MASSIVELY PARALLEL ARCHITECTURE FOR AN UNSUPERVISED SEGMENTATION MODEL

Stephan C. Stilkerich**, *Joachim M. Buhmann***

*EADS Corporate Research Center, Image and Signal Processing Group
D-81663 Munich, Germany
e-mail: stephan.stilkerich@eads.net

**Swiss Federal Institute of Technology, Institute of Computational Science
CH-8092 Zurich, Switzerland
e-mail: jbuhmann@inf.ethz.ch

Abstract - **In this paper we propose the structure of a novel massively parallel system-on-chip (SoC) architecture for a state-of-the-art probabilistic image segmentation model. This probabilistic model is formulated on a regular Markovian pixel grid. The unique combination of algorithmic robustness, SoC and real-time processing capabilities provides a new type of image processing systems in real-world applications. The model and the SoC architecture are extensively tested by natural images. Some chip design examples are also included to finalize the contribution.**

Keywords: Parallel Processing, SoC, Image Segmentation, Hardware Architecture

1. INTRODUCTION

Over the last decade image processing and analysis systems have emerged as crucial building blocks for different applications like video surveillance, medicine, man-machine interface and autonomous vehicle guidance (on ground, air and water). Algorithmic robustness in real world scenarios and real-time processing capabilities are the two contradicting requirements modern image-processing systems have to fulfill to go significantly beyond state-of-the-art. We advocate a system approach where algorithmic robustness is achieved by probabilistic processing models and real-time processing capabilities are provided by massively parallel digital hardware architectures, which exploit the inherent algorithmic parallelism of these proposed models. The tedious problem to estimate the model parameters is also rigorously addressed by the proposed processing scheme and specific VLSI structures of our processing architecture. One of the very first steps towards any kind of image understanding and object recognition is segmenting pixels into k different groups based on statistical similarity or homogeneity of their neighborhood. Histograms, locally generated at each site, represent empirical distributions and from a statistical point of view are robust and reliable representations. The Jensen-Shannon-divergence[1] compares these empirical distributions to estimated prototypical distributions of each segment; the assignment of a site to a specific cluster is then based on the outcome of this measurement.

2. SEGMENTATION MODEL

Let us assume that a set of image sites $s_i, i = 1, ..., n$ is given. These sites are organized as a regular grid of size $N \times N$, $N \in 2^{\mathbb{N}}$ with an imposed spatial neighborhood system, i.e each site is connected, up to some extent, with neighboring sites (Figure 1). The cluster memberships of a site s_i are encoded by boolean assignment variables $M_{i\nu}, \nu = 1, ..., k$ which are summarized in an overall assignment matrix $\mathbf{M} \in \mathfrak{M} =$

[1] $D_{JS}(v||z) = \frac{1}{2} \sum_i \left(v_i \log \frac{2v_i}{v_i+z_i} + z_i \log \frac{2z_i}{v_i+z_i} \right)$

$\{0,1\}^{\mathrm{n}\times\mathrm{k}}$. So we actually set $M_{i\nu}=1$, if site s_i is assigned to cluster ν. We further impose the constraint that $\sum_{i\leq\nu} M_{i\nu}=1$ holds to avoid multiple cluster assignments per site. Each single site holds a histogram x_i, summarized for all sites by $X=x_1,...,x_n$. With p_ν we denote the probability of cluster ν with respect to the actual image data and with q_ν the prototypical distribution of cluster ν. These two values represent free parameters of the model, which have to be estimated and are denoted by $\Theta=\{p_\nu,q_\nu : 1\leq\nu\leq k\}$ in the sequel.

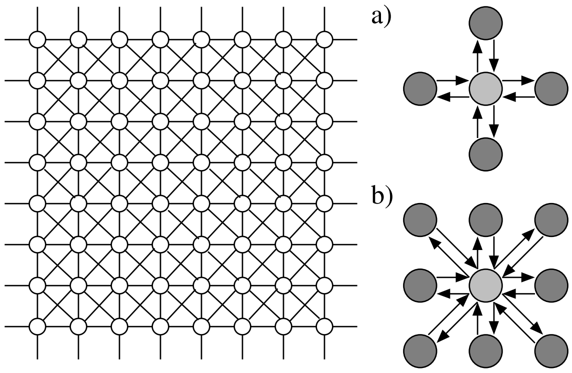

Fig. 1. Site grid with neighborhood system. Neighborhood system of order one (a) and two (b).

The complete data log-likelihood of the model thus is $L(\Theta|X,M)=\sum_i\sum_\nu M_{i\nu}[\log p_\nu + D_{JS}(x_i||q_\nu)]$. From the preceding discussion it becomes clear that we have an alternating 2-phase processing scheme. The well-known Expectation-Maximization Algorithm EM [1] ideally addresses this kind of processing. The inferred solution is further improved by embedding it into the Deterministic Annealing framework. In the first phase (E-Step) the assignment of the sites to clusters will be performed and the equations read:

$$m_{i\nu}=\frac{\exp\frac{1}{T}(\log p_\nu + D_{JS}(x_i||q_\nu))}{\sum_\mu \exp\frac{1}{T}(\log p_\mu + D_{JS}(x_i||q_\mu))}\,. \qquad (1)$$

In the second phase (M-Step) the free parameters p_ν, q_ν are estimated. We derive the following equations for p_ν and q_ν:

$$p_\nu=\frac{1}{n}\sum_{i=1}^{n}(m_{i\nu}),\ \text{and}\ q_\nu=\sum_i x_i m_{i\nu}/\sum_i m_{i\nu}. \qquad (2)$$

For an in-depth presentation and derivation we have to refer the interested reader to additional literature [3,4,2].

3. ARCHITECTURE

Even if we rely on the impressive semiconductor progress to continue in the near future all single serial processing approaches fall short of providing real-time processing capabilities for our unsupervised segmentation model. This deficit is the reason why we introduce a parallel processing scheme and a specialized massively parallel hardware architecture for our model to satisfy the real-time processing needs of many application domains. The next subsections introduce and explain the main building blocks of the hardware architecture derived from the particular processing steps of our model.

Histogram Formation - One processing step is obviously the calculation of the empirical feature distributions (histograms) x_i at each site. We begin with this calculation step since the proposed structure of that building block defines the topological backbone and thus forms the structurally determined part of our hardware architecture. All other building blocks described in the sequel are tightly embedded in this topology. During the histogram formation process the sites s_i consider only values of its neighbors, which are defined by the neighborhood system. At the moment we restrict ourselves to first and second order neighborhood systems (Figure 1) due to the simplicity of VLSI implementation issues. The preceding discussion leads directly to a site structure in our architecture, which is depicted in Figure 1 a,b. As a direct and obvious consequence it follows that each site represents an independent computing unit. Therefore the histogram forming processes can be performed in exactly one δ-time step.

Cluster Assignment - Equation 2 defines the assignment of sites s_i to cluster ν. For this calculation we require to calculate the Jensen-Shannon divergence $D_{JS}(x_i||q_\nu)$ between local histograms x_i and the estimated prototypes q_ν. Secondly the estimated cluster probabilities p_ν and the predefined and decreasing parameter T are needed. If we assume for the moment that the free parameters Θ are estimated and available at each site s_i then every site can conduct the cluster assignment processing-step independent from each other. As a direct consequence of the architecture also this processing step can be performed in exactly one δ-time step.

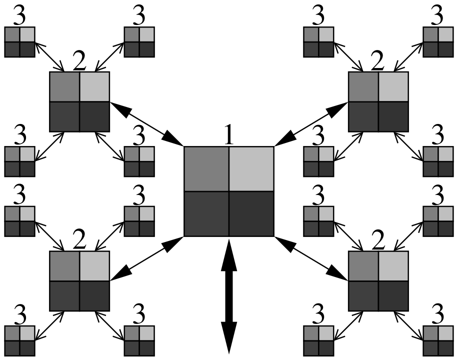

Fig. 2. Distributed memory structure.

Data Distribution - Up to now one serious architectural question remains unanswered: How can we efficiently distribute data to the sites and collect the data back from the sites? The derived building block, which answers this question, also plays an essential role by estimating the global model parameters described in the next subsection. In this contribution we advocate a split and merge distributed memory approach to distribute to and collect the data from the sites. The serial data stream of the image, coming from a sensor or a buffer memory, is split up and stored in four independent memory banks at level 1 (Figure 2). Each of these four memory banks at level 1 further distributes its contents to four memory banks at level 2 and repeats this on all subsequent levels. The same procedure takes place in the opposite direction when the data is collected from the sites. The data distribution and collection is performed at the same time. This scheme guarantees a continuous flow of data to and from the sites with a logarithmically bounded (number of levels) data latency. The proposed architecture requires N^2 δ-time steps to store the data serially at level 1, $\frac{1}{4}N^2$ δ-time steps at level 2 and so on. Thus $\frac{4}{3}N^2$ δ-time steps are needed down the complete hierarchy. The proposed structure is depicted in Figure 2 and a VLSI implementation in Figure 5.

Parameter Estimation - The estimation of the free model parameters, summarized by Θ, is a global process with parallel features, as can be seen in eq. 2. The global nature of the computation arises from the sums over all sites in the statistics p_ν, q_ν. This sum, however, can be split up into local sums which are calculated in parallel. With the earlier described distributed memory structure at hand, an ideally balanced processing with respect to the serial and parallel nature of the free parameter estimation process is achieved. We just have to equip each of the memory banks at every level with the required processing capabilities. To be more precise for p_ν: at the lowest level of the distributed memory structure we count the assignments to a cluster ν of the local sites and store this value in each memory bank of the lowest level. At the next level we just have to sum up the assignment counts of the level beneath and so on. The highest level also performs the summing up of the assignment counts of the level beneath and additionally the division by the overall number of sites. This finalizes the calculation of p_ν; after which p_ν flows along the levels of the distributed memory hierarchy down to the sites. The calculation scheme of the dividend of q_ν is similar to that of p_ν and the divisor of q_ν is already available at the highest level of the distributed memory scheme by the calculation of p_ν. The result of q_ν is also distributed over all levels down to the individual sites. The processing at each single level of the hierarchy is performed in exactly one δ-time step and in $2\text{log}\mathcal{N}^2$ δ-time steps for traversing the hierarchy up and down.

4. RESULTS & CONCLUSIONS

Results - From the preceding δ-time step complexity discussion it follows that the overall processing complexity is dominated by the data distribution term. However we have to be very careful when interpreting this result and rushing to conclusions. We state that the data distribution task represents a pure data transportation process, which can be implemented clock-efficiently in VLSI structures. Whereas for instance the cluster assignment process - one δ-time step - involves complex calculations with respect to VLSI structures and therefore operates less clock-efficiently. With this background the δ-time step result of the data distribution process appears differently in our case and will not affect the real-time processing capabilities of the proposed VLSI architecture. The proposed unsupervised segmentation model as well as the suggested massively parallel processing scheme and its corresponding architecture were intensively tested on natural images in a recently presented novel modeling and simulation environment [5]. Some results of this particular simulation series are depicted in Figure 3. These results are comparable with state-of-the-art results of [2]. To further underpin and prove the practical respectively industrial relevance of

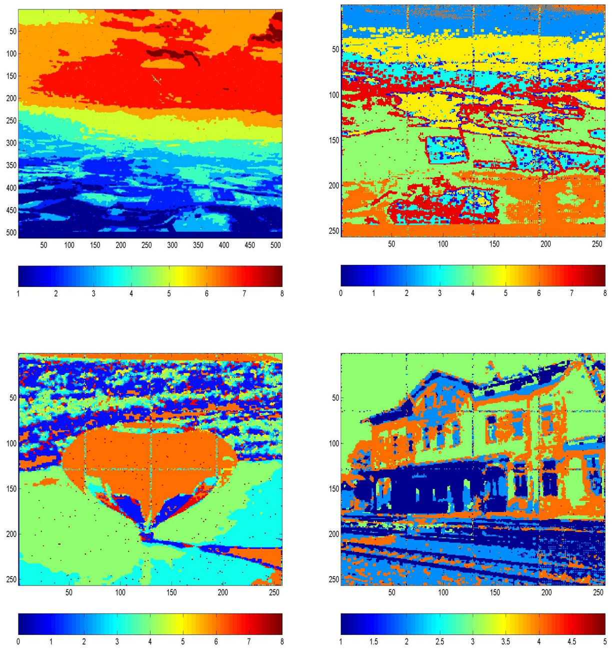

Fig. 3. Segmentation results of four different real-world images.

the proposed unsupervised segmentation model and its massively parallel hardware architecture, detailed VLSI implementations of the site topology and the distributed memory structure have been realized (Figure 5, 6). The processing of real-world images (typically 256×256 or 512×512) with the described model on standard computers takes several ten seconds to several minutes. In contrast our proposed architecture requires only about a 1/10-second to process an image and possesses thus real-time processing capabilities. Exhaustive and detailed performance studies of the proposed architecture - realized in specific semiconductor technologies - are currently under preparation.

Conclusion - In this paper we have described an unsupervised segmentation model with state-of-the-art performance, which is ideally suited for a massively parallel processing schemes on a regular pixel grid. The corresponding massively parallel architecture was introduced and VLSI implementations prove the concept and industrial relevance of the suggested architecture.

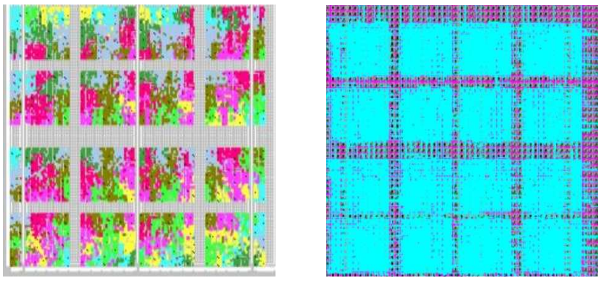

Fig. 4. VLSI floorplan of site grid with 2. order neighborhood system. Site subclusters of size 4×4 are artifically separated for illustration.

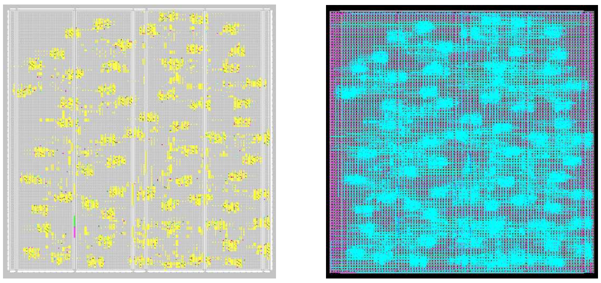

Fig. 5. VLSI Place&Route result of distributed memory structure of grid-size 64×64.

5. REFERENCES

[1] A.P. Dempster, N.M. Laird, and D.B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm", *J. of the Royal Statis. Soc. B, 39:1-38* 1977.

[2] Thomas Zöller, Lothar Hermes, and Joachim M. Buhmann. "Combined Color and Texture Segmentation by Parametric Distributonal Clustering", *Int. Conf. on Pattern Recognition (ICPR'02),* 2002.

[3] K. Rose, E. Gurewitz, and G. Fox. "A deterministic annealing approach to clustering", *Pattern Recognition Letters,* 11:589-594,1990.

[4] J. Puzicha, T. Hofmann, and J.M. Buhmann. "Histogram clustering for unsupervised segmentation and image retrieval", *Pattern Recognition Letters,* 20:899-909, 1999.

[5] Stephan C. Stilkerich. "MRF-Simulation- and MRF-SoC-Development-System", *Int. Signal Processing Conf. (ISPC'03)* Texas, USA 2003.