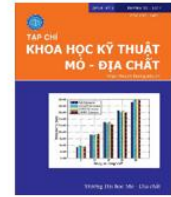




## Tạp chí Khoa học Kỹ thuật Mỏ - Địa chất

Trang điện tử: <http://tapchi.humg.edu.vn>



# Nghiên cứu giải pháp xử lý ảnh UAV dựa trên công nghệ CUDA hỗ trợ tìm kiếm và cứu hộ cứu nạn

Nguyễn Trường Xuân<sup>1,\*</sup>, Nguyễn Thị Mai Dung<sup>1</sup>, Trần Thị Hải Vân<sup>1</sup>, Đào Khánh Hoài<sup>2</sup>, Vũ Văn Trường<sup>2</sup>, Nguyễn Tuấn Anh<sup>3</sup>

<sup>1</sup> Khoa Công nghệ Thông tin, Trường Đại học Mỏ - Địa chất, Việt Nam

<sup>2</sup> Đại học Lê Quý Đôn, Cầu Giấy, Hà Nội, Việt Nam

<sup>3</sup> Cục Bản đồ Quân đội, Bộ Tổng Tham mưu, Việt Nam

### THÔNG TIN BÀI BÁO

Quá trình:

Nhận bài 14/7/2017

Chấp nhận 25/7/2017

Đăng online 30/10/2017

Từ khóa:

GPU

CUDA

UAV

Cứu hộ cứu nạn

### TÓM TẮT

Trong những năm gần đây UAV đã đóng một vai trò quan trọng thu thập thông tin địa lý kịp thời và chính xác, cung cấp cơ sở quan trọng cho việc đánh giá thảm họa, cứu trợ khẩn cấp và tái thiết. Tuy nhiên với đặc trưng của tư liệu ảnh UAV là có độ phân giải cao, cần sử dụng nhiều tấm ảnh trong phạm vi khảo sát, do đó việc xử lý đồng thời các tấm ảnh với dung lượng dữ liệu lớn trên CPU truyền thống không thể đáp ứng được yêu cầu của các tình huống khẩn cấp. Bài báo nghiên cứu giải pháp xử lý ảnh UAV dựa trên công nghệ CUDA (Compute Unified Device Architecture), tận dụng lợi thế tính toán song song mạnh mẽ của GPU, giúp cải thiện đáng kể thời gian xử lý dữ liệu. Kết quả bài báo khẳng định khả năng ứng dụng tư liệu ảnh UAV trong ứng phó khẩn cấp tại những khu vực xảy ra thiên tai.

© 2017 Trường Đại học Mỏ - Địa chất. Tất cả các quyền được bảo đảm.

## 1. Mở đầu

Trong những năm gần đây UAV (Unmanned Aerial Vehicle - Thiết bị bay không người lái) đã đóng một vai trò quan trọng thu thập thông tin địa lý kịp thời và chính xác, cung cấp cơ sở quan trọng cho việc đánh giá thảm họa, cứu trợ khẩn cấp và tái thiết lập.

Với đặc trưng của tư liệu ảnh UAV là có độ phân giải cao, để phủ trùm khu vực khảo sát cần có số lượng ảnh chụp lớn, do đó việc xử lý đồng thời các tấm ảnh với dung lượng dữ liệu lớn trên

CPU (Central Processing Unit - Bộ xử lý trung tâm) truyền thống không thể đáp ứng được yêu cầu của các tình huống khẩn cấp.

Chính vì vậy một xu hướng nghiên cứu mới đã ra đời, xu hướng phát triển các thuật toán song song thực hiện trên GPU (Graphic Processing Unit - Bộ xử lý đồ họa). Khi so sánh với các thuật toán của CPU trong tính toán song song, tính toán phân tán sức mạnh tính toán của GPU đã chứng tỏ là nhanh hơn hàng chục đến một trăm lần (Manavski, 2007). Từ năm 2007, công nghệ CUDA (Compute Unified Device Architecture - Kiến trúc thiết bị tính toán hợp nhất) được phát triển bởi hãng NVIDIA ra đời đã cho phép thực hiện các tính toán song song với

\*Tác giả liên hệ

E-mail: [nguyentruongxuan@humg.edu.vn](mailto:nguyentruongxuan@humg.edu.vn)

các phép tính phức tạp đạt hiệu suất cả ngàn lệnh trong một thời điểm. Với CUDA các lập trình viên có thể nhanh chóng phát triển các ứng dụng tính toán song song cho rất nhiều ứng dụng khác nhau như: Điện toán, sắp xếp, tìm kiếm, xử lý tín hiệu số, ảnh,... Việc nghiên cứu áp dụng CUDA để tăng tốc độ tính toán cho các bài toán có khối dữ liệu đầu vào khổng lồ hoặc các bài toán yêu cầu tính thời gian thực đã thực sự trở thành một vấn đề cấp thiết trong thực tế.

Các nhà nghiên cứu đã triển khai tính toán CUDA cho các bài toán giải mã hình ảnh và video, chuyển đổi video HD, áp dụng trong Toán học, Xử lý ảnh, Sinh học và Hóa học, xử lý hình ảnh, sinh học tính toán và hóa học, phân tích địa chấn, ... (Sudipta, 2006).

Bài báo nghiên cứu giải pháp xử lý ảnh UAV dựa trên công nghệ CUDA, tận dụng lợi thế tính toán song song mạnh mẽ của GPU, giúp cải thiện đáng kể thời gian xử lý dữ liệu. Kết quả bài báo cho thấy thuật toán xử lý ảnh UAV dựa trên CUDA cải thiện đáng kể hiệu suất của công tác xử lý ảnh và đáp ứng đầy đủ các yêu cầu xử lý dữ liệu khẩn cấp khi xảy ra thảm họa thiên tai.

## 2. Tính toán song song trên GPU trong CUDA

CUDA là kiến trúc mới bao gồm cả phần

cứng và phần mềm để phát triển và quản lý việc tính toán trên GPU như một thiết bị tính toán song song mà không cần ánh xạ vào các hàm lập trình đồ họa. Cơ chế đa nhiệm của hệ điều hành chịu trách nhiệm cho việc quản lý truy cập tới GPU bởi các ứng dụng CUDA và ứng dụng đồ họa chạy song song.

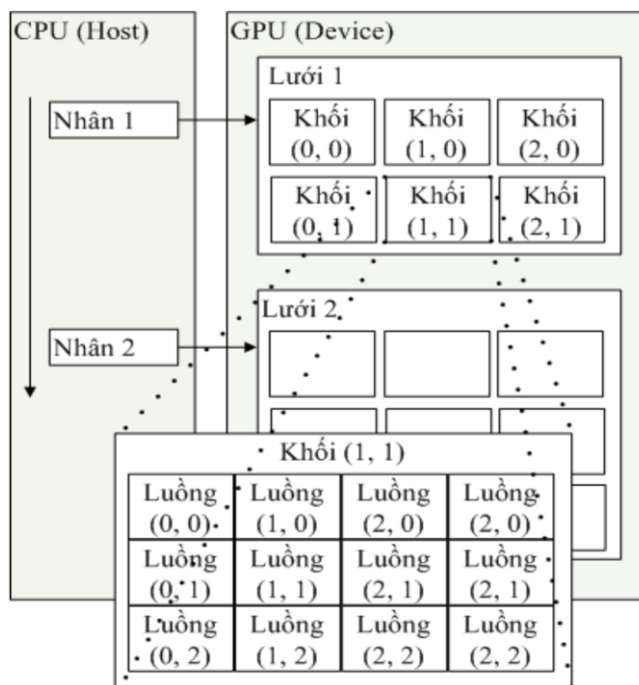
GPU là một công cụ xử lý đồ họa mạnh và là một bộ xử lý hỗ trợ lập trình song song ở mức cao, giúp giải các bài toán số học cần khả năng xử lý số học phức tạp và băng thông bộ nhớ tăng hơn đáng kể so với CPU cùng loại.

Cấu trúc của một chương trình CUDA thường sử dụng hai hàm: Một hàm dành cho việc truy cập dữ liệu và hàm còn lại gọi là hàm kernel dùng cho việc xử lý dữ liệu. Quy trình hoạt động truyền dữ liệu giữa Host và Device được mô tả như sau (Nguyễn Đức Minh, 2016):

Dữ liệu cần tính toán luôn ở trên bộ nhớ của Host, vì vậy trước khi muốn thực hiện trên Device bước đầu tiên là sao chép dữ liệu cần tính toán từ bộ nhớ Host sang bộ nhớ Device.

- Tiếp theo Device sẽ thực hiện việc tính toán trên dữ liệu đó dựa trên các hàm riêng của Device để tính toán.

- Kết thúc quá trình tính toán, dữ liệu được sao chép lại từ bộ nhớ Device sang bộ nhớ Host.



Hình 1. Kiến trúc đa luồng trong mô hình lập trình song song CUDA.

- Trong đó Host được định nghĩa là những tác vụ và cấu trúc phần cứng, phần mềm được xử lý từ CPU. Device là những tác vụ và cấu trúc phần cứng, phần mềm được xử lý từ GPU.

CUDA cho phép đa nhân chạy đồng thời trên một GPU đơn. CUDA coi mỗi nhân như một lưới (grid). Một lưới là một tập các khối (block). Mỗi khối chạy cùng nhân nhưng độc lập với nhau. Trong mỗi khối lại chứa nhiều luồng (thread). Luồng là đơn vị nhỏ nhất trên GPU. Các ứng dụng CUDA được tạo sử dụng ngôn ngữ lập trình C/C++.

Trong lập trình CUDA, GPU được xem như là một thiết bị tính toán có khả năng thực hiện một số lượng rất lớn các luồng song song. GPU hoạt động như là một bộ đồng xử lý với CPU chính. Hình 1 mô tả kiến trúc đa luồng trong mô hình lập trình song song CUDA.

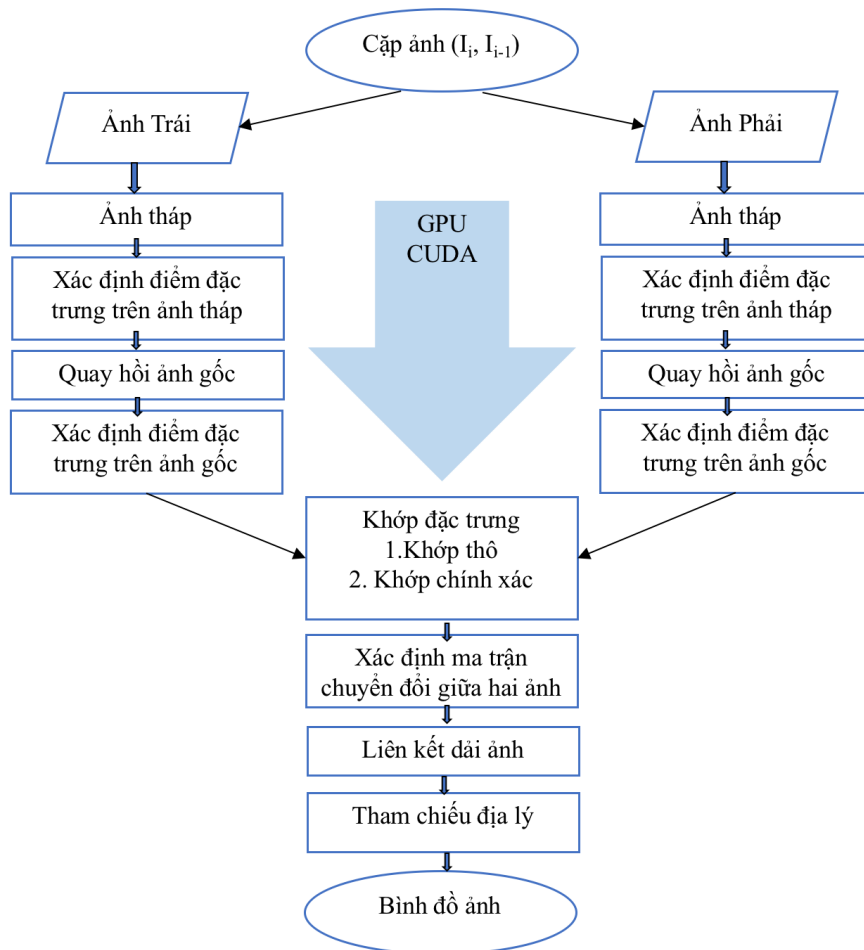
Để chương trình CUDA hoạt động được trong môi trường windows hoặc linux, cần phải có các thư viện hỗ trợ. Các thư viện này do

NVIDIA cung cấp bao gồm các phần: Trình điều khiển thiết bị đồ họa cho GPU của NVIDIA, bộ công cụ phát triển CUDA (CUDA Toolkit) và bộ CUDA SDK.

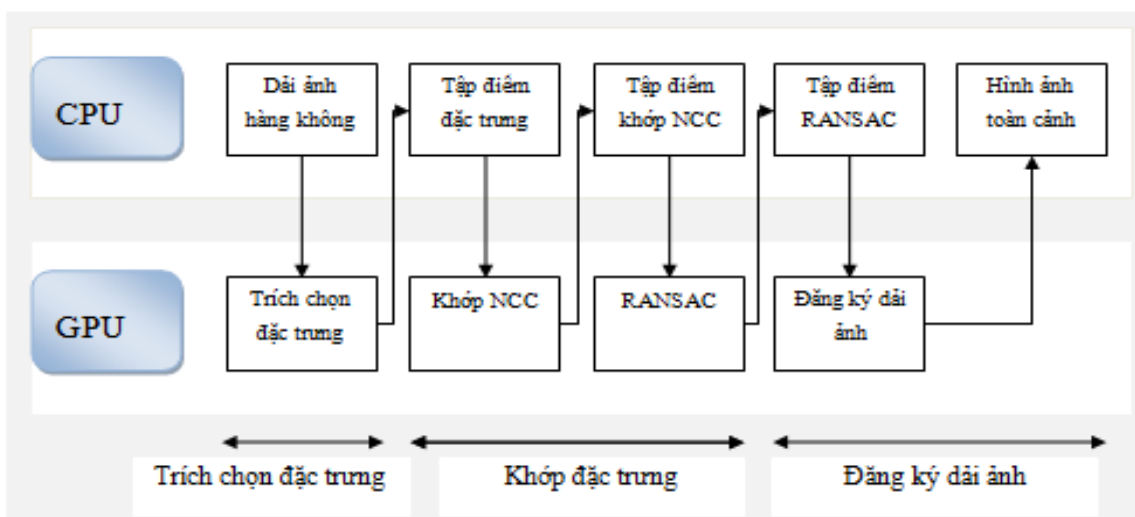
### 3. Giải pháp xử lý nhanh ảnh UAV dựa trên công

Mô hình thuật toán xử lý nhanh ảnh UAV thành lập bản đồ ảnh phục vụ công tác cứu hộ cứu nạn gồm các bước như mô tả ở Hình 2, bao gồm ba bước chính: Trích chọn đặc trưng, Khớp đặc trưng và Tham chiếu địa lý.

Bước trích chọn điểm đặc trưng được thực hiện trên ảnh tháp trung gian và ảnh gốc. Kỹ thuật tháp ảnh được sử dụng trong bước này nhằm giảm không gian tính toán. Không gian tìm kiếm điểm đặc trưng trên ảnh gốc được thu nhỏ lại bằng cách tạo ra các ảnh tháp có cấu trúc tương tự



Hình 2. Mô hình thuật toán xử lý nhanh ảnh UAV.



Hình 3. Phân chia tính toán trên CPU và GPU.

ảnh gốc nhưng kích thước đã bị thu nhỏ giữa các bậc tháp liền kề nhau (Fei et al., 2013).

Thuật toán khớp theo đặc trưng có lồng ghép bước khớp sơ bộ theo vùng giá trị mức xám để giảm thời gian và tài nguyên tính toán đồng thời nâng cao tính ổn định của toàn bộ quy trình khớp.

Bước khớp thô sử dụng kết hợp phương pháp khớp ảnh theo vùng giá trị độ xám và khớp ảnh theo điểm đặc trưng để tìm các cặp điểm ảnh cùng tên trên hai ảnh. Phương pháp đối sánh tương quan chéo chuẩn hóa (NCC - Normalized cross correlation) được áp dụng để so sánh độ khớp nhau giữa mẫu ảnh chọn trên ảnh trái với mẫu ảnh ứng viên trong cửa sổ tìm kiếm trên ảnh phải (Ackermann, 1984).

Bước khớp chính xác, thuật toán RANSAC (Random Sample Consensus - Phương pháp lấy mẫu đồng thuận ngẫu nhiên) (Fischler et al., 1981) được sử dụng để loại bỏ các cặp điểm khớp sai từ bước khớp thô. Từ kết quả của bước khớp thô thu được một danh sách khớp sơ bộ N cặp đặc trưng của hai ảnh. Do cặp ảnh hàng không chụp ở khoảng cách xa nên có thể dùng mô hình xạ ảnh để liên kết hai ảnh (Goshtasby, 2005). Bước khớp chính xác gồm các bước sau (Đào Khánh Hoài, 2014):

- Chọn ngẫu nhiên 4 cặp điểm trên hai tập đặc trưng để xác định các tham số mô hình xạ ảnh.
- Giải các tham số mô hình xạ ảnh. Ảnh xạ tập đặc trưng ảnh trái sang ảnh phải theo mô hình

vừa giải. Sử dụng khoảng cách Hausdorff (Goshtasby, 2005). để lọc ra các cặp đặc trưng thỏa mãn và thêm vào danh sách P. Các cặp đặc trưng được coi là thỏa mãn mô hình xạ ảnh khi khoảng cách Hausdorff từ các đặc trưng ảnh xạ sang ảnh phải nhỏ hơn một ngưỡng cho trước. Để giải mô hình liên kết tương đối giữa hai ảnh thường áp dụng thuật toán tám điểm (Hartley, 1997). Vì vậy trong bước tiếp theo của thuật toán sử dụng tiêu chí  $P > 8$ .

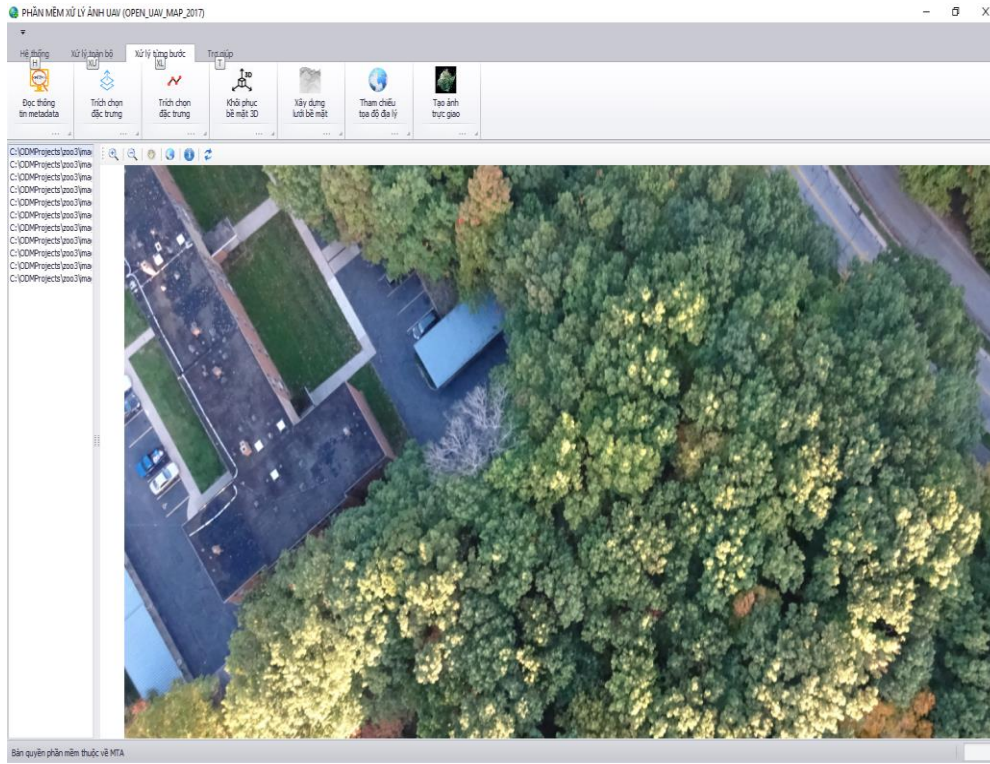
- Nếu số cặp đặc trưng trong P nhỏ hơn 8 thuật toán chạy lại các bước trên. Ngược lại 8 cặp đặc trưng có độ khớp tốt nhất được lọc ra để tính toán mô hình tương đối giữa hai ảnh.

Bước Tham chiếu địa lý, ảnh kết quả của bước khớp ảnh sẽ được chuyển đổi về khung tham chiếu đã được định nghĩa trước để ghép thành một bình đồ ảnh.

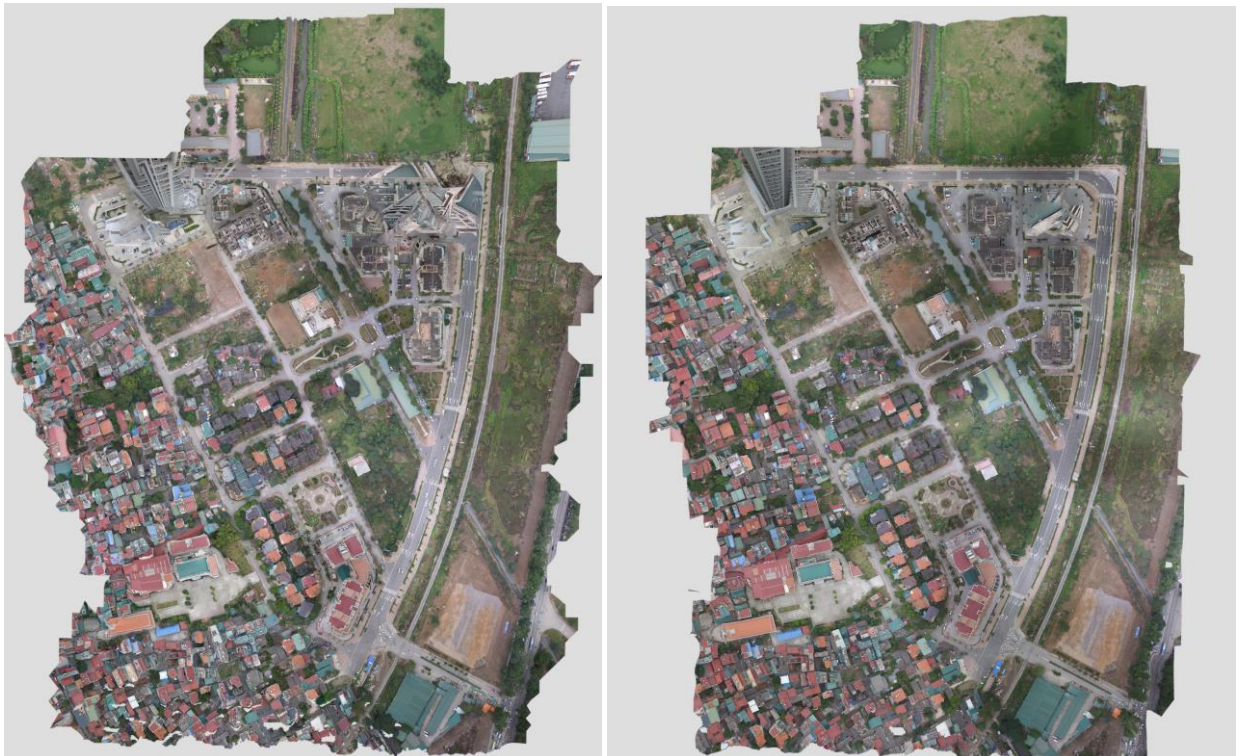
Trong quy trình thuật toán, nền tảng tính toán GPU được áp dụng để tăng tốc độ thực thi. Việc phân chia các bước tính toán trên CPU và GPU như được thể hiện qua Hình 3. Các bước xử lý chính và chiếm nhiều thời gian như Trích chọn đặc trưng, Khớp thô, Khớp chính xác và Đăng ký dải ảnh được thực hiện trên GPU.

Quá trình tính toán trên GPU được thực hiện như sau: Với mỗi ảnh có kích thước Width\*Hight sẽ được chia ra  $(width/16)*(height/16)$  khối. Với mỗi khối ảnh sẽ có 256 luồng xử lý đồng thời. Việc phân chia này nhằm đảm bảo các tính toán độc lập trên mỗi điểm ảnh sẽ luôn có một luồng





Hình 4. Giao diện phần mềm xử lý ảnh UAV.



(a)

(b)

Hình 5. Ảnh trực giao khu vực nghiên cứu tạo bởi (a) phần mềm thiết kế, (b) phần mềm Pix4D.

xử lý. Qua đó toàn khối ảnh sẽ được tính toán đồng thời. Sau khi phân chia luồng, mỗi điểm ảnh được tính toán như sau:

```
int i = blockIdx.x * blockDim.x + threadIdx.x;
int j = blockIdx.y * blockDim.y + threadIdx.y;
if (i < width && j < height)
{
// Xử lý với điểm ảnh (i,j)
}
```

#### 4. Thực nghiệm

Với mục tiêu nghiên cứu giải pháp xử lý nhanh ảnh UAV dựa trên công nghệ CUDA, tận dụng lợi thế tính toán song song mạnh mẽ của GPU, nhóm nghiên cứu đã tiến hành thực nghiệm trên dải ảnh UAV chụp khu đô thị Resco.

Mô hình thuật toán được thực hiện trên nền tảng phần cứng:

CPU: Intel (R) Xeon (R) CPU W3550@3.07GHz 4 lõi 8 luồng.

GPU: NVIDIA Quadro 4000.

Dữ liệu thực nghiệm là ảnh hàng không được chụp từ máy bay không người lái UAV D3Plantom. Kích thước ảnh 4000 × 3000 điểm ảnh, độ phân giải điểm ảnh 0.2 mét. Định dạng ảnh JPG. Độ chính xác yêu cầu trong bước khớp chính xác nhỏ hơn 3 điểm ảnh.

Nhóm nghiên cứu đã tiến hành xây dựng phần mềm xử lý ảnh UAV theo mô hình thuật

toán dựa trên nền tảng công nghệ CUDA. Hình 4 biểu diễn phần mềm xử lý ảnh UAV đã được phát triển.

Kết quả phần mềm xử lý ảnh UAV áp dụng lập trình tính toán song song trên CUDA là ảnh trực giao khu vực nghiên cứu (Hình 5a). Để khẳng định độ tin cậy của thuật toán đề xuất, ảnh trực giao kết quả được so sánh với sản phẩm của phần mềm xử lý ảnh UAV thương mại Pix4D (Hình 5b). Thời gian xử lý 150 cảnh ảnh UAV khu vực nghiên cứu trên phần mềm thiết kế là 3 giờ 30 phút, trên phần mềm Pix4D là 8 giờ.

Các điểm khống chế được thu thập ngoài thực địa, sau đó được đưa vào phần mềm xử lý ảnh UAV để so sánh sự sai lệch về vị trí điểm. Bảng 1 biểu diễn sai lệch về vị trí điểm của 15 điểm khống chế trên ảnh trực giao kết quả và ảnh từ phần mềm Pix4D. Kết quả cho thấy phần mềm Pix4D tuy cho độ chính xác cao hơn nhưng tốc độ xử lý chậm. Phần mềm xử lý ảnh UAV dựa trên công nghệ CUDA hoàn toàn đáp ứng yêu cầu về độ chính xác vị trí và tốc độ xử lý nhanh của bài toán hỗ trợ tìm kiếm và cứu hộ cứu nạn.

#### 5. Kết luận

Khả năng cung cấp thông tin nhanh chóng và chính xác là hai yêu cầu cơ bản cho công tác quản lý thiên tai. Bài báo đề xuất một mô hình thuật toán xử lý ảnh UAV dựa trên nền tảng tính toán



Hình 6. Vị trí điểm khống chế trên ảnh và trên thực địa

Bảng 1. Bảng so sánh sai số vị trí điểm giữa phần mềm thiết kế, phần mềm Pix4D và thực nghiệm.

STT	GCP	Thực địa		Phần mềm thiết kế			Phần mềm Pix4D		
		X	Y	X	Y	Sai lệch (m)	X	Y	Sai lệch (m)
1	31	581413.01	2329683.52	581414.93	2329686.64	3.665	581412.93	2329683.43	0.118
2	32	581407.56	2329690.57	581409.42	2329694.05	3.948	581407.43	2329690.70	0.180
3	33	581388.43	2329601.84	581389.84	2329604.42	2.942	581388.28	2329601.87	0.154
4	34	581386.26	2329608.61	581387.37	2329611.28	2.893	581386.13	2329608.82	0.245
5	35	581367.72	2329527.87	581368.71	2329529.67	2.055	581367.73	2329527.92	0.052
6	36	581343.80	2329483.98	581344.25	2329485.39	1.481	581343.76	2329484.04	0.077
7	37	581334.80	2329473.82	581334.92	2329475.08	1.266	581334.77	2329473.96	0.145
8	38	581349.38	2329443.85	581349.77	2329445.11	1.320	581349.34	2329443.98	0.136
9	39	581307.00	2329422.67	581306.64	2329423.84	1.225	581306.97	2329422.72	0.058
10	40	581211.44	2329517.02	581210.30	2329519.09	2.364	581211.46	2329517.17	0.149
11	41	581166.44	2329625.66	581165.06	2329629.02	3.634	581166.47	2329625.71	0.068
12	42	581148.65	2329723.38	581147.48	2329727.71	4.478	581148.7	2329723.52	0.147
13	43	581070.98	2329824.74	581069.06	2329830.23	5.819	581070.94	2329824.84	0.118
14	44	581237.14	2329728.34	581236.84	2329732.57	4.243	581237.15	2329728.24	0.103
15	45	581318.57	2329740.54	581319.38	2329744.53	4.073	581318.61	2329740.54	0.048

song song GPU trong CUDA. Kết quả thực nghiệm cho thấy mô hình thuật toán là chính xác và đáp ứng đầy đủ các yêu cầu về thiết lập bản đồ ảnh trực giao.

Giải pháp đề xuất sử dụng phương pháp khớp tích hợp, kỹ thuật tháp ảnh, nền tảng tính toán CUDA do đó giảm thời gian tính toán, nhanh chóng thiết lập bản đồ ảnh khu vực xảy ra thiên tai.

Sự cải thiện về tốc độ xử lý có ý nghĩa rất quan trọng để nâng cao hiệu quả của việc xử lý tư liệu ảnh UAV tầm thấp, từ đó giúp khai thác những lợi thế của các hệ thống cảm biến UAV, cũng như nâng cao khả năng ứng phó với thiên tai và cứu hộ khẩn cấp.

### Tài liệu tham khảo

- Aaron Riley, 2016. Getting Started with CUDA. BookRix GmbH & Co. KG 81669 Munich.
- Ackermann, F., 1984. Digital Image Correlation: Performance and Potential Application in Photogrammetry, The
- Dao Khanh Hoai, Vu Van Truong, 2014. Building the automatically matching algorithm applicable for aerial image pairs. Journal of Science and Technology: *The Section on Information and Communication Technology (ICT)*, pp.21-33.
- Fei Yan, Weian Wang, Shijie Liu, Wen Chen, 2013. A Hierarchical image matching method for stereo satellite imagery, International

Archives of the Photogrammetry, *Remote Sensing and Spatial Information Sciences*, Vol XL-7/W1, 3rd ISPRS IWIDF 2013.

Fischler, M. A., R. C. Bolles, 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography, *Comm. of the ACM*, Vol 24, pp 381-39.

Goshtasby, A. A., 2005. 2-D and 3-D Image Registration: for Medical, *Remote Sensing, and Industrial Applications*, Willey Interscience, 284 p.

Hartley, V., 1997. In defense of the eight-point algorithm. *IEEE-T. PAMI*, 19(6):580-593.

Manavski S. A., 2007. CUDA compatible GPU as an efficient hardware accelerate for AES cryptography, *IEEE International Conference on Signal Processing and Communications*, 65-68.

Michael Kipper, Joshua Slavkin, Dmitry Denisenko, Implementing AES on GPU Final Report, University of Toronto, 2009.

Nguyễn Đức Minh, 2016. Nghiên cứu giải pháp công nghệ tính toán hiệu năng cao với bộ xử lý đồ họa GPU và ứng dụng. *Luận văn Thạc sĩ Công nghệ thông tin*, Trường Đại học Công nghệ, Đại học Quốc gia Hà Nội.

NVIDIA, CUDA C Programming Guide. Version 7.5, 2015.



Photogrammetric Record, 11(64), pp. 429-439.

Tracking and Matching. *Department of Computer Science*, 06-012.

Sudipta N Sinha, Jan-Michael Frahm and Marc Pollefeys, 2006. GPU-Based Video Feature

## **ABSTRACT**

### **Research on CUDA-based UAV image processing algorithm for search and rescue assistances**

Xuan Truong Nguyen<sup>1,\*</sup>, Dung Mai Thi Nguyen<sup>1</sup>, Van Hai Thi Tran<sup>1</sup>, Hoai Khanh Dao<sup>2</sup>,  
Truong Van Vu<sup>2</sup>, Anh Tuan Nguyen<sup>3</sup>

<sup>1</sup> *Hanoi University of Mining and Geology, Vietnam*

<sup>2</sup> *Le Quy Don Technical Unviversity, Vietnam*

<sup>3</sup> *Defense Mapping Agency of Vietnam, Vietnam*

In recent years, Unmanned Aerial Vehicle (UAV) systems have played an important role in obtaining timely and accurate acquisition of geospatial information, providing an important basis for disaster assessment, emergency relief, and reconstruction. However, with the high-resolution characteristics of UAV image, it is necessary to use number of images to cover the study areas. Therefore, the traditional CPU-based image processing method cannot meet the requirements of disaster emergencies. In this paper, we propose a Compute Unified Device Architecture (CUDA)-based UAV image processing algorithm, which takes advantage of the powerful parallel processing capability of the GPU, greatly improving the efficiency of processing time. Our experiments confirm the applicability of UAV image for natural disaster emergency response capability.

*Keywords:* CUDA, GPU, UAV, search and rescue assistances.