Assessing the Computational Effort for Structural 3D Vehicle Recognition

Eckart Michaelsen and Uwe Stilla

FGAN-FOM Research Institute for Optronics and Pattern Recognition,
Gutleuthausstr. 1, 76275 Ettlingen, Germany
{mich,usti}@fom.fgan.de

Abstract. A model based structural recognition approach is used for 3D detection and localization of vehicles. It is theoretically founded by syntactic pattern recognition using coordinate grammars and depicted by production nets. The computational effort significantly depends on certain tolerance parameters and the distribution of input data in the attribute domain. A brief theoretical survey of these interrelations is accompanied by comparing the performance on synthetic random data to the performance on data from different natural environments.

1 Introduction

In structural computer vision the computational effort often depends on the data. Investigating such interdependencies therefore is an important issue. For the 3D detection of man-made objects in images model knowledge can be represented by e. g. productions, frames or semantic networks [9]. Utilization of knowledge is commonly understood as a search for corresponding objects in the data. Bottom up, top-down or mixed strategies are used for structural approaches. A* search [10] may serve as a well known example. Some heuristic evaluation function is used, that assesses the maximal or probable merit of intermediate results with respect to the final goal of complete model to data correspondence. There are tasks that hardly permit the formulation of such a function.

Vehicle recognition from oblique and very oblique (nearly horizontal) views is an example for such a task. In contrast to aerial vertical views [6,15] size and aspect are very variable. Also radiometry and contrasts of the target object and other objects in the background or foreground are hardly predictable. Some variations are displayed in Fig. 1. It is difficult to define preferences or exclusions for intensities, contrasts, positions, directions, sizes etc. We propose a structural approach using a complete bottom-up part-of analysis. This approach competes with mutual information methods [4] and some quite similar but probabilistic methods based on generalized cylinders [1]. Since our approach leads to high computational effort we propose to use rather simple well scaling methods and structures. Therefore, the assessment of worst-case and probable efforts and the verification of such assessments on relevant data are a worthwhile endeavour.

F.J. Ferri et al. (Eds.): SSPR&SPR 2000, LNCS 1876, pp. 357-366, 2000.
Springer-Verlag Berlin Heidelberg 2000

Fig. 1. Ground-based images of different vehicles (VWBUS-PICKUP). a) Scene 1: Object distance ~20m, sunny, visibility mediocre, b) Scene 2: Object distance ~130m, diffuse, clear visibility, c) Scene 3: Object distance ~320m, diffuse, dull visibility

Computational complexity analysis is common practice in other related pattern recognition disciplines like e.g. labeling line drawings of polyhedral scenes [11], geometric hashing [17], or structured methods based on volumetric primitives and aspect graph matching [2], but has not jet been challenged in our section of syntactically inspired structural methods.

Section 2 shortly recalls production net definitions, methods and implementations. An example net is given in section 3. Using this example the effort assessment method is discussed in section 4 and practical results are given in section 5.

2 Production Nets for Object Recognition

Most symbolic methods in pattern recognition deal with structures like strings, trees, arrays or graphs. Production net theory is based on coordinate grammars and thus simply uses *sets* [7,8]. The productions work on sets of *instances (s,d)* consisting of a symbol $s \in T \cup N$ from a finite set of terminals and non-terminals and a numeric attribute vector $d \in D$ from a domain which usually contains coordinates, orientations, surface normals etc. Pairs, triples, etc. of such instances are called *configurations*.

2.1 Production

Productions consist of a condition and an action part. The condition part gives a predicate defined on the input configuration. The action part gives a function calculating the output configuration (usually a single object). A simple example is given by

$$\left(\left(LINE, LINE\right), \pi\right) \xrightarrow{\varphi} \left(ANGLE\right) \tag{1}$$

Objects of type *LINE* have image coordinates and orientations as attributes. The condition demands a pair *(LINE,LINE)* fulfilling π which defines 'adjacent and rectangular' with some necessary tolerances. Function φ calculates the intersection of the straight lines corresponding to the input configuration. This coordinate is needed as attribute value for the new object *ANGLE*.

Generally a production contains left and right words Σ and Λ of symbols from $T \cup N$ with at least one non-terminal in Λ. We write $|\Sigma|$ for length of the word Σ and $\{\Sigma\}$ for its corresponding multi-set.

2.2 Production Net

Production nets display the interaction of several such productions in a system. As graphs they resemble Petri nets. The set of nodes is given by the set of symbols (object types) and the set of productions. An edge leads from an object type to a production, if the condition part contains it. If it is contained multiply, it is drawn multiply. An edge leads from a production to an object type, if it produces it. Examples for production nets are published in [12,13,14,15].

2.3 Model Knowledge for Vehicles

There are several possibilities for modeling vehicles geometrically. One may e.g. use articulated 3D models. The projection may also be included in the geometric model, so that finally 2D views - or linear combinations of these - are matched like in [16]. Such modeling may be used, if the camera is directly approaching the target object. Otherwise stereo methods and 3D matching with articulated models are preferred. For the statistic discussion in this context we refer to a hierarchically organized shape fixed model of a little truck already known from [7,8].

2.4 Implementation

Our Implementation is based on a blackboard shell named BPI [13]. Each production defines a separate processing module containing condition test and action part. All modules work on a common memory. They insert new instances, but they do not delete the instances of the input configuration. Thus the system works accumulating instead of replacing. Such irrevocable control facilitates the processing of large data sets at moderate effort scaling [10]. The accumulation method serves as approximation of the semantically correct replacement and backtrack method [7,8]. Associative memory aids the reduction of effort scaling [13].

3 Example Production Net

As example vehicle we choose a small six seated truck and named the corresponding object type *VWBUS-PICKUP*. Fig. 2 shows the production net designed for the recognition of such objects in very oblique image sequences accompanied by informal sketches of the meaning of intermediate object types. This has been published before in more detail [7,8].

Two sub nets may be distinguished according to the dimension of the attribute domain. The main model-to-data match is implemented in the 3D sub net. The 2D sub net contains some simple standard productions for line prolongation, corner and U-structure composition. It is executed on each image separately and linked to the 3D sub net with the stereo production p4. The extraction of line segments from the images has been described in [12].

Provided each 3D part required by the net is visible in at least two images (the sequences used consist of eight frames) a lot of occlusion is tolerable. Invariance of the detection result is given with respect to a large variety of aspects and distances and with arbitrary back- and foreground objects. Due to the deep part-of hierarchy and 3D model use false detection is very unlikely. But a high detection rate requires generous tolerance parameters in the conditions, which becomes computationally expensive.

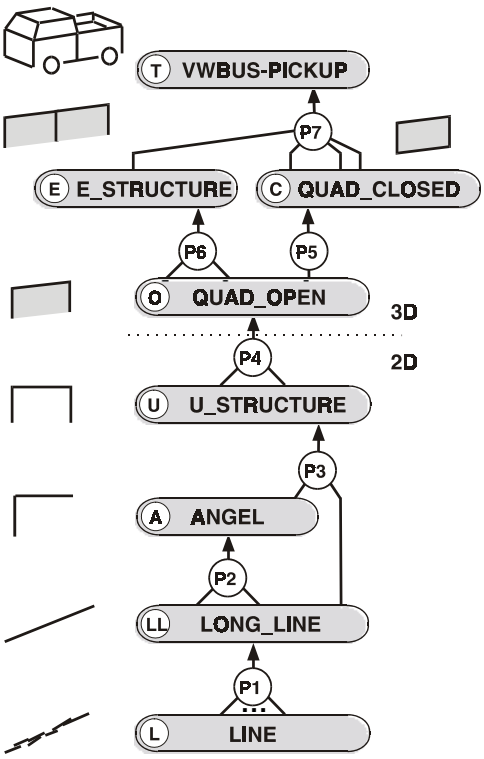

Fig. 2. Production Net VWBUS-PICKUP

4 Statistical Effort Assessment

If s_i denote the object types a standard production like p3 in Fig. 2 may be written as

$$p : \left(\left(s_1, s_2 \right), \pi \right) \xrightarrow{\varphi} \left(s_3 \right) \tag{2}$$

We denote the set of all corresponding input configurations fulfilling π as $\Im_p$ and define the *relative volume* V_p as the ratio between $|\Im_p|$ and the size of the set of all possible configurations. The latter is given by the attribute domain and the number of objects in the input configuration.

$$V_p = \frac{\left|\Im_p\right|}{\left|D^2\right|} \tag{3}$$

This gives a measure for the degree of restriction provided by a production. If e. g. π be 'parallel' in an orientation domain o = {0°,...,179°} with tolerance δo = ±9°. Then we get V_p = 0.106 Often relative volumes will result from a product, because π is composed as conjunction of conditions on independent attributes. If e. g. additionally to 'parallel' also 'adjacent' is required with some tolerance of 10 pixel in Euclidian metrics in an image of 1M pixel size, we get

$$V_p \approx \frac{19}{180} \cdot \frac{314}{10^6} = 0.000033\,. \tag{4}$$

Thus small relative volumes result from high dimensional attributes, narrow tolerances and many independent conditions connected as conjunction.

Provided a random process generates sets of instances S_1 and S_2 of the object types s_1 and s_2 with known distribution in D an expectation may be calculated for the number of instances of s_3 reduced by p (Eq. 2). Equally distributed attribute values in S_2 for instance give a Poisson distribution with parameter $\lambda = |S_2| V_p$ for the number of partners in S_2 fulfilling π together with a fixed instance of s_1 [5]:

$$P(\text{No. of Partners} = k) = \frac{1}{k!} e^{-\lambda} \lambda^k\,. \tag{5}$$

Expectation value for this distribution is λ. We neglect that in rare cases the same instance s_3 may result from different input configurations. We assume independence of the instances s_1 from instances s_2. Then $|S_1|\lambda$ is an expectation for the number of instances s_3 resulting from p and we get

$$E\left(\left|S_3\right|\right) = E\left(\left|S_1\right|\right) E\left(\left|S_2\right|\right) V_p \tag{6}$$

Such equations may be constructed for any production p_j in any net:

$$p_j : \left(\Sigma_j, \pi_j\right) \xrightarrow{\varphi_j} \Lambda_j \text{ with } V_{p_j} = \frac{\left|\Im_{p_j}\right|}{\left|D^{|\Sigma_j|}\right|} \tag{7}$$

$$s \in \left\{\Lambda_j\right\} \text{ and } S = \left\{(s,d); \xrightarrow{p_j} (s,d)\right\} \Rightarrow E\left(|S|\right) = \prod_{s_i \in \left\{\Sigma_j\right\}} E\left(\left|S_i\right|\right) V_{p_j} \tag{8}$$

Cycle free production nets provide an order *O(s)* on the object types given by the length of the longest path leading from a terminal to *s*. For such nets the expectation

equations are used with ascending O to calculate all E(|S|) up to the goal type using the sums

$$E(|S|) = \sum_{s \in \{\Lambda_j\}} \prod_{s_i \in \{\Sigma_j\}} E(|S_i|) V_{p_j} \tag{9}$$

and starting with the known distributions for the terminals.
The probable overall demand for memory is then given by the sum of all these expectations. For the probable computational effort of a full bottom up search we have to weight each sum with the computational costs caused by an instance of its symbol. This is a constant amount mainly consisting of its construction effort plus the costs for the queries it causes because of the Σ in which it appears. All this can be calculated in advance.

5 Experiments

Synthetic random data as well as data from real outdoor images are used to verify the relevance and precision of the calculations presented above. Production p6 of the 3D sub net has been applied to equally distributed random generated sets of instances O with varying sizes and thus densities. The attribute domain here contains four 3D coordinates and one surface normal. Table 1 gives the set sizes.

O	1040	2196	4592	9021	12835	25141	50684	101243
E	2	12	47	227	425	1639	6638	25165

Table 1. 3D Statistics -Random instances O and generated instances E (p6)

The set size of the set of instances E grows quadratic with the set size of the set of instances O. V_{p6} has been estimated at $\approx 10^{-6}$ according to the size of the attribute Domain (3D coordinates in 500^3 and surface normal) and tolerances (±50 in *max*-norm for 3D coordinates and ±0.3*rad*). The data in Table 1 yield a quadratic parameter of $2.7 \cdot 10^{-6}$. Such differences result from imprecision in the theoretic calculations (for instance neglecting special properties at the rim of coordinate spaces or estimations with linearization of orientation manifolds). We regard $\lambda > 1$ as critical values, because the desirable monotone decrease of set sizes with O will be violated. An attribute domain of the given size in the example should therefore not contain more than 370000 instances O.

Fig. 3b-d show natural input data extracted from the images in Fig. 1. The distribution of instances O resulting from such image sequences are rather unequal. Dense clusters and large nearly empty zones occupy the attribute domain (here 2000^3). E. g. in scene 1 (Fig. 3b) 25427 instances E are constructed from 12997 instances O. Consequently the mid density of instances in the overall domain is of less relevance for the effort assessment compared to the density in the clusters (which is much harder to be measured or estimated).

Fig. 3. Instances L from images of Fig. 1 (sections 600x400 Pixel).
a) Data Set Random 2 (1400x700 Pixel), b) Scene 1 (1400x700 Pixel),
c) Scene 2 (3200x1600 Pixel), d) Scene 3 (6400x3200 Pixel)

Differences between effort statistics of synthetic random data and real data are less significant with 2D productions. Columns 1-3 in Tab. 2 confirm the predicted polynomial growth of set sizes with the polynomial degree depending on *O*. Natural data still give different characteristics. Scenes 1 and 2 for instance yield significant minima at object type A not present in the random data. The system tends to make background suppression at this stage (see Fig. 4). Like in 3D mid density is not the most important feature (columns 3 and 5 are similar in this parameter). A more important contribution is given by things like structure and lighting. Scene 2 for instance has a lot of man-made straight lines and high contrast rectangles in it resembling the structure to be detected and thus poses much more challenge than the more blurred and less structured scenes 1 and 3.

Type	Random 1	Random 2	Random3	Scene 1	Scene 2	Scene 3
L	4318	8598	17185	11299	74253	84952
LL	884	3349	11893	5704	85047	46952
A	359	5168	64677	2455	47368	34284
U	59	4084	185801	6185	154076	52131

Table 2. 2D Statistics - Set Sizes for Object Types L, LL, A, and U

Detection success and failure of the system is not the topic here (has been published in [7,8]). Short: Correct instances T result from scenes 1 and 2, whereas scene 3 yields no instances T.

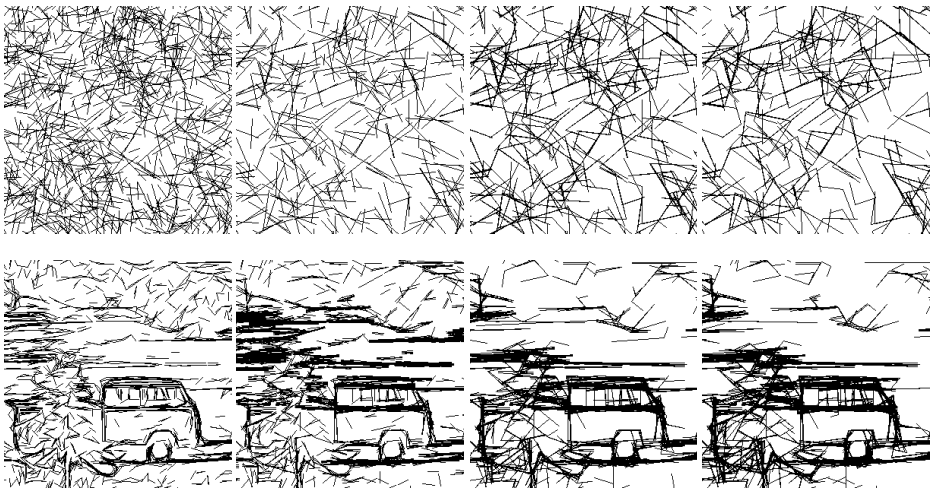

Fig. 4. Instances L, LL, A, and U of data set Random 2 and Scene 1 (sections)

6 Discussion

Principally target objects may be modeled from terminal objects using arbitrary partial objects. For instance a rectangle may be constructed from four lines using angles as well as parallels as intermediate objects. If knowledge about expected background structures is given (e.g. major orientations), then the corresponding structural relations should be avoided in the low order productions of the net (e.g. parallel). Figure 3c shows long straight contours from furrows and right angled structures similar to the ones present in the target model. In such cases high computational effort on background objects can not be avoided.

Certainly the terminal objects extracted from images of natural scenes will not be equally distributed. For the terminal object sets displayed in Fig. 3 the distribution of the attribute orientation is shown in Fig. 5. In ground based images with man-made structure vertical and horizontal lines may dominate (Fig. 5b,c). In vehicle detection tasks the majority of the terminal objects stem from arbitrary structures in the background or foreground, about which nothing is known. In such situation equal density and independence assumptions inherent in the investigations of Sect. 4 are appropriate. However, if the distribution of an attribute is given, the simple calculation of the expectation in Eq. 6 will have to be replaced by explicit integration.

For example, for a production constructing parallel pairs of lines a significant peak in the orientation histogram will rise the expected number of constructed objects.

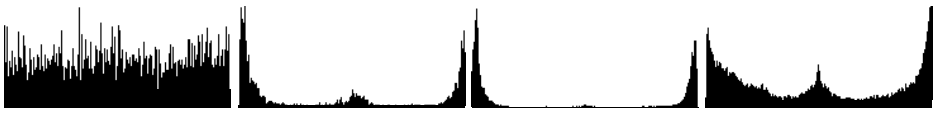

Fig. 5. Histograms of the attribute orientation (0°-179°) of instances LL. a)-d) corresponding to Fig. 3 a-d

Some structural relations allow the assessment of their relative volumes by displaying corresponding search regions. Fig 6 shows examples: Adjacency in vector spaces with a maximum metric and a threshold parameter simply gives an interval (Fig. 6a), a square (Fig. 6c), or a cube (Fig. 6f). Note, that the volume of such regions grows in a polynomial way with power D (the dimension of the domain). Thus fairly small changes in the threshold parameter of a 3D structural relation may have severe consequences on the computational effort. Topologically more complicated are relations on orientation attributes. The second column shows the examples line orientation (Fig. 6b), surface orientation (Fig. 6d), and 3D rotation (Fig. 6g). The exact calculation of the relative volume of the structural relation 'similar in 3D rotation' with the same threshold in all three angles (Fig. 6g) requires techniques from differential geometry. At least for small angles volume growth with power D will still be present. But there are relations, where the power of growth will be less than the dimension. Fig. 6e,h show the search regions corresponding to adjacency of a line. The size of these regions grow linear in the 2D case and quadratic in the 3D case. The length of the rectangle or cuboid is fixed by the length of the line.

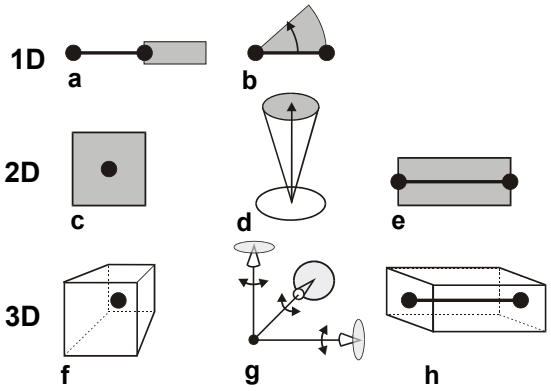

Fig. 6. Search regions for important structural relations

We presented a method for the assessment of the computational effort caused by the analysis of images by a production net. Dependencies on tolerance parameters and densities of instances in the data become evident. These calculations provide valuable quantitative information for the overall system design. Comparisons of the effort between the presented systems and other ones are difficult, because they are not available. Success and effort also strongly depend on the task, the model and the images used. A comparison of the effort and stability of different approaches requires

re-implementations. Subject of ongoing work is e.g. the implementation of aspect based vehicle recognition.

References

1. Binfort T. O.; Levitt T. S.: Model-based Recognition of Objects in Complex Scenes. In: ARPA (Ed.): Image Understanding Workshop 1994 (Montery) Morgan Kaufman, San Franzisco, 1994, pp 149-155.
2. Dickinson S. J.; Pentland A. P.; Rosenfeld A.: From Volumes to Views: An Approach to 3-D Object Recognition}. CVIGP:IU, Vol. 55, No. 2, 1992, pp 130-154.
3. Gruen A., Baltsavias E.P., Henricsson O.: Automatic extraction of man-made objects from aerial and space images (II). Birkhaeuser, Basel, 1997
4. Hermiston K. J.; Booth D. M.; Foulkes S. B.; Reno A. L.: Pose Estimation and Recognition of Ground Vehicles in Aerial Reconnaissance Imagery. In: Jain K.; Venkatesh S.; Lovell B. C. (Eds.): ICPR'98, IEEE, Los Alamitos, 1998, pp. 578-582.
5. Lloyd E.: Probability. In: Ledermann W.: Handbook of Applicable Mathematics, Band II, John Wiley and Sons, Chichester, 1980.
6. Matsuyama T.; Hwang V. S.: Sigma A Knowledge-Based Aerial Image Understanding System, Plenum Press, New York, 1990.
7. Michaelsen E.: 3D Coordinate Grammars. In: Girod B.; Niemann H.; Seidel H.-P. (Eds.): 3D Image Analysis and Synthesis '96, Infix, Sankt Augustin, 1996, pp. 81-85.
8. Michaelsen E.; Stilla U.: Remarks on the Notation of Coordinate Grammars. In: Advances in Pattern Recognition, Joint IAPR Workshops SSPR'98 and SPR'98 in Sydney, Springer, Berlin, 1998, pp 421-428.
9. Niemann H.: Pattern Analysis and Understanding, Springer, Berlin, 1990.
10. Nilsson N. J.: Principles of Artificial Intelligence. Tioga Publ., Palo Alto, 1980 (or Springer, Berlin, 1982).
11. Parodi P.; Lancewicki R.; Vijh A.; Tsotsos J. K.: Empirically-derived estimates of the complexity of labeling line drawings of polyhedral scenes. AI, Vol. 105, 1998, pp 47-75.
12. Stilla U.: Map-aided Structural Analysis of Aerial Images. ISPRS Journal of Photogrammetry and Remote Sensing, Vol 50, 1995, pp 3-10.
13. Stilla U.; Michaelsen E.; Luetjen K.: Structural 3D-Analysis of Aerial Images with a Blackboard-based Production System. In: Gruen A.; Kuebler O.; Agouris P.: Automatic Extraction of Man-Made Objects from Aerial and Space Images, (Ascona Workshop der ETH), Birkhäuser, Basel, 1995, pp 53-62.
14. Stilla U.; Michaelsen E.; Luetjen K.: Automatic Extraction of Buildings from Aerial Images. In: Leberl F.; Kalliany R.; Gruber M.: Mapping Buildings, Roads and other Man-Made Structures from Images, (Proceedings IAPR TC-7 Workshop, Graz, 1996), Oldenbourg, Wien, 1997, pp 229-244.
15. Stilla U.; Michaelsen E.: Semantic Modeling of Man-Made Objects by Production Nets. In: Gruen A.; Baltsavias E. P.; Henricsson O.: Automatic Extraction of Man-Made Objects from Aerial and Space Images II, Birkhäuser, Basel, 1997, pp 43-52.
16. Wang P. S. P.: Parallel Matching of 3D Articulated Object Recognition. Int. Journ. of Pattern Recognition and Artificial Intelligence, Vol 13, 1999, pp 431-444.
17. Wolfson H. J.: Model Based Object Recognition by Geometric Hashing. In: Faugeras O.: Computer Vision - ECCV90, Springer, Berlin, pp 526-536.