

# Low Power Programmable Shader with Efficient Graphics And Video Acceleration Capabilities for Mobile Multimedia Applications

You-Ming Tsao, Shao-Yi Chien, Chin-Hsiang Chang, Chung-Jr Lian, and Liang-Gee Chen, *Fellow, IEEE*

**Abstract--** In this paper, a low power programmable vertex shader with video coding acceleration instructions is proposed. For mobile multimedia applications, supporting both video and graphics is a promising trend. The proposed programmable graphics engine features a unified architecture that can efficiently execute not only vertex shader operations for graphics but also the motion estimation of video coding algorithms. It can achieve the processing speed of 8.3M vertex geometry transformations per second and 6.25M polygons per second with the working frequency of 50 MHz and the power consumption of 20 mW. Furthermore, the floating/fixed-point data path, the reconfigurable memory, and special instructions are designed to be able to accelerate the key operation, motion estimation, in video coding. The execution of motion estimation on the proposed graphics engine is shown to be 80 times faster than RISC type processors and can achieve real-time video coding requirements with diamond search algorithm for 30 CIF (352x288) frames per second with [-16, 15] search range. This powerful graphics and video dual-function programmable engine is shown to be a good solution for multimedia consumer products.

## I. INTRODUCTION

Different from desktop graphics processors, mobile graphics processors operate in resource-limited environments and are power-limited since they are battery-powered. Recently, more and more research works were targeted on mobile graphics processors [1] [2] [3] [4]. Among them, [1] is the first chip which integrates both dedicated geometry engine and rendering engine. However, only fixed graphics pipeline is supported by this chip. The first vertex shader for mobile devices is proposed in [4]. Fixed-point datapath is used instead of floating-point in order to save the power consumption and hardware cost. However, the floating-point data path is still required for precisely rendering complicated scenes. [2] implies that every pixel on a mobile phone should ultimately be rendered with higher quality than on a PC system.

On the other hand, the limited hardware resources also constrain the mobile device. With the video acceleration instruction extension, we could reuse the datapath for motion estimation by reconfigurable memory scheme to make the most hardware resource used efficiency.

## II. EARLY REJECTION AFTER TRANSFORMATION

In this section, the early rejection after transformation (ERAT) architecture is proposed to reduce the computation in geometry stage. Figure 1 shows the overview of the system where the gray part is discussed in this paper. Although the vertex

shader performs shading operations on every vertex, after sending the vertices to the rendering stage, many primitives will be found to be invisible on the screen by the render processor, and a lot of processing power has been wasted on these primitives. If these primitives can be found early in the geometry stage after transformation, the lighting operation, which takes the heavy workload, can be omitted, thus a lot of vertex operations can be saved. Therefore, we propose the early rejection after transformation (ERAT) architecture for this purpose. Three types of triangles should be early rejected right after the vertex shader transforms the vertices form object space to clip space: triangles outside clipping boundary; triangles with zero area, that is, it does not cover any grid point in the screen; and back-faced triangles. The last type of triangle rejection depends on the culling mode decided by the applications. For some applications, this type of triangles should not be rejected from the pipeline. Figure 2 shows the different types redundant triangle percentage rejected by this scheme where the redundant polygon rate is defined as  $(Input\ Polygon\ Number - Rejected\ Polygon\ Number) / Input\ Polygon\ Number$ .

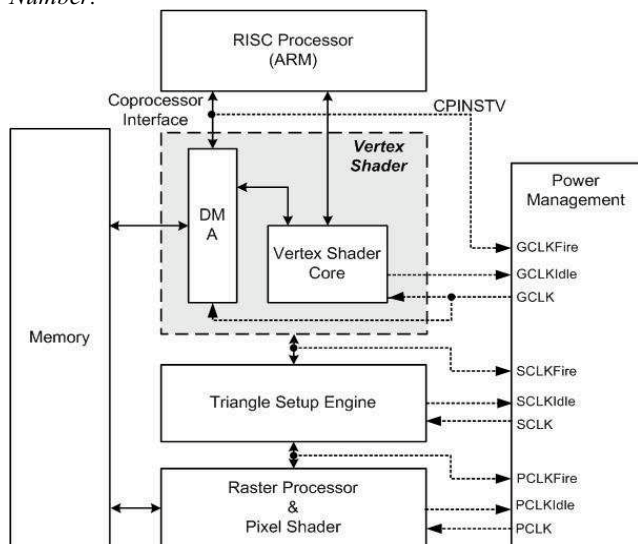


Figure 1: System architecture of the graphic processor.

In average, 30% of the redundant polygons can be rejected in the six test application programs. If the ERAT test performs well, the polygon rate will relatively increase, while more power can be saved by avoiding executing redundant instructions. Figure 3 shows the performance comparison of this work and pervious works by the performance index proposed in [4]. The performance index is defined as  $Vertex\ Process\ Speed\ (vertices/sec) / Power\ Consumption\ (mW)$ .

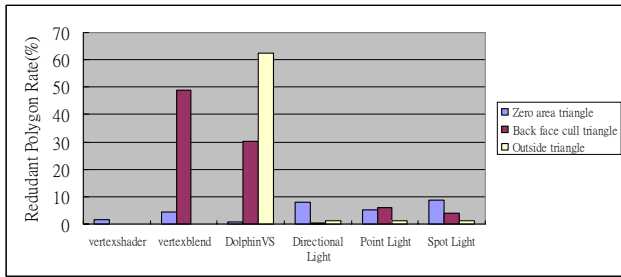


Figure 2: Three types of the redundant triangle rate.

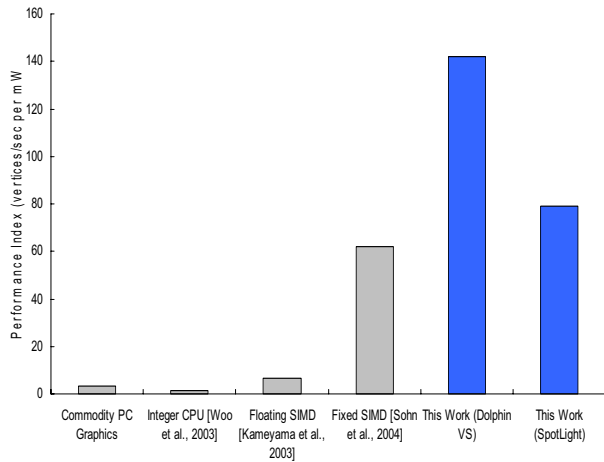


Figure 3: Comparison of different geometry engines.

### III. VIDEO CODING ACCELERATION INSTRUCTIONS SET

A special instruction, sum of absolute difference (SAD), is proposed in the vertex shader to enhance the performance of motion estimation. The SAD value between eight pixels can be calculated with only one instruction. Partial distortion elimination (PDE) is also supported with special hardware and registers. The following vertex program implements the SAD and the PDE operation of one macroblock. The PDE controller is turned on by changing the value of flag register \$F0. Then the loop counter LPCNT3 is set to calculate the SAD value of one macroblock. In each loop, the SAD value of an 8x8 block can be derived, while the PDE hardware will change the flag in register \$\$S12 to indicate if the partial distortion exceeds the current minimum SAD value. If it is true, the program will break the loop. Note that, the numbers in the following program are represented in binary format. The performance of executing ME on the proposed vertex shader is presented in Table 1. Six MPEG-4 test sequences with 100 CIF (352x288) frames are used as test benches. The macroblock size is set to 16x16, and the search range is [-16, 15]. When FSBMA is employed, about 5,000 cycles are required for one frame, which is 80 times faster than RISC type processors [5].

```
Fset $F0 "0000100"
//Turn on the PDE control
Set LPCNT3 "100"
Loop LPCNT3
```

```
SAD V0 C0
SAD V1 C1
SAD V2 C2
SAD V3 C3
SAD V4 C4
SAD V5 C5
SAD V6 C6
SAD V7 C7
Bone $F0 "010"&"$S12"
//If PDE true, break the loop
Inc vOFFSET "01000"
//current block change row
Inc cBASE "1"
//candidate block change row
Loop end
Fset $F0 "0000001" //Return
```

### IV. CONCLUSION

A low power floating/fixed-point programmable vertex shader with video coding acceleration instructions has been developed in this paper. It could achieve the max processing speed of 8.3M vertex geometry transformations per second and 6.25M polygons per second with the working frequency of 50 MHz. In addition, the design is also the first vertex shader core that could accelerate the motion estimation operation. Low power consumption can also be achieved with module and instruction level clock gating and the efficient parallel processing unit. The power consumption is less than 20 mW.

Sequence	FSBMA			
	Avg Cycles	Max Cycles	Min Cycles	Avg PSNR
Coastguard	5,183	6,951	4,124	30.19
Foreman	4,909	5,631	3,656	33.45
Mobile	5,019	5,405	4,641	24.46
Stefan	5,010	5,649	3,096	26.19
Table	5,831	6,860	4,243	30.35
Weather	981	1,543	691	39.3

Sequence	DS				
	Avg Cycles	Max Cycles	Min Cycles	Avg PSNR	PSNR Drop
Coastguard	113	185	92	30.04	0.15
Foreman	148	218	105	33.04	0.41
Mobile	113	127	103	24.21	0.25
Stefan	136	199	89	25.11	1.08
Table	145	218	90	28.45	1.9
Weather	37	55	30	39.15	0.15

Table 1: Performance of executing ME on the proposed vertex shader

### REFERENCES

- [1] KAMEYAMA M., KATO Y.: 3D graphics LSI core for mobile phone "Z3D". In *Proc. Graphics Hardware '03* (2003), pp. 60–67.
- [2] AKENINE-MÖLLER T.: A hardware rasterization architecture for mobile phones. In *Proc. SIGGRAPH '03* (2003), vol. 22, pp. 801–808
- [3] WOO R.: A 210mW graphics LSI implementing full 3D pipeline with 264Mtexels/s texturing for mobile multimedia applications. In *Digest of Technical Papers of IEEE International Solid-State Circuits Conference (ISSCC '03)* (2003)
- [4] SOHN J.-H., WOO R., YOO H.-J.: A programmable vertex shader with fixed-point SIMD datapath for low power wireless applications. In *Proc. Graphics Hardware '04* (2004)
- [5] CHANG H.-C.: Performance analysis and architecture evaluation of MPEG-4 video codec system. In *Proc. International Symposium on Circuits and Systems '00* (2000), vol. 2, pp. 449–452