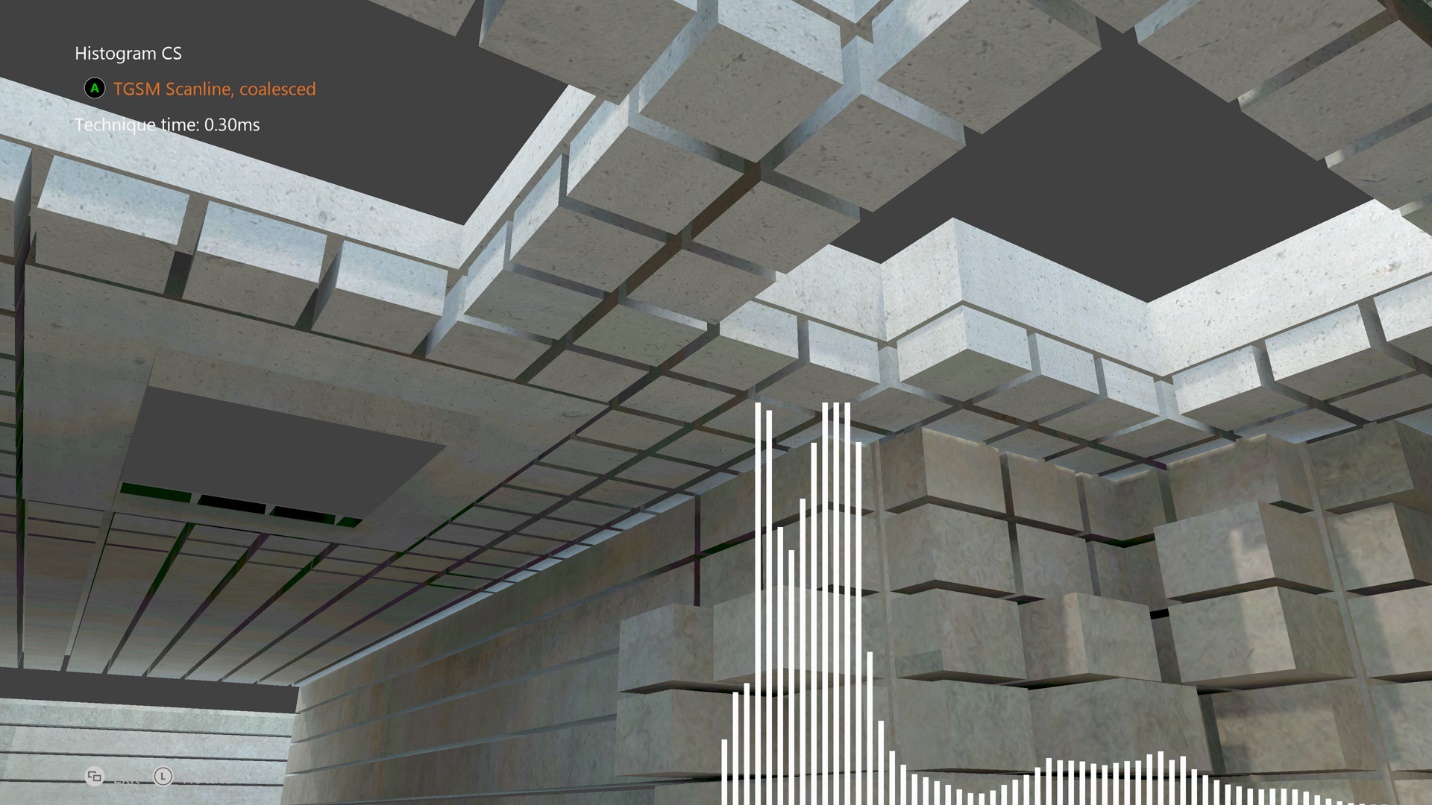


ヒストグラム CS サンプル

*このサンプルは Microsoft Game Development Kit (2020 年 8 月) と互換性があります*

# 説明

# このサンプルは、 Xbox でのコンピュート シェーダーのパフォーマンスに関する考慮事項をいくつか示します。実際には、シーンの強度の 64 バケット ヒストグラムの演算を、(最適な実装によって) 0.2 ミリ秒未満で行うための最適な方法を示します。



# サンプルのビルド

Xbox One 開発キットを使用している場合、アクティブ ソリューション プラットフォームを Gaming.Xbox.XboxOne.x64 に設定します。

Xbox Series X|S 開発キットを使用している場合、アクティブ ソリューション プラットフォームを Gaming.Xbox.Scarlett.x64 に設定します。

*詳細については、GDK のドキュメントの*「サンプルの実行」*を参照してください。*

# サンプルの使用方法

|  |  |
| --- | --- |
| 操作 | ゲームパッド |
| ヒストグラムの手法を循環させる | A ボタン |
| ビューの回転 | 左サムスティック |
| ビューのリセット | 左サムスティック (クリック) |
| 終了 | ビュー ボタン |

# 実装上の注意

# CPU プログラミングでは、作業を複数のコアに分散する場合、多くの場合、各コアがデータの連続ブロックを取得するようにして、キャッシュの使用率ひいてはパフォーマンスを向上させようとします。GPU でアルゴリズムを実装する場合も同じように考えることは容易です。HLSL では、各シェーダー プログラムは独自の領域で動作するように見えるからです (CPU 上のスレッドに似ています)。ただし、最新の GPU は高度に並列化されているため、これは事実とはまったく異なります。

# Xbox One GPU は、実際には、ウェーブと呼ばれるスレッドのグループ上で動作します (最大 64 個)。つまり、各スレッドは、並列スレッドのより大きなグループの一部であり、ロックステップで実行され、すべてがまったく同じ命令を実行します。これは、CPU 上の 64 長の浮動小数点数 SIMD 命令セットと似ていると考えることができます。

# 要するに、各スレッドに隣接データをプルさせると最終的に最適でなくなります。理想的なのは、読み取りを結合して、隣接スレッドが隣接データを読み取るようにすることです。効果的に、スレッド グループ内の各スレッドはループの各反復で 64 おきにアイテムを読み取ります。サンプルからわかるように、これにより、最終的にはるかに最適になります。

# ただし、読み取り順序はこのサンプルでの唯一の阻害要素ではありません。ヒストグラムは、ヒストグラムの各バケットへのアトミック追加を実行することによって生成されます (合計 64 バケットあります)。Xbox One GPU はこれらを非常に迅速に行います (実際には L2 キャッシュで実行されます) が、それでもやはり全体の進捗に影響する要素となります。これがさらに高速に実行されるようにするために、HLSL でグループ共有メモリーを通じて公開される、非常に高速な内蔵 LDS キャッシュを活用できます。各スレッドがスキャンライン全体を処理するようにし、結果を LDS で構築してから、最後に最終結果をアトミックに追加します。LDS でのアトミック操作はメイン メモリー (または L2) よりも大幅に高速であるため、これによってパフォーマンスが大きくスピードアップします。

# この方法の詳細およびその他については、Xfest 2013 のコンピューター シェーダーに関する講演を確認してください。

# 既知の問題

なし。

# 更新履歴

サンプルの元のバージョンは、XSF ベースのフレームワークを使用して記述されました。2020 年 6 月に ATG サンプル テンプレートを使用するように書き換えられました。

# プライバシー ステートメント

サンプルをコンパイルして実行する場合、サンプルの使用状況を追跡するために、サンプル実行ファイルのファイル名が Microsoft に送信されます。このデータ コレクションからオプトアウトするには、Main.cpp の「Sample Usage Telemetry」というラベルの付いたコードのブロックを削除します。

全般的な Microsoft のプライバシー ポリシーの詳細については、「[Microsoft プライバシー ステートメント](https://privacy.microsoft.com/en-us/privacystatement/)」を参照してください。