2225 High Speed Machine Vision Sensing of Cotton Lint Trash

M. G. Pelletier
Agricultural Engineer
USDA, ARS
Cotton Production and Processing Research Unit
Lubbock, TX

Abstract

As machine design in the cotton ginning industry advances, the trend is towards systems that allow for dynamic adjustment of cleaning capabilities. There is however, a lack of real-time sensors suitable for detection of the trash content that would allow for determination of the trash content before the machine performs the cleaning. Current state of the art sensors typically sense the product's trash levels after the cleaning has already taken place or so far before the cleaning takes place, it's impossible to synchronize the cleaning to the lint being cleaning. This adhoc trash level sensing dictates that a large safety buffer is utilized to avoid under-cleaning the cotton in an effort to avoid commodity discounts that occur when the lint is under-cleaned. This research estimates improvement in efficiencies of upwards of 30% are possible, if the trash content was known a-priori via a real-time sensor that adjusts the machine to the actual lint being cleaned, as this would allow for a significant reduction in the dead-band safety buffer. One of the main hurdles to sensing the trash content dynamically is that current image processing systems are too slow to provide the information in time to adjust the industry's new adjustable machines. This research examines the development of a new signal processing algorithms suitable for use in massively parallel vector processors. Performance tests of the new filter showed a processing speed gain in excess of 20 times faster resulting in a system that is capable of processing full 1024x1024 pixel image in less than 17 ms. At this speed, the image processing system's performance is now sufficient to provide a system that would be capable of real-time feed-back control that is in tight cooperation with the cleaning equipment.

Introduction

In the cotton ginning industry, the knowledge of the cotton grade during the processing gives the cotton ginner a unique advantage to reduce the amount of lint lost during the lint-cleaning stage. This advantage becomes more advantageous when the information, regarding the amount of trash in the cotton, is coupled with one of the new lint cleaning machines that have the ability to dynamically adjust how much cleaning they perform on the cotton, figure 1. By dynamically measuring the cotton trash content in real-time, the system can be automatically tuned to avoid wasting valuable lint as well as preventing fiber damage, from over-cleaning, which leads to a higher quality product and hence a more valuable commodity. This is beneficial to all aspects of the industry. New developments into the online classing of cotton have seen the deployment of custom cotton classing systems that utilize computer vision system to accomplish this task. There is however a lack of real-time sensors suitable for detection of the trash content that would allow for determination of the trash content immediately before the machine performs the cleaning. The lack of suitable sensing creates a disconnect between the cotton that's measured to the cotton that's being cleaned. Current state of the art sensors typically sense the product's trash levels after the cleaning has already taken place or so far before the cleaning takes place, it's impossible to synchronize the cleaning to the lint being cleaned. This disconnect results in the users compensating through an increased dead-band range that in effect severely under-utilizes the dynamic cleaning technology. This research was undertaken in the interest of developing new technology that is capable of analyzing the cotton fast enough to provide an immediate response to the cleaning equipment so the cotton that is being analyzed is the cotton that's being cleaned.

Background

To date a limited amount of computer vision has been accomplished on general purpose microcontrollers due to the lack of computing power and the necessary high data transfer rates. Typical microcontrollers run at 20MHz or less with a few exceptions running at 40MHz-80MHz. In addition to these reduced CPU speeds, the microcontrollers are further limited by being designed as an 8 or 16 bit processors with only a general purpose instruction set and limited arithmetic logic unit (ALU). Under these constraints a computer vision application would take minutes if not hours to complete a single complex image analysis. They simply don't have the computational capability that an image processing system requires. This fact has kept the bulk of computer vision applications tethered to the PC or a high end DSP platform that consists of a DSP core with either a video frame-grabbing chipset or an FPGA based vector processor. The alternative to the traditional 8bit microcontroller for an embedded platform ranges from custom DSP boards to pc104 Pentium boards (embedded PC's) to custom hardware circuitry designed into either application specific circuits (ASICS) or high end programmable logic devices such as FPGA's (Buck et al., 2004; DeCoro and Tatarchuk, 2007; Galoppo et al., 2005; Kipfer et al, 2004; Kolb, 2005; Krueger and Westermann, 2003; Larsen and McAllister, 2001; Scheuermann and Hensley, 2007; Strzodka et al, 2005). In recent years however, a new alternative has emerged from the computer graphics arena targeted primarily at the PC based gaming industry. These new graphic processors "GPUs" now have numerous CPU's designed to run in parallel. While these GPUs have been primarily designed for rendering of games, for some specific applications that are inherently parallel in nature, they provide a much faster alternative to computing than the traditional PC CPU, figure 2. This paper presents the development and testing of an algorithm for cotton trash imaging for use on one of the latest GPUs.

Methods and Results

In the interest of developing a rapid analysis system, a new machine vision processing algorithm has been developed that was designed to provide a highly parallel approach to the cotton trash identification problem. As the cotton trash system in practice is, in many instances, a retrofit system that is placed into existing systems, it's been found that controlled lighting is less controlled than would be considered optimal for alternative algorithms such as a pre-computed lookup table based on a Bayesian Classifier approach, Pelletier 1999a,b, 2003. The problem with the Baye's Classifier approach is due to the need for pre-computation of the Bayesian statistics, as the technique demands that a subset of expertly classed pixels are utilized to calculate the statistics. Thus, it's not possible to dynamically adjust the Bayesian statistics. Given the need for stable image statistics, this requirement places a demand on the system that the image statistics remain reasonably static. In practice it was found, in the deployment of the system into several commercial installations, that the static-image-statistic's criteria was not valid. In practice, the changing lighting conditions and system placement as a retrofit onto various types of machines, creates wide variation in the image statistics for each member of the feature set {trash, background, lint; all of which are in full or partially lit areas or immersed in shadows}. To compensate for this, an alternative image processing algorithm was developed to overcome the difficulties of the varying statistics with an additional goal of also providing a highly parallel algorithm suitable for use on a highly parallel vector processor.

The basic overview of the image processing algorithm, figure 3, shows the steps required to process the image from raw color pixels into a set of statistics that informs the system of the quantity and type of trash that requires cleaning by the machine; the basic information required by the optimal control system. The start of the image processing algorithm is to process each pixel, by analysis of target pixel against the local neighboring pixels, to determine or classify the pixel into either lint or trash, figure 4. As the bulk of the time required by the image processing algorithm is tied to this first step of pixel identification, the focus of the new development was to optimize the processing of this stage. To replace the Bayesian Classier, a new single-pass filtering algorithm was developed that effectively partitions the color space in a single pass such that a simple threshold operation following the filter operation will allow for the generation of a binary image where each pixel is classed to be either a trash or lint pixel. In practice the new filter was shown to be remarkably robust across a wide variety of lighting situations. As one of the key criteria's for online classing is the ability of the algorithm to separate cotton from trash in both lit and shadowed areas, a suitable test candidate that exhibits both lit areas as well as shadowed areas with wide variations in trash was utilized, an example of the performance of the new Single-Pass-Filter, hereafter known as SPF, performing this task is detailed in figure 5.

To effect a speed-up, one of the main goals of this research, as well as to provide a baseline performance by which to judge the GPU approach, the algorithm was optimized for use on a Pentium 4 processor using the extended operation set, SIMD, that targets the Pentium 4's single vector processor that is capable of multiplying 4 single-precision floating point numbers in parallel. Using this as well inline expanded and optimized C code, resulted in a processing time of 7.5 frames per second; a significant speedup over the previous algorithm of 2.5 frames per second. The next step was to compare this performance to the same algorithm running on an Nvidia GeForce 8800 Ultra where the code would then have the opportunity to take advantage of the GPU's 132 vector processors. We do note here that while the GPU has 132 vector processors, each capable of multiplying 4 single precision floating point numbers in parallel, the core is only running at 500 MHz versus the Pentium's core at 3.0GHz. Other potential problem areas are bottle necks in pushing the image data across the pci-x bus and into the GPU's video ram. In short, one doesn't expect a 132-X gain from running on the GPU core versus normal operations that take place via computation on the CPU.

In moving the algorithm off of the PC's traditional computing platform, the CPU, to the GPU; the CPU passes both the algorithm as well as the image data to the GPU. As traditionally the CPU passes the graphics data to the GPU's video ram along with a list of triangles (vertices), which are then rasterized into individual pixels. At both the vertices stage as well as pixel stage of the stream processing, the GPU can execute custom code to effect rendering effects. In short the GPU's have become highly customizable via the ability to download application specific code to run at each stage of the video processing stream, an over-view of the GPU processing stream is shown in figure 6.

To effect general computation on the GPU, the algorithm much be transformed to fit this highly specialized stream processing. As such only certain algorithms that are inherently parallel in nature, can be converted, however for those suitable algorithms, once the transformation is performed, the massively parallel architecture of the modern GPU's becomes available which enables dramatic increases in performance over the traditional CPU performance. To ease the transformation process, the graphic card developers have developed an augmented C programming language that allows one to specify how the image is to be broken into numerous sub-images that will all be computed concurrently as well as the algorithm each vector processor should execute. By combining both the SPF algorithm with the threshold and non-linear median-filter, for shot-noise reduction, to form a combined SPF-TM analyzer, off of the CPU and onto the GPU, the program now has the ability to break the large image into numerous sub-images that can all be analyzed concurrently. Once the GPU analyzes the data per specification of the SPF - TM, it then transfers the fully analyzed binary image solution, figure 7, back to the CPU along with trash statistics such as trash content. In testing of the SPF-TM algorithm for cotton trash identification, we found the following:

- By transitioning from the Bayesian Classifier to a single pass SPF, along with optimizations of the algorithm, improvements over processing speed was gained.
  - Optimization of algorithm effected a speed up of 2.5 times (7.5 frames/sec).
  -
- By moving the code from the CPU to the GPU and utilizing the combined SPF-TM algorithm, further improvements were gained:
  - When utilizing a single Nvidia Gforce 8800 GPU with 132 vector processors; a speed up of 20 times was gained (60 frames/sec) over the Bayesian Classifier and a speed up of eight was gained over running highly optimized code on the CPU.
- At 60 Frames/second:
  - 110,000 sq-in. cotton can be imaged/bale with the new SPF-TM algorithm running on the GPU.
- versus 2.5 Frames/second, utilizing the previous Bayesian Classifier approach:
  - 5,500 sq-in. cotton can be imaged/bale.

Summary

As the cotton ginning industry moves toward machines that have the ability to dynamically adjust the amount of cleaning the machine performs; a great deal of valuable lint is saved along with a significant reduction in fiber damage. The missing element at this time is the ability of sensors to determine the required amount of cleaning for the cotton as it is feed into the machine. This research has demonstrated that through the use of massively parallel processing, that is now possible on today's programmable GPU's, a machine vision algorithm suitable for real-time classing of cotton in the process stream, can be processed in a significantly reduced time that is sufficient to open the door for the possibility of processing the trash content of the incoming lint in time to set the machine so that it cleans the cotton that was just analyzed. This just-in-time analysis can then provide a system that is optimized to clean the cotton that is being fed into the machine at the optimal cleaning level. Once this transition from the current system, that looks at a sample of cotton taken either way before processing or after the cleaning has already taken place, moves toward one where the machine is cleaning the cotton that was analyzed as it is being fed into the machine; performance gains can be expected to be upwards of 30% improvements. This level of improvement can be expected due to the fact that today's systems use a very large dead-band to protect the users against both the inherent wide variability in the cotton lint's trash distribution, as well as the potential for changes to take place before the machine can react to the changing cotton trash levels. It is also expected that this technology will likely drive new machine designs that can not only optimize the cleaning across the entire width of the machine, which effectively cleans 100% of the cotton to remove the trash from the 4% of the cotton that actually have trash particles, towards a machine that only cleans the 4% of the cotton that actually contains the trash. Once this level of automation is reach, significant reductions in lint loss as well as fiber damage will become possible as there should be an additional 96% reduction in lint loss. As the typical first stage lint cleaners generate upwards of 10-15 lbs of lint loss, if 10 lbs of lint per bale can be saved, this would represent a $100 million of added annual revenue to US cotton growers.

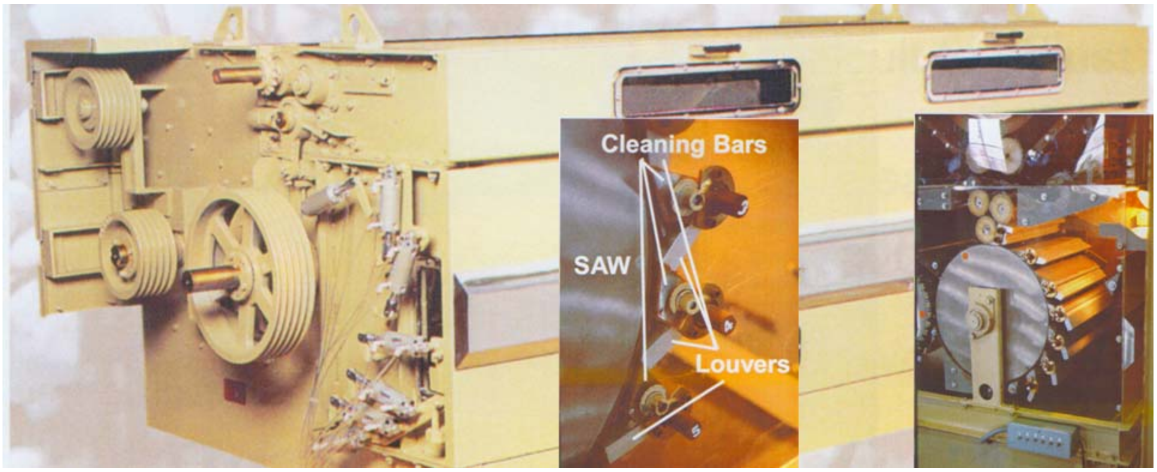

Figure 1: Lint cleaner developed by USDA-ARS that is dynamically adjustable. Photo's courtesy of Continental Eagle Corp.

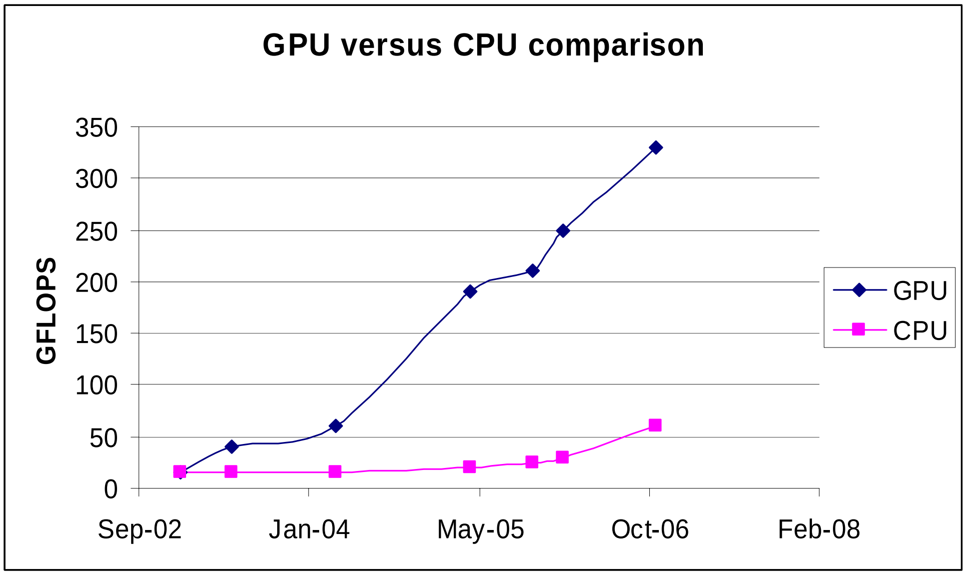

Figure 2: Graph detailing incredible performance increase over the last several years of the GPU over the traditional general purpose CPU on today's modern PC's. (Nvidia 2006)

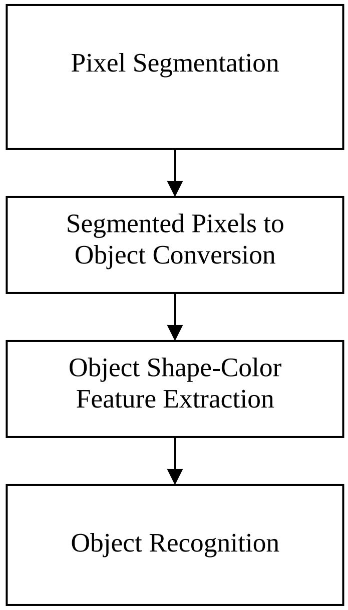

Figure 3: Image analysis to extract the quantities of the various trash constituents.

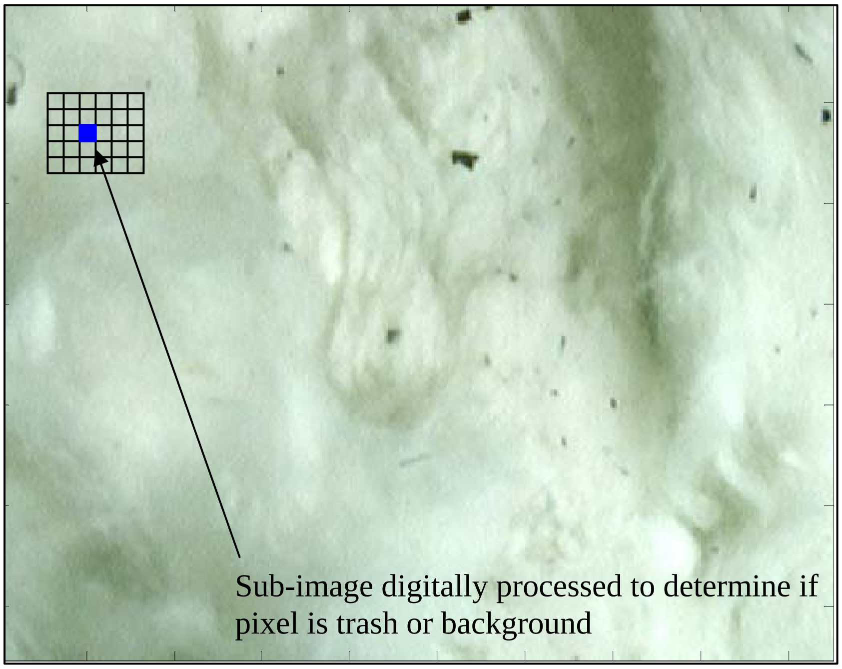

Figure 4: Sub-image is analyzed to determine if pixel is trash or lint.

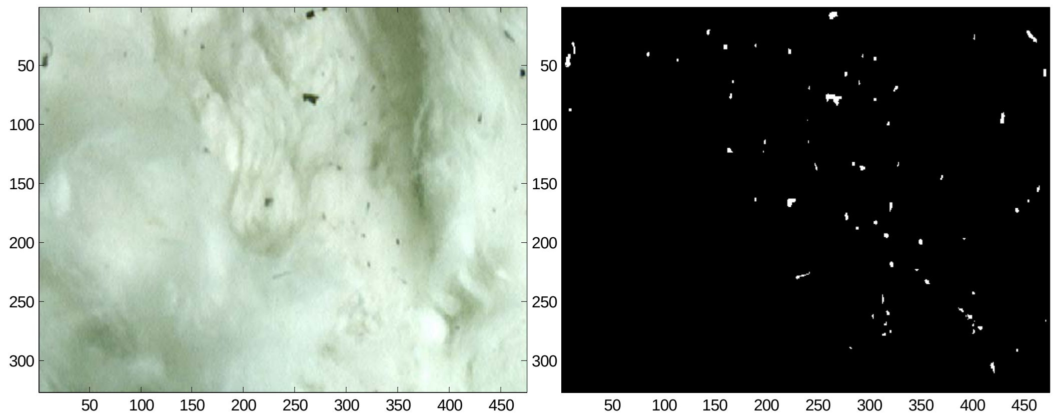

Figure 5: Performance of new single pass GCK algorithm.

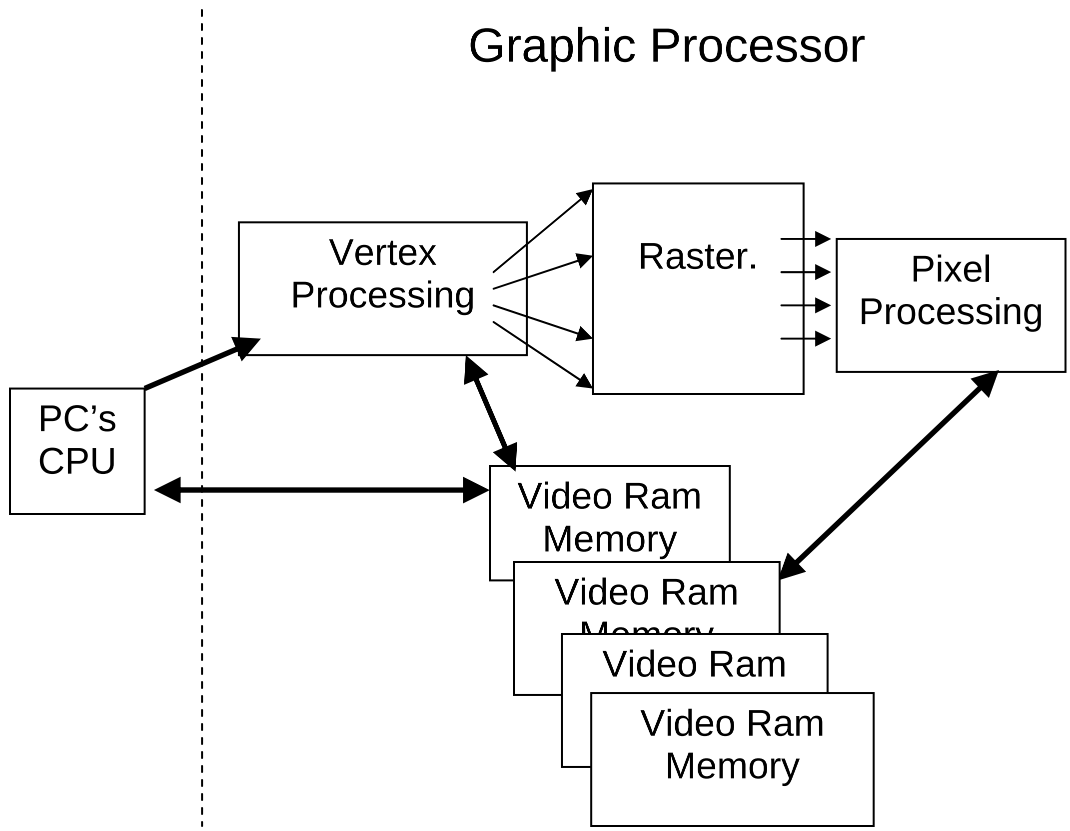

Figure 6: Processing flow using a GPU processor

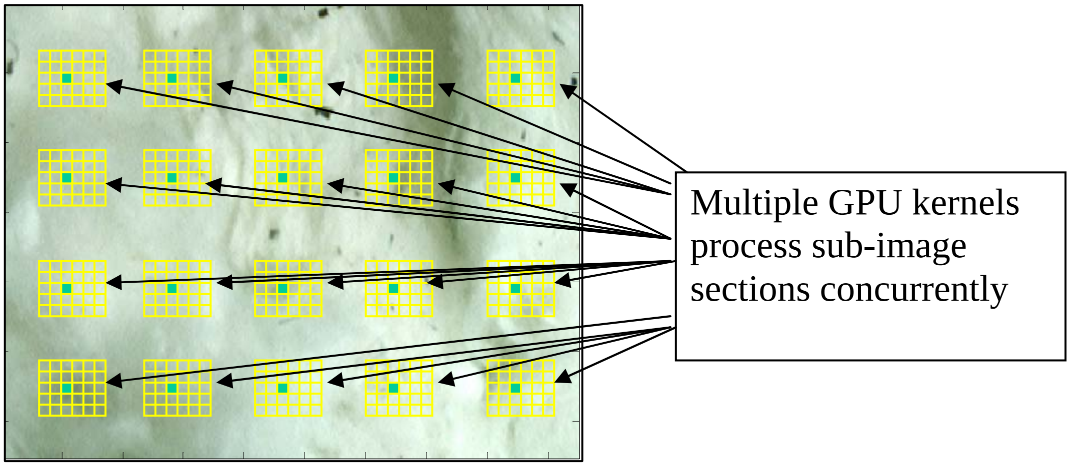

Figure 7: Massively parallel approach to image processing showing how numerous sub-images are all analyzed concurrently using 132 vector processors on a GPU.

References

Buck, I., Foley, T., Horn, D., Sugerman, J., Fatahalian, K., Houston, M., Hanrahan, P., 2004. Brook for GPUs: Stream computing on graphics hardware. In: ACM Proc. SIGGRAPH. Vol. 23. pp. 777–786.

DeCoro, C., Tatarchuk, N. 2007. Real-time Mesh Simplification Using the GPU. To appear in proceedings of *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (SI3D '07)*.

Kipfer, P., Segal, M., Westermann, R., 2004. Uberflow: A GPU-based particle engine. In: Proc. Graphics Hardware. ACM/Eurographics, pp. 115–122.

Kolb, A., 2005. Dynamic particle coupling for GPU-based fluid simulation. In: 18th Symposium on Simulation Technique. To appear.

Kolb, A., Latta, L., Rezk-Salama, C., 2004. Hardware-based simulation and collision detection for large particle systems. In: Proc. Graphics Hardware. ACM/Eurographics, pp. 123–131.

Krueger, J., Westermann, R., 2003. Linear algebra operators for GPU implementation of numerical algorithms. ACM Transactions on Graphics (TOG) 22 (3), 908–916.

Galoppo, N., Naga K. Govindaraju, Michael Henson and Dinesh Manocha: LU-GPU: Efficient algorithms for solving Dense Linear systems on Graphics hardware. Super Computing 2005

Larsen, E.S. and D. McAllister. Fast matrix multiplies using graphics hardware. Super Computing 2001.

Pelletier, M. G., Barker, G. L. and Baker, R. V. 1999a. Non-contact image processing for gin trash sensors in stripper harvested cotton. ASAE Paper #991126. 9 pp.

Pelletier, M. G., Barker, G. L. and Baker, R. V. 1999b. Image processing for stripper harvested cotton trash content measurement: a progress report. Proc. Beltwide Cotton Prod. Res. Conf. pp. 1412-1415.

Pelletier, M. G. 2003. Real time measurement system for seed cotton or lint. U.S. Patent No. 6,567,538.

Scheuermann, T., Hensley, J. 2007. Efficient Histogram Generation Using Scattering on GPUs. To appear in proceedings of *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (SI3D '07*

Strzodka, R., Doggett, M. and Kolb, A. 2005. Scientific computation for simulations on programmable graphics hardware. Simulation Modeling Practice and Theory, Special Issue: Programmable Graphics Hardware, 13(8):667–680, Nov 2005.