

University of Mosul
College of Engineering



Accelerated Implementation of Primitive Functions for Computer Graphics

Sura Nawfal Abd-Alrazzaq

Doctor Philosophy Thesis in
Computer Engineering / Computer Graphics

Supervised by
Assist. Prof. Dr. Fakhruddin H. Ali

2020 A.D.

1441 A.H.

Abstract

In this thesis, the main stages of three dimensional (3D) graphics pipeline are modified and accelerated on GPU to deal with millions of vertices in real-time interactive manner. Importing 3D models is the task of the first unit that feeds the pipeline with the extracted vertices.

The first implemented stage is the vertex transform unit which is essential for viewing, animation, and interactive graphics. It requires many matrix operations that are needed to be performed in a real time. Two parallel architectures GPU and multi-core MCCPU, or even both, are used, where all vertices are transformed using one concatenated matrix simultaneously by assigning each vertex attribute to a thread and multiplying them as Single Instruction Multiple Data (SIMD). This implemented stage can be considered as a general design that is capable of processing multiple geometric transformations in any order, in addition to the special rotation about any axis.

The second implemented stage is 3D clipping which is the most complicated one among others. In this work, the Cohen-Sutherland algorithm is modified by adding more cases for early decision as well as for the special cases that can occur in clipping operation. Two granularity sizes are used for parallelism, edge-parallel, and triangle-parallel, the first method considers edges (represented by two vertices each) as a processing element, and the other assigns a triangle to each processor to be clipped simultaneously.

The third implemented unit, that precedes displaying objects, is the voxelization unit. The acceleration of this stage is achieved by using a modified 3D Digital Differential Analyzer (DDA) algorithm for line

voxelization that gives better results in parallel implementation than the standard Bresenham algorithm. The scan-line and the edge-function algorithms are parallelized and implemented for triangle filling; the latter one has been executed at a higher speed than the other one.

These algorithms are implemented in a parallel fashion using GPU. LabVIEW, visual Studio and OpenGL platforms are used to realize the complete implementations with parallel programming language by applying Compute Unified Device Architecture (CUDA). The performance of the designs is evaluated by measuring the execution time and throughput for processing different object sizes.

The experimental results showed that a significant speed-up is gained in comparison with the sequential implementation and with the parallel OpenGL implementation on the same GPU device. The maximum throughput of 1.04G vertex/sec is obtained in 3D transformation, while the throughput in the clipping unit reached 250 M triangle/sec using the triangle-parallel approach that outperforms the edge-parallel method. Finally, the acceleration factor of parallel 3D DDA was about 4352x faster than the OpenGL implementation, and the edge-function approach outperforms the scan-line approach which is capable of voxelizing about 4500 M vertex/sec with 4238x speedup factor with respect to OpenGL execution.

الملخص

اصبحت وحدة المعالجة الرسومية (GPU) من المعالجات الأكثر انتشاراً في الوقت الحاضر, ليس في مجال تقنيات الرسومات الحاسوبية فحسب وإنما أصبحت هذه الوحدة تستخدم لأغراض عامة لتسريع العديد من التطبيقات. ان التطور الكبير في قابليات المعالجات الرسومية وتقنيات تصنيع المعالج نفسه يحتاج الى تعديلات مستمرة في خط نقل المعالجة الرسومية. لذلك تم في هذه الرسالة تصميم وتنفيذ الدوال الرئيسية لمعالجة وعرض الرسومات الحاسوبية الثلاثية الابعاد بالإضافة الى تسريعها على وحدة معالجة الرسومات للتعامل مع ملايين الرؤوس بطريقة تفاعلية وفي الزمن الحقيقي , اذ تحتاج هذه الخاصية الى سرعة عالية لاعطاء المستخدم الاحساس بالواقعية في عرض المشاهد سواءاً على الشاشة او على العارضات الثلاثية الابعاد.

بداية تم تصميم برنامج لادخال النموذج الثلاثي الابعاد واستخلاص الرؤوس منه وذلك لتغذية خط النقل بهذه الرؤوس, بعد ذلك تم تصميم وحدة التحويلات التي تعد من المراحل المهمة للعرض وللرسوم المتحركة والتفاعلية التي تستهلك معظم وقت التنفيذ لأنها تحوي العديد من عمليات المصفوفات التي يجب تنفيذها في الزمن الحقيقي. تم استخدام نوعين من المعالجات في التنفيذ المتوازي لهذه المرحلة وهما المعالج المتعدد النوى (MCGPU) والمعالج الرسومي, اذ تم التصميم بحيث تكون هذه الوحدة قادرة على التنفيذ المتوازي للعديد من التحويلات على الرؤوس في آن واحد باستخدام مصفوفة متسلسلة واحدة. اما من ناحية التوازي, فكان التصميم قادراً على تحويل كل رأس من رؤوس النموذج الرسومي بأستخدام أربعة أشرطة تعمل بصورة متوازية, اي يكون العمل بصيغة تنفيذ ايعاز واحد على بيانات متعددة (SIMD). بالنتيجة , يمكن اعتبار هذه المرحلة المنفذة تصميمًا عامًا قادرًا على معالجة العديد من التحويلات الهندسية بأختلاف تسلسلاتها فضلاً عن التدويرات الخاصة حول أي محور كان.

ان الوحدة الثانية التي تم تنفيذها هي وحدة التقليم, اذ تعتبر هذه الوحدة هي الأكثر تعقيداً من بين المراحل الأخرى فهي تقوم بتقليم الاجزاء الغير المرئية باستخدام خوارزميات خاصة. في هذا العمل تم تعديل خوارزمية كوهين-سذرلاند بأتجاهين, الاتجاه الاول لتكون مناسبة لتقليم النماذج الثلاثية الابعاد بطريقة التنفيذ المتوازي , اما الاتجاه الثاني فهو اضافة خاصية معالجة الحالات الخاصة الممكن حدوثها في عملية التقليم. اما بالنسبة للتنفيذ المتوازي

فتم استخدام معماريتين مختلفتين لذلك ، في الاولى يكون التوازي على مستوى الحواف (رأسان لكل شريط) اما الثانية فتكون المعالجة على مستوى المثلثات (مثلث لكل شريط). اما الوحدة الثالثة فهي وحدة الـ voxelization التي يتم فيها تحويل المضلعات الى نقاط ثلاثية الابعاد لتشكل النموذج الرسومي ككل. تم استخدام خوارزمية 3DDDA المعدلة لتمثيل الخط المستقيم اذ كانت اكثر ملائمة للتنفيذ المتوازي من خوارزمية برزنهام القياسية. كذلك تم تحويل وتنفيذ خوارزميتي scan-line و edge function لملى المثلث بالنقاط الصورية, وكانت الخوارزمية الاخيرة ذات اداء افضل من خوارزمية الـ scan-line.

تم تطبيق كل هذه الخوارزميات باستخدام وحدة المعالجة الرسومية GeForce 1050 و GeForce610 وكذلك MCCPU, اما البيئات البرمجية المستخدمة فقد تم التنفيذ عن طريق برنامج الـ labview و visual studio وتم تحقيق التوازي بأستخدام النموذج البرمجي المتوازي CUDA. اما تقييم أداء هذه التصاميم فقد تم عن طريق قياس وقت التنفيذ وكمية الإنتاجية في معالجة مدى واسع من أحجام الكائنات الرسومية.

اظهرت النتائج التي تم الحصول عليها تحقيق زيادة كبيرة في التسريع مقارنة بالتنفيذ المتسلسل فضلا عن المقارنة مع الأعمال الأخرى السابقة. اذ وصلت الإنتاجية القصوى في تحويل الاشكال الثلاثية الابعاد الى 1.04 مليار رأس في الثانية الواحدة وذلك بسرعة حوالي 5268x مقارنة مع التنفيذ المتوالي ، في حين كانت محصلة الإنتاجية لوحدة القطع حوالي 250 مليون مثلث في الثانية وذلك باستخدام معمارية توازي المثلثات التي تفوقت على معمارية توازي الحافات. أخيرًا ، كان مقدار تسارع تنفيذ خوارزمية 3DDDA أسرع بنحو 4352 مرة مقارنة مع تنفيذ الـ OpenGL ، اما في الـ voxelization فكان أداء خوارزمية دالة الحافة افضل من خوارزمية خط المسح ، اذ كانت قادرة على عرض حوالي 4.5 مليار رأس في الثانية الواحدة بعامل تسريع يصل الى 4238x مقارنة مع تنفيذ الـ OpenGL.



تسجيل تنفيذ الدوال الأساسية في الرسومات الحاسوبية

سُرى نوفل عبد الرزاق

اطروحة دكتوراة فلسفة في
هندسة الحاسوب/الرسوم الحاسوبية

بإشراف

أ.م.د. فخر الدين حامد علي