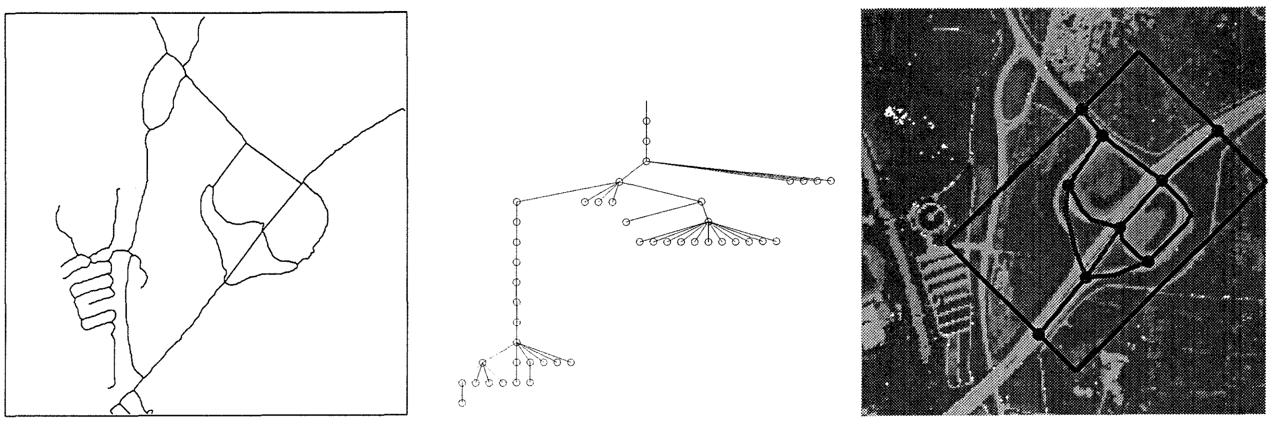

Figure 3: Example street1: extracted image lines (left), search tree (middle), result (right)

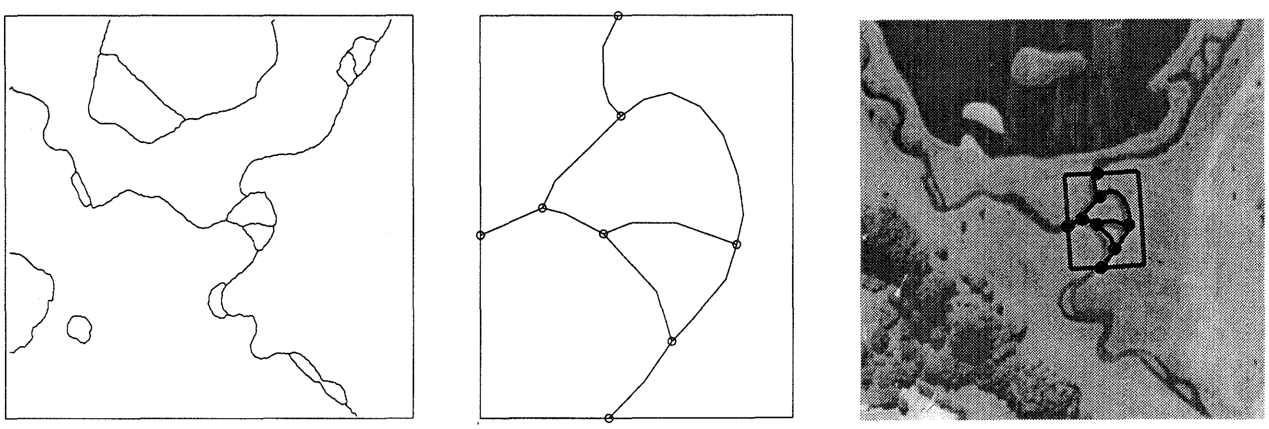

Figure 4: Example river

transformations with different point combinations were tried until an acceptable solution was found. In our application, written in the programming language POP-11, the algorithm needed 227 seconds CPU time on a VAX Station 3200 in order to find a match for example street1.

Figures 4-5 give some results of the matching algorithm for other examples. The figures show the extracted line images, the landmark models, and the models projected into the images.

In the following table the most important parameters of the calculated matchings are summarized. The table contains the number of model primitives (units), the number of extracted image primitives (labels), the number of examined nodes in the search trees (nodes), the number of tried transformations (trans.), and the search time in CPU seconds.

Example	units	labels	nodes	trans.	CPU [sec]
street1	25	67	52	6	227
street2	27	59	47	1	100
street3	23	71	52	6	228
parcel1	21	91	58	2	149
parcel2	14	116	3307	1263	3557
river	18	62	136	32	179

The number of units and labels influence the number of nodes that have to be expanded and therefore the CPU time that is needed to find the match. Still there are some other factors influencing the complexity of the search space. These factors make it hard to predict the time the algorithm needs to find the match. The example parcel2 demonstrates, that the search time strongly increases, if the image contains many objects or object parts with similar primitives and relations. This is caused by the exponential enlargement of the search space if there are many possible correspondences between model and image primitives.

Differences between the description of the landmark model and the image, which make it necessary to use wildcard assignments also have a great influence on the size of the search tree. To demonstrate this, we produced an incorrect image description of example street1 by misclassifying several pixels by hand (figure 6). The right solution still can be found, but the size of the expanded search tree (figure 7) increased to 293 nodes and the CPU time increased to 351 seconds.

Figure 6: Image lines with segmentation error

6 CONCLUSIONS

Landmarks can be located by matching relational descriptions

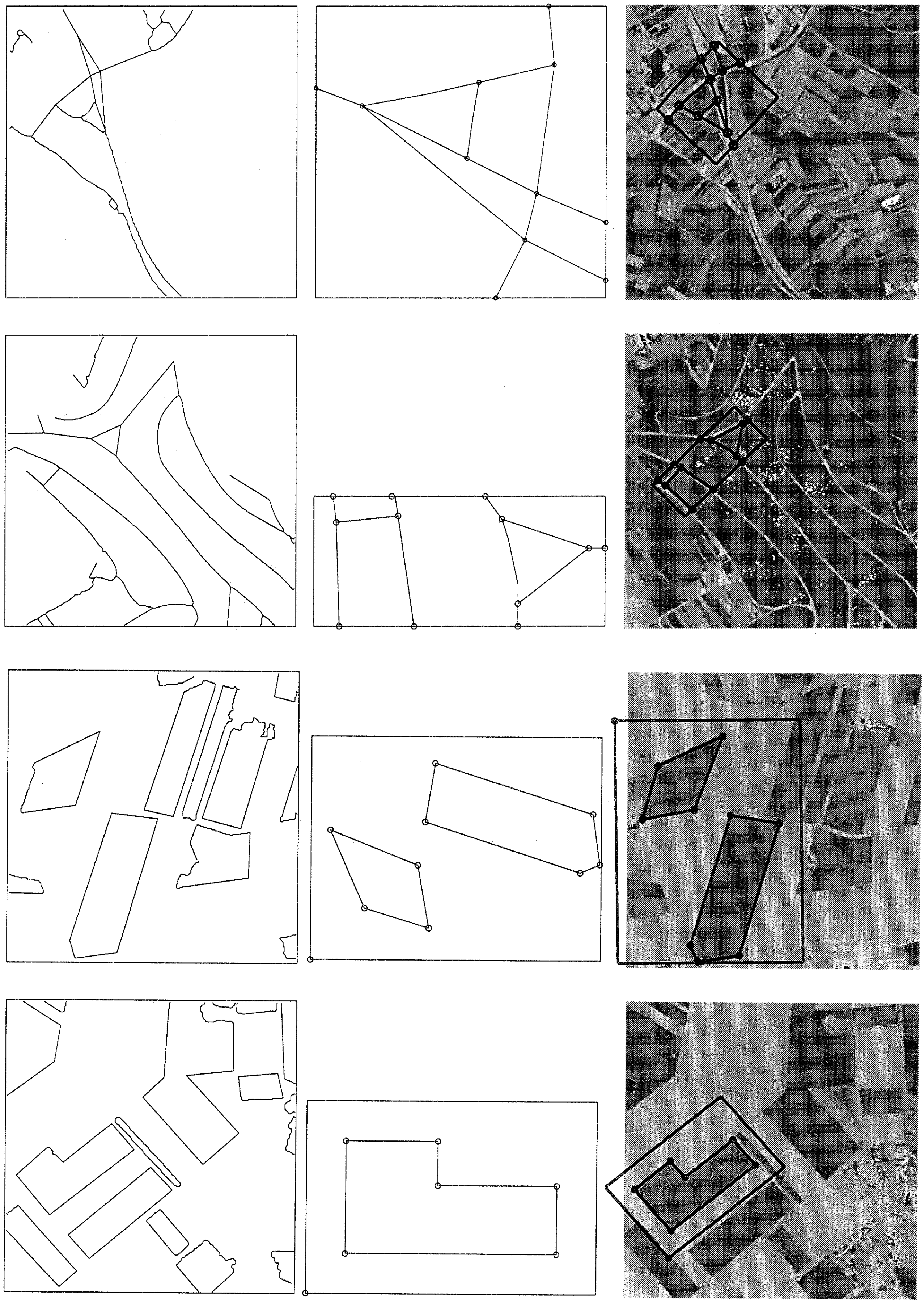

Figure 5: Examples street2, street3, parcel1 and parcel2

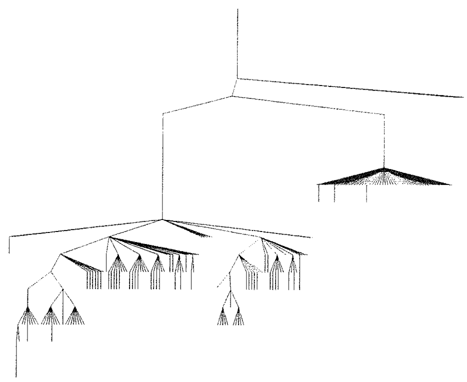

Figure 7: Search tree for the incorrect image description

of landmarks and images. In this paper we focussed on the recognition of a landmark in the image. We did not discuss the exact measurement of the landmark's coordinates. Once a landmark has been located approximately, other methods can be used for the coordinate measurement (e.g. a robust least squares adjustment [Sester and Förstner 1989]). For the recognition of a landmark inaccurate models and a simple geometric transformation (affine transformation) are sufficient. A precise measurement would require accurate 3-dimensional landmark models and a full model of the perspective transformation.

The relational image descriptions we used were extracted from colour images and colour infra-red images. The colour information was of crucial importance for the feature extraction. Without the use of colour (or multi-spectral) images, a reliable extraction of road and rivers is hardly possible.

In order to determine the exterior orientation of an image, it is necessary to measure three landmarks at least. The landmarks we used all contained a minimum number of five points. If the terrain coordinates of those points would be known, one could calculate a spatial resection after the measurement of only one landmark. Of course, the accuracy of this resection would be bad, because the points of the landmark lie closely together. However, the transformation parameters can be used to constrain the search space for the recognition of the landmarks that remain to be measured. The relational matching algorithm does not require approximate values, but, if approximate values are available (e.g. for scale rotation or position of the landmark), they are very useful for reducing the search space.

Such approximate values can be easily integrated into the evaluation of the mappings with the mutual information. The more accurate these values are, the higher the conditional probabilities will be. E.g., if the image scale factor is known to be near S, one knows that the length of a line in the landmark should be about S times the length of the corresponding line in the image description. This helps to discriminate between correct and wrong correspondences. The conditional probabilities (and also the mutual information) of the correct correspondences will be high and the conditional probabilities of the wrong correspondences will be low. The calculation or empirical acquisition of the probabilities does require quite some effort before one can start with the matching. But this only has to be done once. In all six examples we used the same probability tables. Once the probabilities have been determined, the maximum likelihood mapping between two descriptions can be found by maximizing the mutual information.

Unfortunately, the search time needed to find a match is hard to predict. In the first place it depends on the number of image and model features that have to be matched. Two other factors also have a strong impact on the search time: the quality of the image description and the uniqueness of the landmark attributes. Differences between the geometry or topology of the image and landmark description lead to a substantial increase of the search space. These differences are usually caused by errors in the segmentation of the image. A good image segmentation is therefore very important. In the previous section the example parcel2 showed a relatively high search time. This was caused by the fact that the image contained many features and relations between features with similar attribute values. This increases the search effort that has to be made in order to find the correct mapping. One therefore should use such landmarks that have unique attribute values. This limits the number of mappings that have to be evaluated and therefore limits the search time.

LITERATURE

Ackermann, F. and P. Krzystek [1991]: MATCH-T: Automatic Mensuration of Digital Elevation Models. Presented paper to the 3rd Technical Seminar of the Sociedad Espanola de Cartografia Fotogrametria y Teledeteccion, April, 12 1991.

Arcelli, C. and G. Sanniti di Baja [1985]: A Width-Independent Fast Thinning Algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 7, no. 4, 1985.

Frey, H. [1990]: Die Verarbeitung von Farbbildern nach Helligkeit, Sättigung und Buntton. In: *Mustererkennung 1990*, Informatik Fachberichte, no. 254, Aaalen, September 1990, Springer Verlag, pp. 324-331.

Haralick, R.M. and G.L. Elliott [1980]: Increasing Tree Search Efficiency for Constraint Satisfaction Problems. *Artificial Intelligence*, vol. 14, pp. 263-313, 1980.

Nilsson, N.S. [1982]: Principles of Artificial Intelligence. Springer-Verlag, Berlin Heidelberg New York, 1982.

Pearl, J. [1984]: Heuristics: Intelligent Search Strategies for Computer Problem Solving. Addison-Wesley, Reading, MA, 1984.

Schenk, T., J-C Li and C. Toth [1991]: Towards an Autonomous System for Orienting Digital Stereopairs. *Photogrammetric Engineering & Remote Sensing*, vol. 57, pp. 1057-1064, 1991.

Sester, M. and W. Förstner [1989]: Object Location based on Uncertain Models. In: *Mustererkennung 1989*, Informatik Fachberichte, vol. 219, Hamburg, October 1989, Springer Verlag, pp. 457-464.

Shannon, C.E. and W. Weaver [1949]: The Mathematical Theory of Communication. University of Ilinois Press, 1949.

Shapiro, L.G. and R.M. Haralick [1981]: Structural Description and Inexact Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 3, pp. 504-519, 1981.

Tsingas, V. [1991]: Automatische Aerotriangulation. In: *Proceedings of the 43th Photogrammetric Week*, Series of the Institute of Photogrammetry, Stuttgart University, vol. 15, Stuttgart, September 1991, pp. 253-268.

Vosselman, G. [1992]: A Probabilistic Measure for Comparing Relational Descriptions. In: *Proc. Int. Workshop on Robust Computer Vision - Quality of Vision Algorithms*, Bonn, March 1992, Wichmann Verlag, pp. 151-169.