

Song song hóa và tối ưu hóa một ứng dụng

Trần Trung Kiên
ttkien@fit.hcmus.edu.vn

Cập nhật 30/12/2020



fit@hcmus

KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Qui trình song song hóa / tối ưu hóa

Ta nên ưu tiên song song hóa / tối ưu hóa phần nào trong chương trình?

- Đo thời gian của các phần để quyết định
- Khi tối ưu hóa (sau khi đã song song hóa), ta có thể nhanh chóng đo thời gian của các hoạt động liên quan đến GPU: `nvprof --print-gpu-trace ./a.out`

1. Phân tích

2. Thiết kế

3. Cài đặt

4. Đánh giá

Chương trình có chạy nhanh hơn?
Nếu không thì tại sao?

Song song hóa / tối ưu hóa như thế nào?

- Mỗi vòng lặp sẽ tạo ra một phiên bản mới dựa trên các phiên bản trước đó
- Nên đi từng bước một, từ tuần tự đến song song, từ song song đến song song được tối ưu hóa

Quy trình song song hóa / tối ưu hóa

Ta nên ưu tiên song song hóa / tối ưu hóa phần nào trong chương trình?

- Đo thời gian của các phần để quyết định
- Khi tối ưu hóa (sau khi đã song song hóa), ta có thể nhanh chóng đo thời gian của các hoạt động liên quan đến GPU: `nvprof --print-gpu-trace ./a.out`

1. Phân tích

2. Thiết kế

3. Cài đặt

4. Đánh giá

Chương trình có chạy nhanh hơn?
Nếu không thì tại sao?

Song song hóa / tối ưu hóa như thế nào?

Làm sao để thực hiện quy trình này một cách tốt nhất có thể?

Một số lời khuyên:

- **Giữ tâm tĩnh**
- Giữ code ngăn nắp, dễ đọc
- Code nhanh hay chậm?
- Không nên ngồi code quá lâu mà nên có các khoảng nghỉ ngắn
- Dùng một editor tốt và học cách sử dụng cho hiệu quả

Định hướng tối ưu hóa nói chung

- Đưa đủ các công việc độc lập để tận dụng các tài nguyên phần cứng của GPU
 - ▣ Đưa đủ số lượng block để tận dụng các SM (nếu số lượng block của một hàm kernel không đủ nhiều thì có thể xem xét dùng CUDA stream để “nhét” thêm các block đến từ các hàm kernel khác)
 - ▣ Trong mỗi SM, đưa đủ số lượng câu lệnh độc lập (có thể đến từ cùng một warp, có thể đến từ các warp khác nhau) để tận dụng các execution pipeline, che độ trễ (một cách đơn giản là thay đổi block size)
- Truy xuất DRAM hiệu quả
 - ▣ Cố gắng không để các thread trong cùng một warp truy xuất đến các phần tử nằm “rải rác” trong DRAM
 - ▣ Dùng SMEM để giảm các truy xuất DRAM, cũng như để truy xuất DRAM hiệu quả
- Giảm thiểu phân kỳ warp

Đồ án cuối kỳ

Song song hóa và tối ưu hóa một ứng dụng
(nhóm bạn có thể tự chọn ứng dụng; nếu
không thì nhóm bạn có thể chọn ứng dụng
mà Thầy đề xuất)

Đồ án cuối kỳ - Nội dung cần trình bày trong file Colab

1. Mô tả ứng dụng

- ❑ Ứng dụng mà nhóm bạn chọn là gì?
 - Input? Output?
 - Ý nghĩa của ứng dụng trong thực tế?
- ❑ Ứng dụng này có cần phải tăng tốc không?

Đồ án cuối kỳ - Nội dung cần trình bày trong file Colab

2. Cài đặt tuần tự

- ❑ Thiết kế: Mô tả các bước để đi từ input đến output (mô tả một cách đơn giản và rõ ràng, đừng show code)
- ❑ Đánh giá:
 - Mô tả kịch bản thí nghiệm
 - Chạy code để xem kết quả
 - Có đúng không?

Đồ án cuối kỳ - Nội dung cần trình bày trong file Colab

3. Cài đặt song song

- ❑ Phân tích: Nhóm bạn song song hóa những bước nào và tại sao lại là những bước này?
- ❑ Thiết kế: Cách nhóm bạn song song hóa? (mô tả một cách đơn giản và rõ ràng, đừng show code)
- ❑ Đánh giá:
 - Mô tả kịch bản thí nghiệm
 - Chạy code để xem kết quả
 - Có đúng & *nhANH hơn* không? Nếu không thì nhóm bạn có biết tại sao?

Đồ án cuối kỳ - Nội dung cần trình bày trong file Colab

4. Cài đặt song song + tối ưu hóa

Nhóm bạn nên có ≥ 2 phiên bản được tối ưu hóa

Với mỗi phiên bản:

- ❑ Phân tích: Nhóm bạn tối ưu hóa những phần nào (thường là: những hàm kernel nào)? Tại sao lại là những phần này?
- ❑ Thiết kế: Cách nhóm bạn tối ưu hóa? (mô tả một cách đơn giản và rõ ràng, đừng show code)
- ❑ Đánh giá:
 - Mô tả kịch bản thí nghiệm
 - Chạy code để xem kết quả
 - Có đúng & *nhANH hơn* không? Nếu không thì nhóm bạn có biết tại sao?

Đồ án cuối kỳ - Nội dung cần trình bày trong file Colab

5. Nhìn lại quá trình làm đồ án

- ❑ Mỗi thành viên: Đã gặp những khó khăn gì?
(Hay mọi chuyện đều thuận lợi)
- ❑ Mỗi thành viên: Có học được gì hữu ích?
(Hay không học được gì)
- ❑ Nhóm: Nếu có thêm thời gian thì sẽ làm gì?

Đồ án cuối kỳ - Nội dung cần trình bày trong file Colab

6. Tài liệu tham khảo

Để hoàn thành đồ án này, nhóm bạn đã tham khảo những tài liệu nào?

Đồ án cuối kỳ - Các file code

Mỗi phiên bản (phiên bản tuần tự, phiên bản song song, phiên bản song song được tối ưu hóa lần 1, phiên bản song song được tối ưu hóa lần 2, ...) nên được tách ra trong một file code

Đồ án cuối kỳ - Làm việc nhóm

Tất cả thành viên trong nhóm đều phải hiểu về đồ án của nhóm mình (tất nhiên là phải hiểu cả code)

Khi làm việc nhóm, nên có một file kế hoạch về các công việc cần làm, lượng thời gian cho mỗi công việc, phân công ai làm công việc nào (file này có thể sẽ cần điều chỉnh nhiều lần trong suốt quá trình làm việc nhóm)

Đồ án cuối kỳ - Nộp bài & vấn đáp

- ☐ **Trước 23h55 22/01:** mỗi nhóm cử một bạn đại diện upload đồ án lên moodle, gồm:
 - ☐ File làm việc nhóm
 - ☐ File Colab
 - ☐ Các file code
- ☐ **23-24/01:** trình bày và vấn đáp qua zoom (cần bật webcam)

Sẽ có 15 phút trình bày (mỗi bạn sẽ trình bày một nửa, ai trình bày nửa nào là do mình chỉ định) + 5 phút hỏi đáp
- ☐ Khoảng ngày 26/1 sẽ có điểm trên moodle và mình sẽ để khoảng 2 ngày để bạn phúc khảo

Ứng dụng mà Thầy đề xuất để song song hóa và tối ưu hóa:

Thay đổi kích thước ảnh bằng phương pháp “seam carving”

1. Mô tả ứng dụng

- ☐ **Input:** một tấm ảnh (đề án này: ảnh RGB)
- ☐ **Output:** tấm ảnh được thay đổi kích thước mà **không làm biến dạng các đối tượng quan trọng** (đề án này: tấm ảnh được thu hẹp chiều rộng lại)
- ☐ Ý nghĩa thực tế của ứng dụng: một use case là ta có thể muốn có các phiên bản với các kích thước khác nhau của cùng một tấm ảnh để hiển thị trên các thiết bị khác nhau (máy tính, điện thoại, ...)
- ☐ Ứng dụng này có cần tăng tốc không?
Cần, ứng dụng này chạy chậm khi cài đặt tuần tự



Ảnh ban đầu



Scaling



Cropping



Seam carving

2. Cài đặt tuần tự (ý tưởng)

Một seam: một tập các pixel, mỗi dòng một pixel, pixel của dòng r & dòng $r+1$ được kết nối với nhau



Tìm **seam** ít quan trọng nhất

Xóa seam này

Tìm seam ít quan trọng



...



Lặp lại qui trình này cho đến khi đạt được chiều rộng mong muốn

Demo ...

Tìm seam ít quan trọng nhất

Tìm độ quan trọng của mỗi pixel
(Một cách: dùng edge detection)

Tìm seam ít quan trọng nhất
từ độ quan trọng của các pixel



Tìm độ quan trọng của mỗi pixel

Một cách: dùng edge detection

- ❑ Chuyển ảnh RGB sang ảnh grayscale
- ❑ Phát hiện cạnh theo chiều x (1): thực hiện convolution giữa ảnh grayscale với bộ lọc x-Sobel

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

- ❑ Phát hiện cạnh theo chiều y (2): thực hiện convolution giữa ảnh grayscale với bộ lọc y-Sobel

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

- ❑ Độ quan trọng của một pixel = $|\text{kết quả tương ứng của (1)}| + |\text{kết quả tương ứng của (2)}|$

Tìm seam ít quan trọng nhất từ độ quan trọng của các pixel

Tính độ quan trọng của tất cả các seam có thể có?

$O(\text{width} \times 3^{\text{height}})$ ☹️

1	4	3	5	2
3	2	5	2	3
5	3	4	2	1

Độ quan trọng của pixel

Tìm seam ít quan trọng nhất từ độ quan trọng của các pixel

Cách $O(\text{width} \times \text{height})$
để tìm seam ít quan trọng nhất

1	4	3	5	2
3	2	5	2	3
5	3	4	2	1

Độ quan trọng của pixel

Độ-quan-trọng-ít-nhất tính tới dưới cùng

Tìm seam ít quan trọng nhất từ độ quan trọng của các pixel

Cách $O(\text{width} \times \text{height})$
để tìm seam ít quan trọng nhất

1	4	3	5	2
3	2	5	2	3
5	3	4	2	1

Độ quan trọng của pixel

5	3	4	2	1

Độ-quan-trọng-ít-nhất tính tới dưới cùng

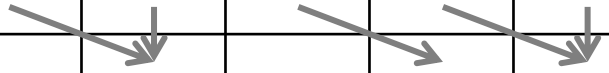
Tìm seam ít quan trọng nhất từ độ quan trọng của các pixel

Cách $O(\text{width} \times \text{height})$
để tìm seam ít quan trọng nhất

1	4	3	5	2
3	2	5	2	3
5	3	4	2	1

Độ quan trọng của pixel

6	5	7	3	4
5	3	4	2	1



Độ-quan-trọng-ít-nhất tính tới dưới cùng

Tìm seam ít quan trọng nhất từ độ quan trọng của các pixel

Cách $O(\text{width} \times \text{height})$
để tìm seam ít quan trọng nhất

1	4	3	5	2
3	2	5	2	3
5	3	4	2	1

Độ quan trọng của pixel

6	9	6	8	5
6	5	7	3	4
5	3	4	2	1

Độ-quan-trọng-ít-nhất tính tới dưới cùng

Tài liệu tham khảo về “seam carving”

(Thuật ngữ: “energy” = độ quan trọng)

- ☐ Shai Avidan & Ariel Shamir, Seam Carving for Content-Aware Image Resizing (bài báo gốc)
- ☐ 18.S191 MIT Fall 2020, Seam Carving: [part1](#), [part2](#)
- ☐ [Wikipedia, Seam Carving](#)

Nói thêm ...

- ☐ Làm sao để hiển thị file ảnh pnm trong Colab?
- ☐ Làm sao để hiển thị kết quả edge detection trong Colab?

Thank you

