MAKING STRUCTURAL PATTERN RECOGNITION TRACTABLE BY LOCAL INHIBITION

E. Michaelsen, L. Doktorski and M. Arens

Forschungsinstitut für Optronik und Mustererkennung, Gutleuthausstr. 1, 76275 Ettlingen, Germany

Keywords: Structural pattern recognition, Local inhibition.

Abstract: Declarative knowledge and control decisions on the sequence of interpretation acts are separated in a structural pattern recognition system. The control can be optimized leaving the knowledge fixed. A simple production system is used as declarative example knowledge. It is tailored to recognize and locate rectangles in images – where object primitives are several thousand very short contour segments. Different control strategies can be realized: (i) a simple quality driven bottom-up control; (ii) an heuristic strategy punishing object instances which have been partner in an already performed reduction and (iii) a new psychologically inspired strategy that combines local inhibition with less local excitation. These strategies are compared quantitatively on synthetic data and qualitatively on a real aerial image.

1 INTRODUCTION

Controlling the search for feasible reductions given a declarative knowledge structure – such as a production system – and a set of measured primitive objects has been an interesting topic ever since the proposal of structural pattern recognition was put forth decades ago. In particular production systems can implicitly define huge combinatorial search spaces that cannot be systematically explored with feasible effort. The following three major requests are set for the control unit of a production system interpreter suitable for pattern recognition and machine vision: (i) it should avoid visiting the whole combinatorial search space; in fact it should get along with working out only a very tiny portion of it; (ii) it should be capable of handling large numbers of object instances; and (iii) it should have anytime capability in the sense that it provides some reductions of full depth very early and use additional time to check alternatives and improve the evidence for the validity of the already found. If run to a complete end or infinitely the system should approach a correct interpreter.

For these requests the correctness of the interpretation system may be traded. The practitioner is satisfied with approximate correctness as long as it does not affect the usability and reliability of the system on real data. In Section 2 of this contribution we give an approximating interpreter that is tailored to these requests. A particular simple system containing only a few simple productions is given in Section 2. It is complex enough to elaborate the differences – yet simple enough to maintain an overview on what is going on. Section 3 compares three control strategies using this setup. Quantitative testing is done with synthetic data and there is an additional qualitative assessment with a real aerial image. The remainder of this chapter will set the work in relation to the published state-of-the-art in the field.

Automatic "image understanding" has been a major issue in pattern recognition for many decades (Basu et al., 2005; Matsuyama and Hwang, 1990; Draper et al.,1989). Most such proposals were tailored to automatic recognition – of man-made structure in particular – from remote sensing data. But structural and cognitive methods for image analysis are not restricted to remote sensing applications. There are also examples of similar structure that were originally designed for medical applications (Niemann et al., 1990) and for automatic reasoning for safety in robotics (Qureshi et al., 2005). All of these proposals contain also approaches to optimizing the control – in particular ERNEST had a sophisticated theory of optimal control with it. Syntactic methods for image understanding recently gain interest again. E. g., (Zhu and Mumford, 2006b) have introduced a stochastic grammar for the understanding of images. Emphasis is not on efficient control of user defined fixed knowledge but on defining a sound description langugage capable of capturing and learn-

ing diverse and general visual categories. Biological inspiration for our inhibition/excitation function presented in Section 3 comes from interdisciplinary work between perceptual psychology and computer vision such as (Aziz and Mertsching, 2007). Our work can be seen as a straight decendent and continuation of the blackboard-based accumulating interpretation system proposed in (Stilla and Michaelsen, 1997). There the main application was seen in the recognition of man-made structures – such as buildings – in aerial imagery. The system has also been used for other purposes (compare (Michaelsen et al., 2006; Michaelsen et al., 2006)).

Intelligent control has always been a major issue. Using Gaussian local inhibition in order to acquire a priority sorting for 3D-surface plane patches in laserdata has been reported by (von Hansen et al., 2006a). Though this work used the same interpretation dispatcher the inhibition was only performed after the search in order to rank order the resulting object set. In this contribution such inhibition is used for the control of the search for all kinds of objects during the run.

2 A SIMPLE SYSTEM

The example production system acts on primitive objects called line, which are extracted from an image with a gradient filter. For simplicity, the system only has three non-primitive object classes, namely longline, angle and rectangle. The system contains the three productions given in Table 1.

Table 1: Example production system. The terminal symbol l denotes lines, whereas the non-terminal symbols L, A, and R denote long lines, angles, and rectangles, respectively.

$$L \xrightarrow[regression]{co-linear \,\wedge\, overlapping} l \cdots l \quad (1)$$

$$A \xrightarrow[intersection]{rectangular \,\wedge\, adjacent} L\, L \quad (2)$$

$$R \xrightarrow[intersection]{crosswise\ adjacent} A\, A \quad (3)$$

We distinguish two normal forms of productions: Normal form 1 generates a pair of objects from a single non-terminal object. Productions of this form are used for part-of intentions (compare productions (2) and (3) in Table 1). Normal form 2 generates a set of objects of arbitrary (finite) size of the same type from a single non-terminal object. Such productions are used for cluster intentions (Michaelsen et al., 2008). Production (1) for example codes a Hough-type accumulator for straight lines.

The system codes declarative knowledge for searching rectangles in pictures. The interpretation system we use approximates correct parsing. Pseudocode scheme Algorithm 1 explains the interpreter. It works accumulating instead of reducing and thus avoids backtracking. Hypotheses are constructed of single (triggering) objects, and productions where such objects appear on the right-hand side. Such hypotheses are tested by looking for appropriate partner objects. Once the set fulfilling the query is found the procedure is different for the two normal forms: for productions of normal form 1 all elements of the set are handled separately – according to the combinatoric nature of the production system. Each possible partner leads to a new and separate reduction possibility. For productions of normal form 2 all elements of the set are used for one minimization calculation. They lead to exactly one new instance of the cluster object type of the left hand side of the production.

Algorithm 1: Approximative interpretation of production systems.

```
while queue not empty OR other break do
    sort(queue);
    set_of_hopothesis = choose_best_n(queue);
    foreach trigger_hypo ∈ set_of_hopothesis do
        if p=nil then
            remove_queue(trigger_hypo);
            foreach q where trigger_obj ∈ right-hand
            side do
                new_priority = prio(q) *
                priority(trigger_hypo);
                append_queue(trigger_elem, q,
                new_priority);
            end
        else
            actual_query =
            construct_query(trigger_hypo);
            candidate_set = pose_query(actual_query);
            if p of Normal_form_1 then
                foreach partner ∈ candidate_set do
                    p:new_elem ← (trigger_elem,
                    partner);
                    add_database(new_elem);
                    construct_null_hypo(new_elem);
                end
            else
                p:new_elem ← candidate_set;
                add_database(new_elem);
                construct_null_hypo(new_elem);
            end
        end
    end
end
```

---

3 LOCAL INHIBITION

A frequent observation with the dispatcher control unit of our systems showed that the same or very similar intermediate non-primitive objects were multiply reduced from the same objects. Of course there

is a test for existence before the new elements are appended to the set of reducible objects, so as to avoid multiple listing in it, however, the computational effort for the query and construction method is wasted. A more detailed observation of the phenomenon revealed that, e. g., neighbouring primitive line segments often get similar assessments, too. Thus the corresponding working hypothesis tend to cluster together also in the process queue and cause very similar queries and constructions. One way to remedy these unpleasant repetitions is the inclusion of a remove-queue command for the hypothesis corresponding to all the right-hand-side objects into the modules coding the productions. This leads to very efficient systems. However, it contradicts the combinatoric nature of such production systems: only the actual triggering hypothesis can (and must) be removed, the others will cause different queries and thus open different possibilities. With the remove-queue command these possibilities are cut away which alters the declarative semantics of the system. The remove-queue command may be replaced by a re-assess command. This does not alter the declarative semantics of the production system. It shifts all the repetitions to the end of the interpretation run. If the interpretation is halted long before the queue runs empty they will not be performed anymore. For the experiments we used an appropriate inhibition constant $\gamma = 0.5$ (compare Algorithm 2).

Algorithm 2: Heuristic inhibition.

remove_queue(hypothesis(triggering-element));
foreach *objects $x \in$ candidate_set (and not x=triggering-element)* **do**
| priority(hypo(p,x))=γ*priority(hypo(p,x));
end

Algorithm 3: 'Biological' local inhibition.

foreach *hypo(o,p) $\neq$ hypo(ot,pt)* **do**
| priority(hypo(o,p))=$\omega(o,ot)$*priority(hypo(o,p));
end

The sequence of inspection – or saccade or gaze control – has been subject of psychological investigations for a long time. There are also works on incorporating such behaviours into computer vision systems. E. g., recently (Aziz and Mertsching, 2007) described a control mode they call "examine behaviour". I. e. a particular part of the space under observation becomes uninteresting when the focus of interest has been there. The whole closer neighbourhood is lowered in its priority. Instead a less close neighbourhood is getting higher priority (is excited). In particular such objects get more priority which have similar other attributes concerning prop-

erties like orientation, colour etc. Thus a sequence of observation is achieved which follows perceptual Gestalts almost the same way like human subjects do. Following such ideas we have implemented a priority upgrade function ω which is between zero and one in a close neighbourhood and slightly greater than one in the further neighbourhood:

$$\omega = 1 - (1+\alpha)e^{-\delta^2} + \alpha e^{-\frac{1}{2}\delta^2}$$

where δ^2 indicates a specific metric distance from the object of the triggering hypothesis

$$\delta^2 = \sigma_{\text{loc}} * |\text{loc} - \text{loc}(ot)|^2 + \sigma_{\text{ori}} * |\text{ori} - \text{ori}(ot)|^2$$

with weights σ_{loc} and σ_{ori} balancing location in the image against orientation. The values of these parameters where set as $\sigma_{loc} = 0.0004$ and $\sigma_{ori} = 0.08$ (in 512x512 images and with orientation measured in degree) after systematic optimization of the performance. Thus local inhibiton is quite far reaching with respect to the image location but quite narrow with respect to orientation. ω becomes zero for $\delta^2 = 0$ and ω is approaching one for $\delta^2 \rightarrow \infty$. The weight of the exciting versus the inhibiting effect of the function depends on the parameter alpha. The experiments indicated below used $\alpha = 0.9$. Whenever a particular hypothesis (ot,pt) is tested all other hypotheses of the same object and production type are getting a priority upgrade using this function.

4 EXPERIMENTS AND RESULTS

Experiments were performed with 200 randomly generated images each containing one ramdomly rotated, sized (25-100), and positioned square with randomly set grey value on black background, two circular disks drawn with the same specifications, and ten lines of three pixels width drawn accordingly. The images were blurred and Gaussian noise was added. Each image results in two or three thousand primitive lines constructed with a gradient filter. The interpretation using the production-system given in Section 2 was halted when the queue ran empty or an instance of the class rectangle was reduced. In the latter case the experiment was counted as success if the object was found in the correct position (with five Pixels tolerance).

We also made experiments with a real aerial image. In the real data experiment the interpretation was stopped, when twelve or more rectangles were reduced. Tab. 1 shows the results for the three runs (no inhibition, heuristic inhibition and Gaussian inhibition/excitation). Red crosses indicate the centres of the found rectangle objects.
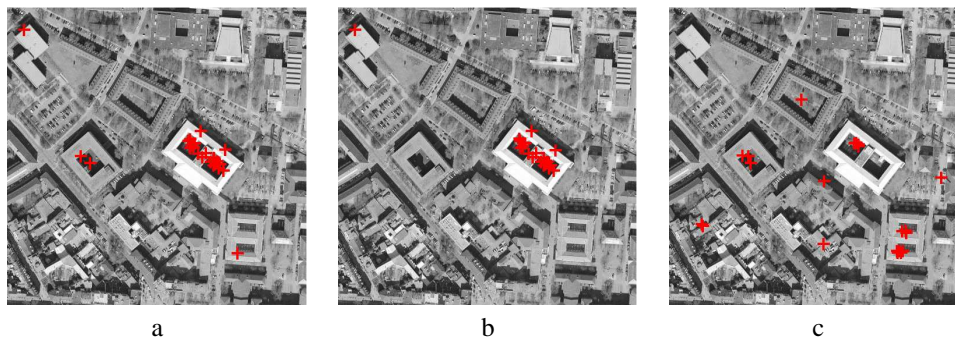
Figure 1: Objects Rectangle found on a real aerial image: a) no inhibition, b) heuristic inhibition, c) Gaussian inhibition/excitation.

Table 2: Results on synthetic images without, with heuristic, and with biologically inspired inhibition.

	success rate	computational effort
no inh.	27%	100%
heur. inh.	29,3%	61,4%
bio.	30,8%	69,2%

While both – heuristic inhibition and Gaussian inhibition/excitation – need much less interpretation cycles than the run without any inhibition (3200 and 3700 versus 5800 cycles) the spreading of the resulting rectangle objects looks much different: the Gaussian control spreads its interest much more in the image space. It marks several less salient rectangles as well.

5 CONCLUSIONS

Both – the heuristic and the more sophisticated control scheme – are successful. An important point is whether the extra administration effort spend for the control calculations equalizes the gain in search effort. This is of course implementation dependent. An important drawback are the extra parameters introduced by the control which are domain-dependent.

REFERENCES

M. Z. Aziz and B. Mertsching: *An Attentional Approach for Perceptual Grouping of Spatially Distributed Patterns.* In: F. Hamprecht et al. (eds.): Proc. 29th DAGM-Symposium 2007, LNCS 4713, Springer, pp. 345-354.

M. Basu et al. (eds): PAMI Special Issue on Structural and Syntactical Methods, 27:7 (2005).

B. Draper, R. Collins, J. Brolio, A. Hanson, E. Riseman: *The Schema System.* IJCV, Vol. 2 (1989) 209-250.

T. Matsuyama and V. S.–S. Hwang: *Sigma a Knowledge–based Image Understanding System.* Plenum Press, New York (1990).

E. Michaelsen, L. Doktorski, and M. Arens: *Shortcuts in Production Systems - A way to include clustering in structural Pattern Recognition.* Proceedings of PRIA-9-2008, Vol. 2, Lobachevsky State University, Nischnij Nowgorod, pp. 30–38.

E. Michaelsen, W. von Hansen, M. Kirchhof, J. Meidow, and U. Stilla: *Estimating the Essential Matrix: GOODSAC versus RANSAC.* ISPRS Symposium on Photogrammetric Computer Vision PCV 2006, Bonn, Germany, September 20-22 2006, Int. Archives of Photogrammetry and Remote Sensing. Vol. XXXVI, Part 3, PPV 2006, pp. 161–166, 2006.

E. Michaelsen, U. Soergel, and U. Thoennessen: *Perceptual Grouping in Automatic Detection of Man-Made Structure in high resolution SAR data.* Pattern Recognition Letters 27:4(2006) 218–225.

H. Niemann, G. Sagerer, S. Schröder, F. Kummert: *ERNEST: A Semantic Network System for Pattern Understanding.* PAMI 12:9(1990) 883–905.

F. Qureshi, D. Macrini, D. Chung, J. Maclean, S. Dickinson, P. Jasiobedzki: *A Computer Vision System for Spaceborne Safety Monitoring.* 8th Int. Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS), Munich (2005).

U. Stilla and E. Michaelsen: *Semantic modelling of man-made objects by production nets.* In: A. Grün et al. (eds.): Automatic Extraction of Man–Made Objects from Aerial and Space Images (II). Birkhäuser–Verlag, pp. 43–52.

W. von Hansen, E. Michaelsen, U. Thönnessen: *Cluster Analysis and Priority Sorting in Huge Point Clouds for Building Reconstruction.* In Proc. 18th Int. Conf. on Pattern Recognition (ICPR'06), Vol. 1, pp. 23-26, 2006.

S.–C. Zhu and D. Mumford: *A Stochastic Grammar of Images* Foundations and Trends in Computer Graphics and Vision 2:4 (2006) 259–362.