A Load Balancing Approach for Silicon Retina based Asynchronous Temporal Data Processing

Christoph Sulzbachner, Jürgen Kogler
AIT Austrian Institute of Technology
Safety & Security Department
Safe and Autonomous Systems
Donau-City-Strasse 1, 1220 Vienna, Austria
Email: {christoph.sulzbachner, juergen.kogler}@ait.ac.at

Abstract

In this paper we present a load balancing approach for Silicon Retina technology based computer vision applications. The Silicon Retina technology is a new kind of bio-inspired analogue sensor that is derived from the human vision system. In comparison to conventional imagers that provide frame information captured by a fixed frame-rate, Silicon Retina imagers only provide information of pixels with variations of intensity in a scene. The high amount of pixels without intensity variations need neither be transmitted nor processed. Due to these special characteristics, the imager delivers asynchronous data with data-rates up to a peak of 6M events per second (Meps) per channel and a time resolution of 10ns. A distributed embedded system consisting of a single-core processor for data acquisition and load balancing and a multi-core processor for data processing is used.

We discuss both the Silicon Retina technology, the principles of the computer vision algorithms being used, and the load balancing approach.

1 Introduction

The growing data-rates for reliable advanced driver assistance systems (ADAS) increases steadily. The EU-funded project ADOSE[1] (reliable *a*pplication-specific *d*etection of road users with vehicle *o*n-board *se*nsors) is focused on cost-effective sensor technologies for ADAS [7]. Our aim is to develop a stereo vision system for pre-crash warning/preparation side impact detection using two Silicon Retina imagers and a distributed digital signal processing embedded system [3].

Essentially for the high performance of computer systems, especially embedded digital signal processing systems is the underlying very large scale integration (VLSI) technology, which allows increasing clock-rate and the number of components on a chip. The basic principle of this success story is parallelism. A higher number of parallel resources on a processing unit means that more instructions can be processed at the same time assumed that there are no dependencies [4].

To improve the performance of a single processor system, the compiler's task is to optimize the code by functional inlining, loop transformations or code reordering to minimize instruction and data dependencies. On the other hand the developer need to understand the compiler and create code in a sufficient quality.

In distributed embedded systems tool support is limited. Generally, in distributed systems more than one processor are available for processing. To improve the performance of these systems, loads need to be balanced among these parallel and distributed processors to maximize data throughput.

The remainder of this paper is outlined as follows: Section 2 gives an overview of load balancing. Section 3 introduces the optical sensor and describes its characteristics and the stereo matching algorithm. Section 4 gives an overview of the embedded system and the processor units for acquisition, load-balancing and algorithm processing. Finally, we give a conclusion about the work.

2 Related Work

Culler and Singh [4] describe the job of parallelization at a high level by identifying the parallel processes, determining how to distribute the work, managing the

[1] `http://www.adose-eu.org`

data access, communication and synchronization. Following Culler, the job of creating a parallel program from a sequential one consists of the following four steps:

- Decomposition means breaking up the computation into a sum of arbitrarily defines pieces of work. An upper bound of parallelism is given by the maximum number of processed available at a time.
- Assignment is the step of balancing the workloads among processors, to reduce the intercommunication overhead and to charge the capacities of the available processing units.
- Orchestration means mechanisms for exchanging data and synchronize one another during data processing.
- Mapping means assigning processes to dedicated processors. Depending on the operating environment and system this step can be statically or dynamically assigned.

The maximum speedup factor is limited by the time required for the longest sequential fraction of an application. Equation 1 shows Amdahl's law [4] which defines the speedup of a parallel system with a number of processors (p) and a sequential execution factor (s).

$$speedup = \frac{1}{s + \frac{1-s}{p}} \tag{1}$$

Sharma et. al. [13] present a performance analysis of various static and dynamic load balancing algorithms based on different parameters. In static load balancing approaches the processors are assigned to the processes before runtime. A master processor partitions the total load to slave processors that are used for data processing. In dynamic load balancing approaches a master processor assigns a processor during runtime. Unlike static approaches dynamic load balancing algorithms allocates processes dynamically.

Rahmawan and Gondokaryono [6] handle a simulation of static load balancing algorithms in their work. They cover Round Robin, Randomized, Central Manger, and Threshold algorithm approaches.

3 Optical Sensor

The Silicon Retina technology goes back to 1970, where Fukushima et. al. [10] developed a first electronic model of a retina. Mead and Mahowald [2] first implemented a retina imager on silicon basis in 1988.

In ADOSE, we are both using a low-speed imager with a resolution of 128×128 pixel and a time-resolution of 1ms, and a high-speed imager with a resolution of 304×240 pixel and a time-resolution of 10ns [12, 11].

3.1 Address Event Concept

The Silicon Retina technology is based on the address-event-representation (AER) of information. In AER, an event $\mathbf{E}$ is a tuple of a time-stamp $\mathcal{T}$, the coordinates $\mathcal{X}$ and $\mathcal{Y}$ and the polarity $\mathcal{P}$ of the event. Due to the asynchronous characteristic of the Silicon Retina, a precise time-stamp $\mathcal{T}$ is required for data processing, alike the frame-rate of conventional image processing. The polarity $\mathcal{P}$ indicates whether an event detects increasing illumination (on event) or decreasing illumination (off event).

For data exchange and to minimize the traffic, a compressed approach exists that allows transmitting the time-stamp $\mathbf{E}(\mathcal{T})$ separate for each event $\mathbf{E}(\mathcal{X}, \mathcal{Y}, \mathcal{T}, \mathcal{P})$. Thus, events with the same time-stamp need to transmit their time-stamp only once. An empirical data-rate estimation shows that the average data-rate of the high-speed imager is about 2M events per second (eps) and the maximum data-rate is about 8Meps.

3.2 Stereo Approach

Due to the special characteristics of the Silicon Retina technology, novel approaches for signal processing are required to exploit the potential. The challenge of stereo vision is the reconstruction of depth information of a scene. Scharstein and Szeliski [5] present both a taxonomy of existing algorithms and a testbed for quantitative evaluation of stereo algorithms for area-based approaches. Shi and Tomasi [9] give an overview of feature based image processing and Tang et. al. [1] cover feature matching. In conventional stereo vision, there exist various approaches for feature-based and area-based stereo matching.

An evaluation of area- and feature-based approaches for Silicon Retina based stereo matching showed that these approaches generally can be applied, but the characteristics of the Silicon Retina cannot be exploit. These algorithms were applied to visualized images that contain AER over a certain time-period Δt [8].

New stereo matching approaches are applied on AER data. The acquired AER from the left and right imager are timely and locally correlated to each other to find the pixel correspondences. In contrast to the existing approaches, this novel approach works completely asynchronous. Figure 1 shows the principles of the algorithm. Based on the calibration of the imagers, the coordinates of each event is rectified and lens undistorted. Afterwards each of the events is matched to the history of events from the opposite side and the correlation result is weighted. The weighted data is aggregated for finding a maxima that allow calculating the disparity of an event for generating a disparity map.

4 Embedded System

The embedded system used for data acquisition and processing is shown in Figure 2. Both imagers are connected to an adapter-board that implements a buffered interface that is memory-mapped to the C6455 digital signal processing starter kit (DSK). The adapter-board is connected to the external memory interface (EMIF) of the processor. The DSK partitions and transmits the acquired data to the evaluation module (EVM) over Serial RapidIO (SRIO).

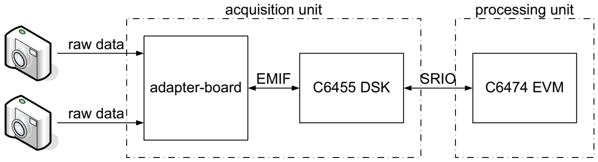

Figure 2. Embedded System

4.1 Target Platform

The embedded system for data acquisition and preprocessing is based on a TMS320C6455 single-core fixed-point DSP from Texas Instruments. The system for data processing is based on a TMS320C6474 multi-core fixed-point DSP from Texas Instruments. Both devices are based on the C64x+$^{\text{TM}}$ core.

The TMS320C64x+$^{\text{TM}}$ architecture from Texas Instruments consists of two data paths (A and B), each of which contains four functional units (.L, .S, .M, and .D) and 32 32-bit general-purpose registers. The .M units are used for multiplications, the .D units are used for load and stores, the .S units are used for shifts, branches and compares and the .L units are used for logical and arithmetic operations. The TMS320C64x+ DSP Megacore Reference Guide [15] gives a detailed overview.

4.2 Data Acquisition

The imager consists of a parallel interface similar to an asynchronous memory interface. Time-stamping of the AER is done in hardware in the imager after an event has occurred. Thus, uncorrectable distortions of the time-stamp of an event are equal to all events and minimized.

In conventional image acquisition, the amount of data is known and a technique can be optimized for acquisition. Due to the variable amount of data from the Silicon Retina imagers, the acquisition system need to be able to acquire the maximum amount of data transmitted by both imagers. As soon as enough data has been fetched by the adapter-board for a channel, an interrupt is initiated to indicate the DSP to start a direct memory access (DMA) transfer that flushes the buffers. To allow the DMA to work on both channels separately, each imager needs to be mapped to an own memory segment. The EDMA3 controller [14] of the TMS320C64x+$^{\text{TM}}$ supports a linking mechanism for automatically reloading a DMA context for the following transfer. Linking allows to parameterize a multi-buffered input. As soon as a transfer has completely finished, the CPU is notified and the pre-processing can be applied.

4.3 Analysis of the Stereo Approach

For processing the data on a multi-core system, the sequential fraction of the algorithm is of interest. In the case of the timely and locally correlated stereo matching approach, each sequential fraction is shown as a box in Figure 1. Each fraction can be optimized for a single-core system using known optimization techniques. Table 1 shows the sequential fractions of the stereo matching algorithm and their data dependencies for processing individual data (AER), access of internal data structure e.g. history of AER, and external data structures.

Fraction	Individual	Int. Data	Ext. Data
Rectification	r	-	-
Matching	r	rw	-
Weighting	-	rw	-
Aggregation	-	rw	-
Find Maxima	-	rw	-
Generate Disparity	-	r	w

Table 1. Data Dependencies of the stereo matching approach. The types are r for read, w for write, and rw for read and write access

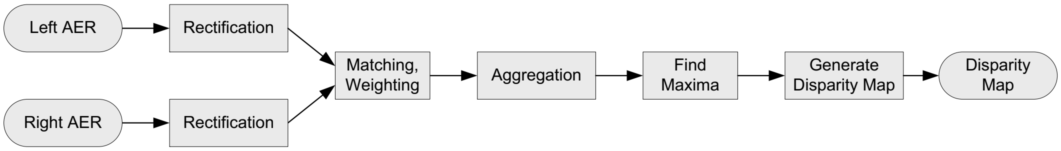

Figure 1. Silicon Retina Stereo Matching Algorithm

4.4 Load Balancing

Load balancing of a system can be centered on paralleling work, or data. In most cases work and data decomposing is strongly related and cannot be distinguished. In ADOSE, the stereo matching algorithm is processed by all cores of the multicore processor. Thus, the load balancer does not partition processes on the processors, however it partitions the amount of data.

Equation 2 shows the identification of the destination core n depending on $\mathbf{E}(\mathcal{Y})$, the number of parallel units N, and the height of the optical sensor H.

$$n = \frac{\mathbf{E}(\mathcal{Y})N}{H} \tag{2}$$

Due to the event atomic design of the algorithm, loads can be balanced fair and the speedup factor is restricted by the number of processors.

The compressed AER data format separates between an event $\mathbf{E}_i$ and a time-stamp $\mathbf{t}_i$, where i is a continuing identifier. A time-stamp $\mathbf{t}_i$ is relevant for all processors and sent to all processors. An event $\mathbf{E}$ is only sent to processor n, shown in Equation 2.

The balanced data is not sent directly to the slave processors after balancing. It is handled in buffers to afford sending in a burst using a DMA transfer.

4.5 Memory Management

Each processor has read access to its individual data and all internal data structures. However, a processor only has write access to an individual data space of the output. Due to the architecture of the multi-core DSP, all cores have access to the same memory. Thus, accessing data over cores does not have much overhead irrespective synchronization.

The advantages of data level parallelism are a predictable processing load of the processors and scalability. Furthermore the source code can be developed in a simpler development process. In the case of Silicon Retina applications, predictable processing load assumes on uniformly distributed data.

For data exchange from the master to the slave processor Serial Rapid IO$^{\mathrm{TM}}$ is used. Serial Rapid IO$^{\mathrm{TM}}$ is an electronic communication standard, which affords a reliable, high-performance packet switched interconnection technology for chip-to-chip and board-to-board. On the architectural hierarchy Serial Rapid IO$^{\mathrm{TM}}$ consists of three layers: The logical layer specifies the protocols, the transport layer defines addressing schemes, and the physical layer contains the device interface. It is optimized for data exchange in embedded systems. The advantages are a low overhead, variable packet sizes and data transfer sizes up to 4kiB. Higher amounts of data need to be sent in 4kiB chunks. The symbol-rate per pair of Serial Rapid IO$^{\mathrm{TM}}$ is up to 3.125Gbaud. In the C64x+$^{\mathrm{TM}}$ architecture, the Serial Rapid IO$^{\mathrm{TM}}$ peripheral device has its own DMA engine. Thus, it is directly connected to the switched central resource of the core and has access to the memory. For transferring data no CPU is required. Serial Rapid IO$^{\mathrm{TM}}$ affords a master processor to directly access the memory of a slave processor without any interaction of the slave. It is also possible for the master to trigger interrupts on the slave device.

The master processor has a double buffered output stage, where the balanced events are stored. As soon as a buffer is full, the double-buffer is switched to the other segment and the other one is initiated to be transmitted. Due to the maximum data-rate of the imagers, it can be guaranteed that the double buffer is flushed before it need to be switched back to the initial position.

The slave processor also has a dedicated memory segment that can be used as an input buffer with the size L that can be access over Serial Rapid IO$^{\mathrm{TM}}$. Using the maximum burst size $L/4kiB$ buffers are available. The master processor writes data to the next segment of the multi-buffer and triggers the specific processor to process the pending data. Due to the timing constraints it can be guaranteed that the ring buffer cannot be overflowed.

5 Results and Conclusion

This paper presented a load balancing approach for processing asynchronous temporal data from Silicon Retina imagers for a stereo vision application. For load balancing work and data parallelism techniques exist.

Due to the complexity of the algorithm a data parallelism approach was used. The embedded system used for data acquisition, load balancing and processing is a distributed digital signal processing system based on a single- and a multi-core DSP.

The concept is restricted to processors on a chip, because it is assumed that all cores have access to a shared memory. Further performance improvements are available by processing the rectification step before load balancing on the acquisition system.

Acknowledgment

The research leading to these results has received funding from the European Community's Seventh Framework Program (FP7/2007-2013) under grant agreement $n°$ ICT-216049 (ADOSE).

References

[1] B. Tang and D. Ait-Boudaoud and B. J. Matuszewski and L.-K. Shark. An Efficient Feature Based Matching Algorithm for Stereo Images. *Proceedings of the IEEE Geometric Modeling and Imaging Conference (GMAI)*, 2006.

[2] C. Mead and M. Mahowald. A silicon model of early visual processing. *Neural Networks Journal*, 1:91–97, 1988.

[3] C. Sulzbachner and J. Kogler and E. Schoitsch and W. Kubinger. A 3D Event-Based Silicon Retina Stereo Sensor . *ERCIM News*, 79, 2009.

[4] D. E. Culler and J. P. Singh. *Parallel Computer Architecture – A Hardware/Software Approach.* Morgan Kaufmann, 1999.

[5] Daniel Scharstein and Richard Szeliski. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. *International Journal of Computer Vision*, 47:7–42, 2001.

[6] H. Rahmawan and Y. S. Gondokaryono. The Simulation of Static Load Balancing Algorithms. *International Conference on Electrical Engineering and Informatics*, 2009.

[7] J. Kogler and C. Sulzbachner and E. Schoitsch and W. Kubinger. ADOSE: New In-Vehicle Sensor Technology for Vehicle Safety in Road Traffic. *ERCIM News*, 78, 2009.

[8] J. Kogler and C. Sulzbachner and W. Kubinger. Bioinspired stereo vision system with silicon retina imagers. *International Conference on Computer Vision Systems (ICVS)*, 2009.

[9] J. Shi and C. Tomasi. Good Features to Track. *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference (CVPR)*, 1994.

[10] K. Fukushima and Y. Yamaguchi and M. Yasuda and S. Nagata. An Electronic Model of Retina. *Proceesings of IEEE*, 58(12):1950–1951, 1970.

[11] M. Hofstätter and P. Schön and C. Posch. An integrated 20-bit 33/5M events/s AER sensor interface with 10n time-stamping and hardware-accelerated event pre-processing. *IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2009.

[12] P. Lichtsteiner and C. Posch and T. Delbruck. A 128×128 120 dB 15 μs Latency Asynchronous Temporal Constrast Vision Sensor. *IEEE journal of solid-state circuits*, 43(2):566–576, 2008.

[13] S. Sharma and S. Singh and M. Sharma. Performance Analysis of Load Balancing Algorithms. *World Academy of Science, Engineering and Technology*, 38, 2008.

[14] Texas Instruments. TMS320C645x DSP Enhanced DMA (EDMA3) Controller User's Guide, Jan. 2007. literature number: spru966b.

[15] Texas Instruments. TMS320C64x+ DSP Megamodule Reference Guide, Aug. 2008. literature number: spru871j.