

# CHỒNG LỚP THÔNG TIN TRONG TÍNH TOÁN VÀ HIỂN THỊ ĐỊA HÌNH SỐ HÓA BA CHIỀU

Lê Anh<sup>1</sup>, Nguyễn Thị Lan<sup>1</sup>, Nguyễn Trung Kiên<sup>1</sup>

<sup>1</sup> Viện Công nghệ Mô phỏng, Học viện Kỹ thuật Quân sự  
lenhoanh@gmail.com, lanletech@gmail.com, kienmophong@gmail.com

**TÓM TẮT**— Địa hình số hóa ba chiều đóng vai trò quan trọng trong các ứng dụng mô phỏng quân sự. Hình ảnh trực quan về địa hình hỗ trợ cho người chỉ huy đánh giá đặc điểm và giá trị địa hình một cách chính xác, làm cơ sở cho việc lập phương án tác chiến. Dữ liệu địa hình ba chiều thông thường gồm tập các mặt tam giác và ảnh phủ bề mặt. Khi hiển thị, toàn bộ ảnh được phủ hết lên bề mặt dữ liệu địa hình (mỗi điểm ảnh được phủ lên một khu vực địa hình nhất định). Chính vì vậy, nếu quan sát ở cự ly gần một khu vực thì thực chất ta chỉ quan sát được một số thông tin điểm ảnh tương ứng phủ lên bề mặt khu vực đó. Để nâng cao chất lượng hiển thị hình ảnh bề mặt địa hình, đồng thời đảm bảo tốc độ hiển thị hình ảnh theo thời gian thực. Bài báo này trình bày cách tiếp cận, đưa ra mô hình phân lớp dữ liệu địa hình và chu trình xử lý chồng thêm các lớp ảnh phủ phù hợp lên bề mặt địa hình dựa trên nền tảng công nghệ lập trình GPU.

**Từ khóa**— Mô phỏng, địa hình số hóa ba chiều, ảnh phủ, lập trình GPU, texture, shader.

## I. ĐẶT VẤN ĐỀ

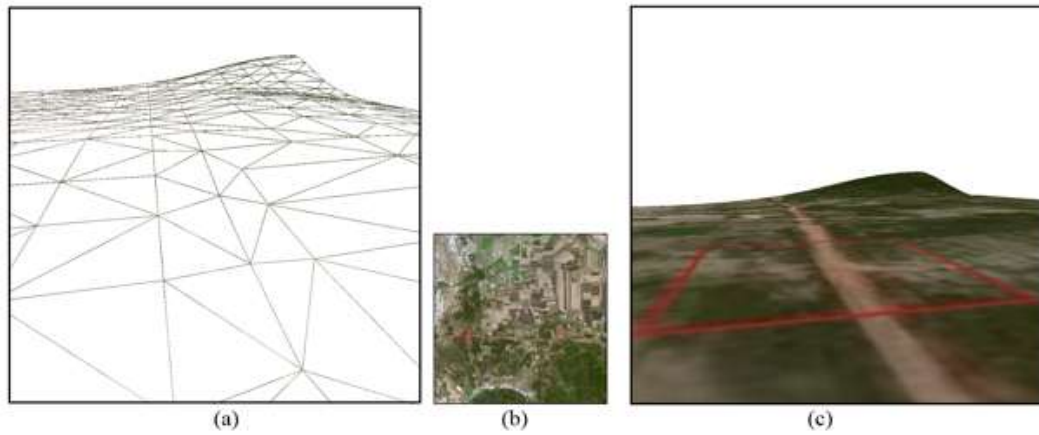
Các ứng dụng mô phỏng ngày càng thể hiện được vai trò thiết thực trong nhiều lĩnh vực của đời sống xã hội, đặc biệt là an ninh quốc phòng. Trong quân sự, người chỉ huy thường thảo luận, lập phương án tác chiến trên nền bản đồ địa hình được in sẵn. Để trực quan hóa khu vực tác chiến thì người chỉ huy thường sử dụng sa bàn đắp bằng cát hoặc chất liệu dễ tạo hình. Việc xây dựng sa bàn thường tốn nhiều thời gian, công sức, phải phá bỏ đi và đắp lại địa hình khi thay đổi khu vực tác chiến. Mô hình hóa địa hình ba chiều khu vực tác chiến là một giải pháp tối ưu nhằm trợ giúp người chỉ huy trong quá trình lập phương án tác chiến. Người chỉ huy có thể quan sát khu vực tác chiến ở nhiều vị trí, nhiều góc nhìn khác nhau để nắm được đặc điểm thuận lợi, khó khăn của địa hình từ đó đưa ra phương án tác chiến phù hợp.

Trong tác chiến hiện đại, môi trường tác chiến rất đa dạng và trải rộng trên một vùng không gian rộng lớn. Quá trình sử dụng các bản đồ in sẵn bộc lộ nhiều nhược điểm [3]. Phát triển các hệ thống mô hình hóa quang cảnh ba chiều của chiến trường là xu hướng của quân đội nhiều nước trên thế giới. Hệ thống mô phỏng chiến trường Dragon [1] của Jim và nhóm nghiên cứu đã được ứng dụng trong các bài huấn luyện quân sự của quân đội Mỹ. Hệ thống này đã thể hiện được hình ảnh ba chiều của địa hình và tương tác giữa các đối tượng quân sự. Phần mềm mô phỏng quang cảnh chiến trường Sandbox [3] được đề xuất bởi Baki và Erkan, dữ liệu dạng DEM được sử dụng để dựng lên địa hình khu vực tác chiến. Ưu điểm của các hệ thống trên là địa hình ba chiều được xây dựng trực quan, đặc biệt khi địa hình được quan sát toàn cảnh từ góc nhìn trên cao. Tuy nhiên, hệ thống chỉ sử dụng một ảnh bản đồ để trải lên toàn bộ bề mặt địa hình, nên chất lượng hình ảnh bị suy giảm khi quan sát chi tiết bề mặt địa hình.

Thực tế, bề mặt địa hình tồn tại nhiều lớp đối tượng như: đường xá, thảm thực vật, sông suối,... Do đó, để nâng cao chất lượng hình ảnh thì các lớp đối tượng trên bề mặt địa hình càng được hiển thị chi tiết càng tốt, nhưng vẫn phải đảm bảo tốc độ hiển thị thời gian thực. Đây là một vấn đề khó, bởi vì khi thông tin được thể hiện càng chi tiết thì độ phân giải của ảnh phủ trên địa hình yêu cầu càng phải cao, dẫn đến dung lượng ảnh càng lớn, đòi hỏi nhiều thời gian xử lý, tốc độ hiển thị hình ảnh sẽ giảm xuống. Bài báo đề xuất cách tiếp cận để nâng cao chất lượng hiển thị hình ảnh bề mặt địa hình, bằng cách phân lớp dữ liệu bề mặt và tổng hợp các lớp ảnh phủ tương ứng với mỗi lớp. Toàn bộ quá trình xử lý và tính toán được thực hiện trên bộ xử lý khả lập trình GPU của card đồ họa đảm bảo tốc độ hiển thị theo thời gian thực.

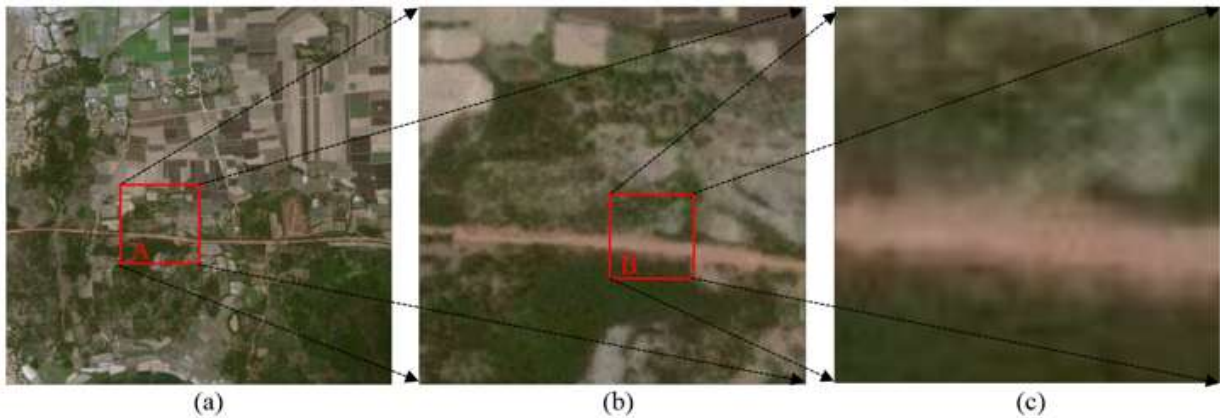
## II. PHỦ ẢNH ĐỊA HÌNH

Địa hình số hóa ba chiều là một yếu tố không thể thiếu và được sử dụng thường xuyên trong các ứng dụng mô phỏng. Dữ liệu mô hình hóa địa hình có thể là ảnh độ cao (height map), DTM (Digital Terrain Models), DEM (Digital Elevation), DSM (Digital Surface Models). Bề mặt địa hình thông thường được thể hiện qua tập hợp các mặt tam giác có kích thước khác nhau và không chồng lên nhau. Để tăng tính trực quan, bề mặt địa hình ba chiều thường được trải phủ bởi một ảnh vệ tinh. Tuy nhiên, khi địa hình có diện tích càng lớn thì mỗi điểm ảnh phải trải lên một vùng không gian rộng lớn hơn. Ở mô tả một vùng nhỏ trên ảnh vệ tinh (0b) phải trải lên một vùng diện tích lớn trên bề mặt địa hình (0c).



**Hình 1.** Phủ ảnh địa hình: (a) Địa hình số hóa ba chiều; (b) Ảnh vệ tinh; (c) Địa hình được phủ ảnh

Vấn đề gặp phải là khi bề mặt địa hình càng lớn thì chất lượng hình ảnh càng bị giảm, đặc biệt khi địa hình được quan sát ở cự ly càng gần (0). Nhận thấy, khi địa hình được nhìn ở khoảng cách xa ( $0a$ ) thì hình ảnh địa hình vẫn tương đối rõ nét, tuy nhiên khi địa hình được nhìn ở khoảng cách rất gần thì chất lượng hình ảnh bị suy giảm đáng kể,  $0c$  thể hiện hình ảnh một đoạn đường trên địa hình bị mờ và không còn rõ nét khi được quan sát gần. Nguyên nhân là một vùng nhỏ gồm số ít các điểm ảnh trong ảnh phủ phải trải lên một vùng rất rộng trên bề mặt địa hình, dẫn đến màu sắc ảnh phủ bị phân tán và mất dần đi sự chi tiết. Một giải pháp có thể đưa ra là sử dụng ảnh vệ tinh có độ phân giải lớn hơn, nhưng điều này dẫn đến dung lượng ảnh lớn, tốn nhiều không gian lưu trữ, khối lượng tính toán nhiều hơn, không đáp ứng được tính thời gian thực trong các ứng dụng mô phỏng.

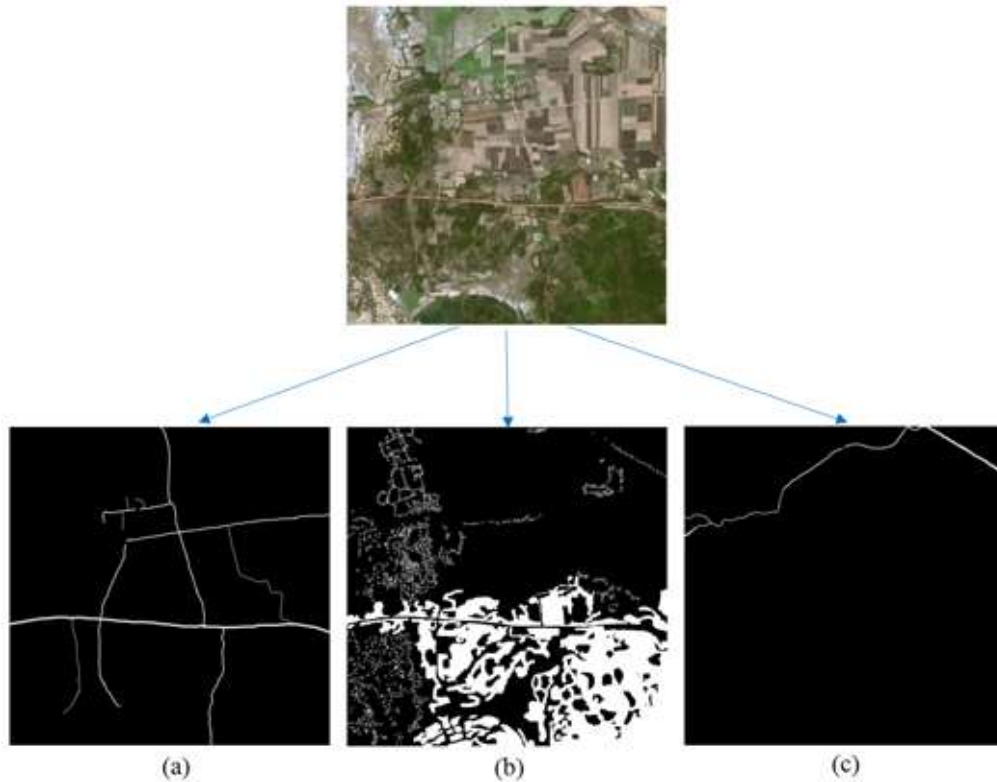


**Hình 2.** Chất lượng hình ảnh giảm khi địa hình được quan sát từ xa đến gần:  
(a) Quang cảnh địa hình ở khoảng cách xa; (b) Quang cảnh nhìn chi tiết của vùng A;  
(c) Quang cảnh nhìn chi tiết của vùng B

### III. PHÂN LỚP ĐỊA HÌNH VÀ XỬ LÝ CÁC LỚP ẢNH ĐẠI DIỆN

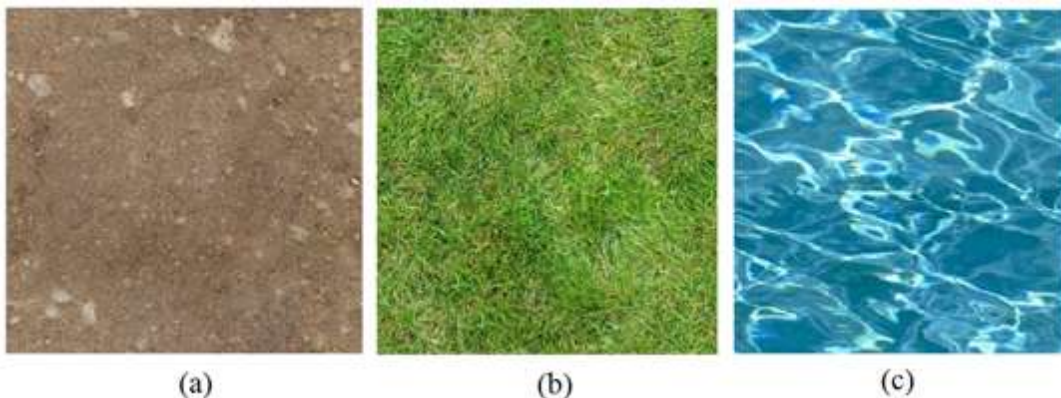
#### A. Phân lớp địa hình

Dựa trên thông tin ảnh vệ tinh, các đối tượng trên bề mặt địa hình được phân loại thành các lớp như: đường xá, thảm thực vật, sông ngòi. Các đối tượng trên thường có màu sắc khác nhau và dễ dàng phân biệt bằng mắt thường. Từ ảnh vệ tinh, các lớp địa hình trên có thể được phân loại thông qua các thuật toán phân loại dựa trên điểm ảnh, phân loại hướng đối tượng [11][12][13][15], hoặc qua phần mềm xử lý ảnh như photoshop, đây được xác định là bước tiền xử lý ảnh phủ. Ở mô tả sự phân lớp địa hình từ ảnh phủ địa hình qua phần mềm xử lý ảnh photoshop. Quá trình phân loại các lớp trong ảnh vệ tinh được tiến hành bằng cách sử dụng các màu cấp xám để đánh dấu các đường, hoặc các vùng của lớp đối tượng cần quan tâm. Cụ thể, một lớp ảnh mới được tạo và được tô màu đen (xác định các lớp đối tượng trên địa hình chưa được phân tách), được gọi là lớp ảnh phân loại. Sau đó, lớp ảnh vệ tinh được chọn làm nền cho việc phân tách các lớp và các kênh màu R, G, B của lớp ảnh phân loại lần lượt được sử dụng để đánh dấu các lớp đối tượng tương ứng là các con đường, các vùng thực vật, các đoạn sông (các đoạn hoặc các vùng được tô màu trắng). Bài báo đưa ra trường hợp mà các đối tượng trên bề mặt địa hình được phân loại thành 3 lớp là lớp đường, lớp thực vật và lớp sông ngòi, nhưng trong trường hợp tổng quát trên một bề mặt địa hình bất kỳ, số lượng các lớp được phân loại có thể lớn hơn 3.



**Hình 3.** Phân lớp địa hình từ ảnh vệ tinh: (a) Lớp đường xá; (b) Lớp thực vật; (c) Lớp sông ngòi

0 thể hiện các ảnh đại diện cho từng lớp. Ảnh đại diện của một lớp là ảnh chụp bề mặt lớp đó ở cự ly gần trong thực tế, nó thể hiện tương đối chi tiết và trung thực hình ảnh của lớp đó. Ví dụ, ảnh đại diện của lớp đường là ảnh mặt đường, ảnh đại diện của lớp thực vật cỏ là ảnh mặt cỏ, ảnh đại diện của lớp sông ngòi là ảnh mặt nước được chụp lại ở cự ly gần. Kết hợp các ảnh đại diện với ảnh phân loại của từng lớp sẽ nâng cao chất lượng hình ảnh bề mặt địa hình, cách thức kết hợp được trình bày chi tiết ở phần tiếp theo.



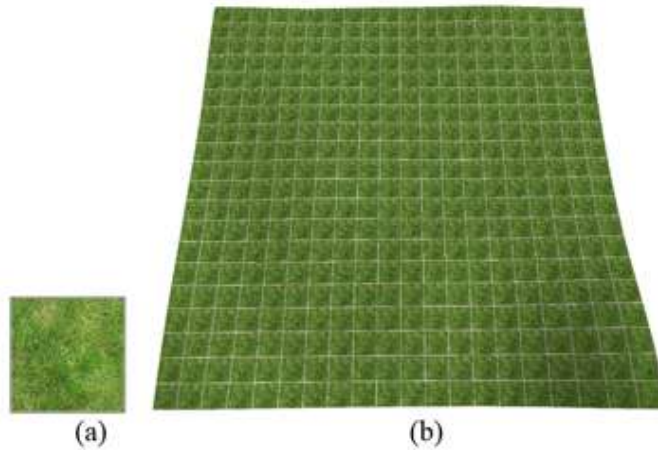
**Hình 4.** Ảnh đại diện của các lớp: (a) Lớp đường xá; (b) Lớp thực vật cỏ; (c) Lớp sông ngòi

### B. Xử lý ảnh đại diện của từng lớp trên bề mặt địa hình

Dữ liệu địa hình số hóa ba chiều thông thường là tập hợp của các mặt tam giác mà mỗi đỉnh ngoài thông tin tọa độ ba chiều  $(x, y, z)$  còn chứa thông tin tọa độ ảnh  $(u, v)$  có giá trị trong khoảng  $[0..1]$  được sử dụng cho quá trình ánh xạ ảnh lên trên bề mặt địa hình. Đối với địa hình có diện tích lớn, thay vì chỉ phủ một ảnh lên toàn bộ địa hình, số lượng ảnh phủ có thể tăng lên bằng cách thực hiện phép nhân tọa độ ảnh  $(u, v)$  với một hệ số  $d$  lớn hơn 1 ( $d$  là hệ số chi tiết do người dùng định nghĩa) để nhận được tọa độ ảnh chi tiết  $(u', v')$ :

$$(u', v') = d^*(u, v) \quad (1)$$

Áp dụng công thức (1), kết hợp với việc thiết lập chế độ ánh xạ ảnh là WRAP hoặc REPEAT cho cả hai trục  $u$  và  $v$ , một ảnh phủ đại diện sẽ được chi tiết hóa. 0 thể hiện hình ảnh địa hình được phủ ảnh đại diện được chi tiết hóa, với hệ số chi tiết bằng 20.



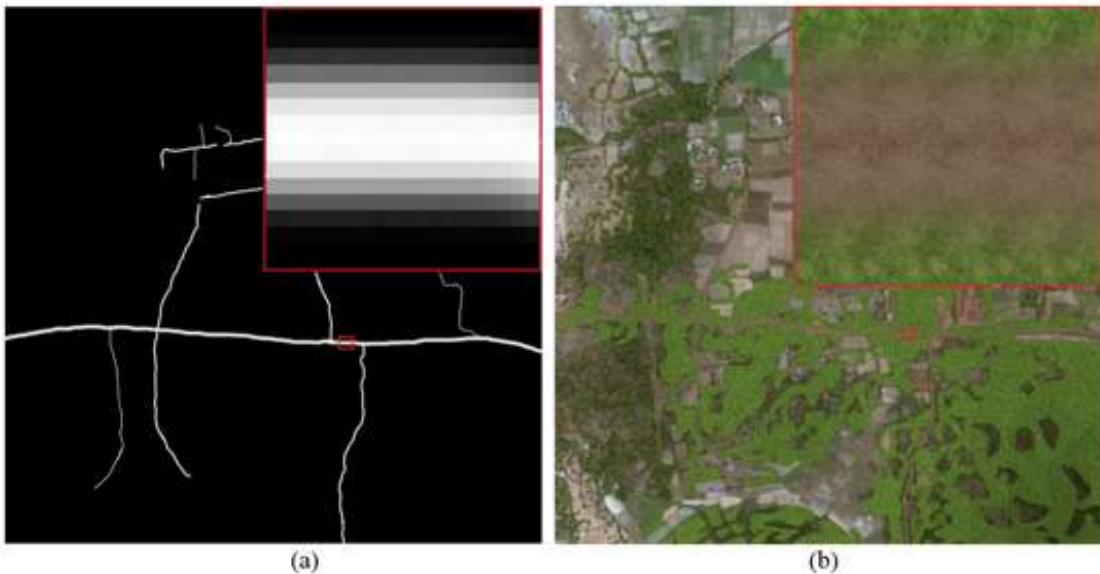
**Hình 5.** Chi tiết hóa ảnh đại diện của lớp thực vật: (a) Ảnh đại diện; (b) Bề mặt địa hình được phủ ảnh

Sau khi chi tiết hóa, ảnh đại diện của từng lớp được kết hợp với ảnh vệ tinh để trải lên các bề mặt lớp tương ứng nằm trên địa hình bằng kỹ thuật hòa trộn ảnh (blending texture). Nếu  $P_1$  là một điểm ảnh trong ảnh đại diện của lớp 1 và  $P_2$  là một điểm ảnh trong ảnh đại diện của lớp 2, thì  $P$  là điểm ảnh kết hợp màu giữa điểm ảnh  $P_2$  và  $P_1$  được thể hiện qua công thức sau:

$$P = (1 - w) * P_1 + w * P_2 \quad (2)$$

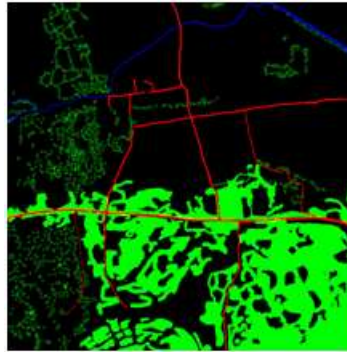
Trong đó,  $w$  là trọng số hòa trộn màu có giá trị trong khoảng  $[0..1]$ .

Thực tế, dữ liệu phân lớp trình bày trong mục A không chỉ lưu trữ giá trị 0 hoặc 255 để đặc tả lớp thông tin mà có thể lưu trữ giá trị bất kỳ trong khoảng  $[0..255]$  để mô tả ảnh hưởng của ảnh phủ tại vị trí đó. 0 mô tả ảnh hưởng của lớp đường trên bề mặt địa hình với dữ liệu ảnh phân lớp của lớp đường, vùng biên của lớp này sẽ mờ dần, tương ứng với giá trị điểm ảnh giảm dần từ 255 về 0, thể hiện vùng chuyển tiếp giữa lớp đường và lớp thực vật cỏ.

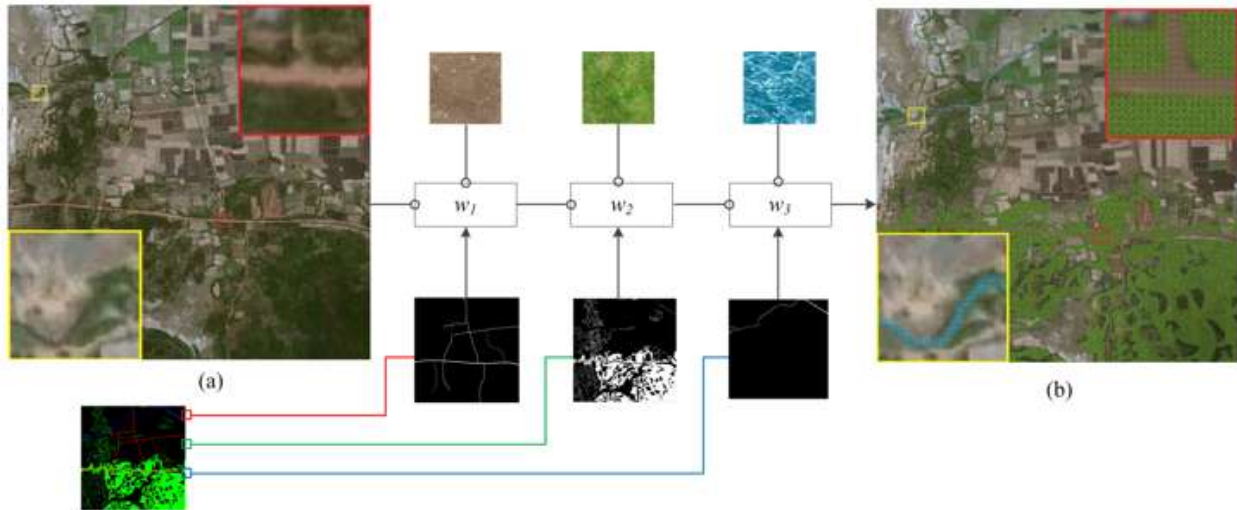


**Hình 6.** Ảnh hưởng của lớp đường trên bề mặt địa hình: (a) Ảnh phân lớp; (b) Địa hình được phủ ảnh dựa trên ảnh phân lớp

Các giá trị màu của các điểm ảnh trong dữ liệu ảnh phân lớp có thể được ánh xạ từ khoảng  $[0..255]$  về khoảng  $[0..1]$ , và được sử dụng như các trọng số hòa trộn màu của các lớp. Bên cạnh đó, để hạn chế việc phải nạp nhiều dữ liệu ảnh phân lớp của các lớp khi xử lý hình ảnh trên GPU, đồng thời chỉ cần truy vấn một lần để có được các giá trị trọng số hòa trộn màu của các lớp, thì dữ liệu phân lớp của lớp đường, lớp thực vật, lớp sông ngòi được lưu trữ lần lượt vào các kênh màu R, G, B của một ảnh được gọi là ảnh trọng số (0). Bên cạnh đó, như đã trình bày ở phần A, số lượng các lớp trên bề mặt địa hình có thể lớn hơn 3 và một ảnh có 4 kênh màu R, G, B, A nên nó có thể chứa tối đa 4 ảnh phân lớp. Do đó, nếu số lượng ảnh trọng số được sử dụng là  $n$  ( $n$  lớn hơn 1), thì số lượng các lớp trên bề mặt địa hình được phân loại và xử lý tối đa sẽ là  $4*n$ .



Hình 7. Ảnh trọng số

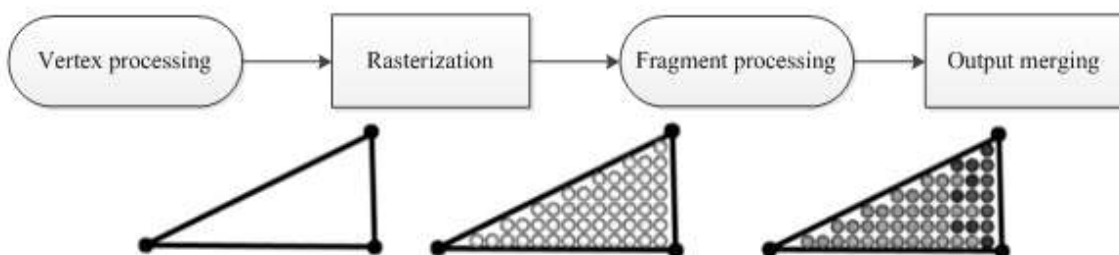


Hình 8. Phù ảnh đại diện của các lớp lên bề mặt địa hình  
(a) Địa hình được phù ảnh vệ tinh; (b) Địa hình được phù ảnh đại diện của các lớp

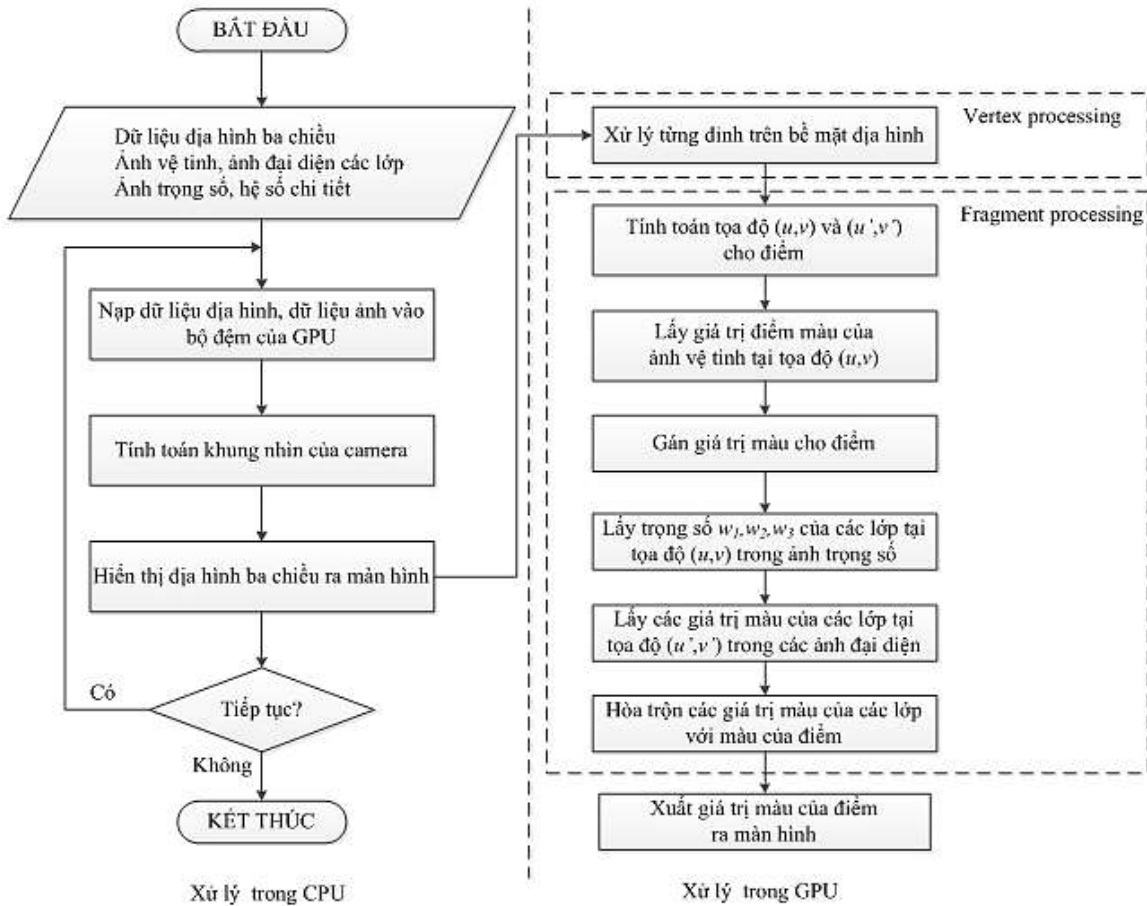
Ảnh trọng số được sử dụng trong quá trình phù ảnh đại diện của các lớp lên bề mặt địa hình (0). Đầu tiên, địa hình được phù ảnh vệ tinh (0a) được gọi là ảnh cơ sở. Tiếp theo, ảnh trọng số cung cấp các trọng số  $w_1$ ,  $w_2$ ,  $w_3$  của các lớp gồm: lớp đường, lớp thực vật và lớp sông. Các trọng số này được sử dụng để hòa trộn màu giữa ảnh đại diện của từng lớp với ảnh cơ sở áp dụng công thức (2). Kết quả nhận được là hình ảnh bề mặt địa hình được phù ảnh đại diện của các lớp (0b). Nhận thấy, chất lượng hình ảnh bề mặt địa hình được nâng cao hơn khi được phù thêm ảnh đại diện của các lớp.

### C. Sơ đồ thuật toán

Toàn bộ quá trình phù ảnh đại diện của các lớp lên bề mặt địa hình được thực hiện trên GPU. 0 mô tả chu trình xử lý đồ họa ba chiều cơ bản trong GPU. Từng mặt tam giác trong mô hình địa hình ba chiều được đưa vào chu trình xử lý đồ họa. Tại bước xử lý đỉnh (vertex processing) từng đỉnh trong mặt tam giác được xử lý, ví dụ các phép biến đổi không gian được thực hiện. Kế tiếp, mặt tam giác gồm tập các đỉnh đã được xử lý sẽ được phân rã thành các điểm (fragment) trong bước phân mảnh (rasterization). Tiếp theo, bước xử lý điểm (fragment processing) thực hiện cập nhật màu cho mỗi điểm. Cuối cùng, bước hợp nhất đầu ra (output merging) kết hợp giá trị màu của điểm đang xử lý với giá trị màu của điểm ảnh đã lưu trong bộ đệm màu để hiển thị ra màn hình. Trong chu trình này, bước phân mảnh và bước hợp nhất đầu ra là các bước xử lý cố định của card đồ họa, nhưng bước xử lý đỉnh và bước xử lý điểm cho phép can thiệp từ bên ngoài thông qua tập các câu lệnh của GPU.



Hình 9. Chu trình xử lý đồ họa ba chiều



**Hình 10.** Thuật toán chồng lớp và hiển thị ảnh đại diện của các lớp trên bề mặt địa hình

0 mô tả sơ đồ khối thuật toán chồng lớp và hiển thị ảnh đại diện của các lớp trên bề mặt địa hình. Dữ liệu đầu vào bao gồm: dữ liệu địa hình ba chiều, ảnh vệ tinh, ảnh đại diện của các lớp, ảnh trọng số, hệ số chi tiết. Theo cách xử lý thông thường trong CPU, dữ liệu địa hình và dữ liệu ảnh được nạp vào bộ đệm của GPU, sau đó khung nhìn của camera được tính toán để xác định không gian hiển thị hình ảnh trên màn hình, cuối cùng các lệnh vẽ được gọi và hình ảnh địa hình ba chiều được hiển thị trên màn hình. Tuy nhiên, các card đồ họa hiện nay đã cho phép can thiệp vào quá trình xử lý hình ảnh trên GPU. Tại bước xử lý đỉnh, từng đỉnh trên bề mặt địa hình được xử lý. Sau đó, tại bước xử lý điểm, thông tin tọa độ  $(u,v)$  và tọa độ  $(u',v')$  tính toán cho điểm được xác định theo công thức (1). Tiếp theo, giá trị điểm màu của ảnh vệ tinh tại tọa độ  $(u,v)$  được tính toán và gán cho điểm. Kế tiếp, các giá trị trọng số  $w_1, w_2, w_3$  của các lớp được truy xuất tại tọa độ  $(u,v)$  trong ảnh trọng số, các giá trị màu của các lớp được xác định tại tọa độ  $(u',v')$  trong các ảnh đại diện của các lớp. Cuối cùng, quá trình hòa trộn các giá trị màu của các lớp với điểm màu của điểm được thực hiện như đã trình bày ở phần III.B, giá trị màu của điểm được xuất ra màn hình.

**D. Thực nghiệm**

Thực nghiệm thuật toán được tiến hành trên máy tính có cấu hình Intel Core i5-4210H CPU 2.90GHz, RAM 8GB, card màn hình Nvidia GTX850M, độ phân giải 1920x1080. Môi trường lập trình là Microsoft Visual Studio 2010 kết hợp với thư viện lập trình đồ họa ba chiều DirectX 10. Bản chất của quá trình xử lý trên GPU là xử lý song song và hiển thị đồng thời các điểm ảnh, do đó tốc độ tính toán, xử lý và hiển thị các hình ảnh trên bề mặt địa hình ba chiều trên GPU nhanh hơn trên CPU.

0 mô tả bước xử lý mảnh trên GPU tính toán chồng lớp và hiển thị ảnh đại diện của các lớp đường xá, thực vật, sông ngòi trên bề mặt địa hình số hóa ba chiều. Đầu vào dữ liệu gồm: các ảnh vệ tinh, ảnh đại diện của các lớp, ảnh trọng số (dòng 1, 2, 3), và một số thiết lập đầu vào cho bộ lọc ảnh và hệ số chi tiết (dòng 4, 5). Đầu tiên, các giá trị tọa độ  $(u,v)$  và  $(u',v')$  của điểm được tính toán (dòng 7, 8). Tiếp theo, các giá trị màu của ảnh vệ tinh, ảnh đại diện của các lớp được xác định (dòng 9, 10, 11, 12) và trọng số của các lớp được truy xuất từ ảnh trọng số (dòng 13) thông qua tọa độ  $(u,v)$  và  $(u',v')$ . Cuối cùng, giá trị màu của điểm được tính toán bằng kỹ thuật hòa trộn ảnh, qua các hàm nội suy tuyến tính lerp của GPU (dòng 14). Tính thời gian thực của thuật toán được thể hiện thông qua tốc độ xử lý và hiển thị hình ảnh. Tốc độ hiển thị hình ảnh thực nghiệm nhận được xấp xỉ 150 fps và 100 fps với kích thước cửa sổ hiển thị hình ảnh là 512x512 và 1024x1024, đáp ứng được tính thời gian thực trong các ứng dụng mô phỏng.

```

1. Texture2D mapTexture; // ảnh vệ tinh
2. Texture2D roadTexture, grassTexture, riverTexture; // ảnh đại diện của các lớp
3. Texture2D weightTexture; // ảnh trọng số
4. SamplerState linearFilter; // bộ lọc ảnh
5. float detailFactor; // hệ số chi tiết

6. float4 main(PixelInputType input) : SV_Target
{
    // tính toán tọa độ (u,v), (u',v') của điểm
7. float2 globalCoord = input.tex;
8. float2 detailCoord = globalCoord * detailFactor;

    // lấy giá trị màu của ảnh vệ tinh và của ảnh đại diện của lớp
9. float4 mapTextureColor = mapTexture.Sample(linearFilter, globalCoord);
10. float4 roadTextureColor = roadTexture.Sample(linearFilter, detailCoord);
11. float4 grassTextureColor = grassTexture.Sample(linearFilter, detailCoord);
12. float4 riverTextureColor = riverTexture.Sample(linearFilter, detailCoord);

    // lấy các trọng số của các lớp
13. float4 weightMap = weightTexture.Sample(linearFilter, globalCoord);

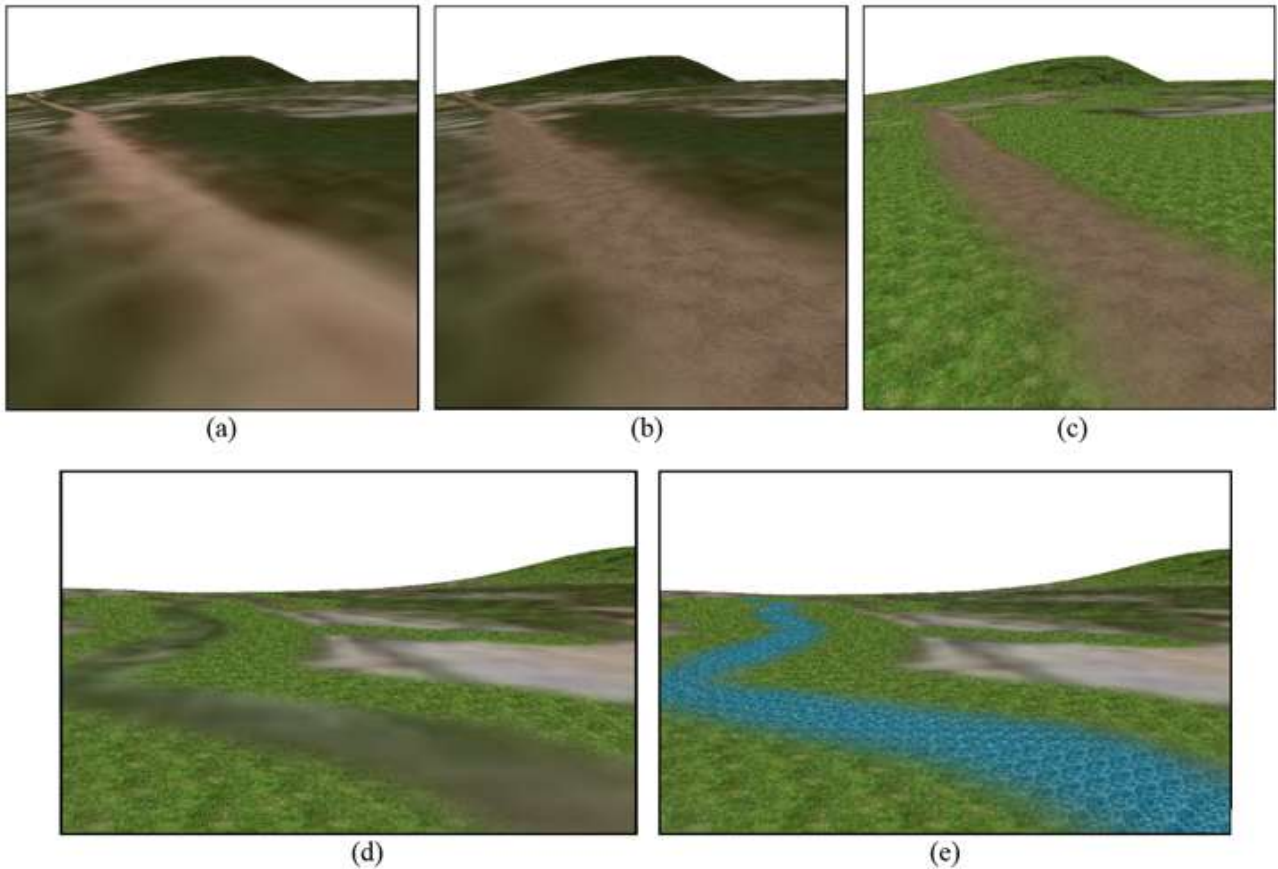
    // hòa trộn các giá trị màu của các lớp
14. float4 color = lerp(mapTextureColor, roadTextureColor, weightMap.r);
    color = lerp(color, grassTextureColor, weightMap.g);
    color = lerp(color, riverTextureColor, weightMap.b);

    return color;
}

```

**Hình 11.** Bước xử lý điểm trong tính toán và hiển thị bề mặt địa hình

0 thể hiện hình ảnh bề mặt địa hình khi được phủ ảnh vệ tinh và ảnh đại diện của các lớp. Nhận thấy, chất lượng hình ảnh được nâng cao hơn khi bề mặt địa hình được phủ thêm ảnh đại diện của từng lớp.



**Hình 12.** Hình ảnh bề mặt địa hình

- (a) Khi phủ ảnh vệ tinh; (b) Khi phủ ảnh lớp đường; (c) Khi phủ ảnh lớp thực vật  
 (d) Trước khi phủ ảnh lớp sông; (e) Sau khi phủ ảnh lớp sông

#### IV. KẾT LUẬN

Các ứng dụng mô phỏng trong quân sự thường có khối lượng tính toán rất lớn, trong đó yêu cầu về độ chính xác trong tính toán được đặt lên hàng đầu. Tuy nhiên, nhu cầu nâng cao chất lượng hình ảnh hiển thị đang ngày càng được chú trọng để tăng tính trực quan sinh động cho chương trình mô phỏng. Với sự phát triển của công nghệ phần cứng, các card đồ họa có năng lực tính toán ngày càng nhanh hơn và hỗ trợ lập trình can thiệp vào quá trình xử lý hình ảnh nhiều hơn. Nhưng để có thể khai thác được tiềm năng đó, yêu cầu đặt ra là ta phải cải tiến các thuật toán, đưa ra các cách xử lý mới phù hợp và phát huy được nền tảng công nghệ mới.

Trong thực tế huấn luyện và chiến đấu, người chỉ huy cần phải nắm bắt được chính xác đặc điểm địa hình khu vực chiến đấu, do đó thông tin về địa hình được thể hiện càng chi tiết càng tốt. Tuy nhiên, hình ảnh hiển thị càng chi tiết, sinh động thì thời gian xử lý hình ảnh yêu cầu càng nhiều, ảnh hưởng đến các tính toán khác trong ứng dụng mô phỏng, dẫn đến tốc độ xử lý của chương trình không đảm bảo thời gian thực. Bài báo này đưa ra cách thức xử lý, nâng cao chất lượng hiển thị hình ảnh bề mặt địa hình ứng dụng công nghệ lập trình GPU, đồng thời tốc độ hiển thị vẫn đảm bảo tính thời gian thực, đáp ứng được yêu cầu của ứng dụng mô phỏng. Kết quả nghiên cứu này có thể áp dụng vào trong các ứng dụng mô phỏng tác chiến, cung cấp thêm thông tin trực quan cho người chỉ huy trong quá trình xây dựng phương án huấn luyện và chiến đấu cho đơn vị.

#### TÀI LIỆU THAM KHẢO

- [1] Jim Durbinr, Edward Swan, Brad Colbert, John Crowe, Rob King, Tony King, Battlefield visualization on the responsive workbench, IEEE Visualization proceedings, 1998.
- [2] M. Koutek, F.H. Post, The responsive workbench simulator: a tool for application development and analysis, WSCG proceedings, 2002.
- [3] Baki Koyuncu, Erkan Bostanci, 3D battlefield modeling and simulation of war games, Communications and Information Technology proceedings, 2009.
- [4] Imran A Syed, Nageli VS, Sangeeta S, S Rakshit, Digital Sand Model using Virtual Reality Workbench, International Conference on Computer Communication and Informatics (ICCCI -2012), Jan. 10 – 12, 2012.
- [5] Richard S. Wright, Nicholas Haemel, Graham Sellers, Benjamin Lipchak, OpenGL SuperBible: Comprehensive Tutorial and Reference, 5<sup>th</sup>, Addison-Wesley Professional, 2010.
- [6] Marc Olano, A Programmable Pipeline for Graphics Hardware, PhD. Thesis. University of North Carolina at Chapel Hill, Department of Computer Science, 1998.
- [7] Randima Fernando, Mark J.Kilgard, The Cg Tutorial: The Definitive Guide to Programmable Real-Time Graphics, Addison-Wesley Longman, 2003.
- [8] JungHyun Han, 3D Graphics for Game Programming, A Chapman and Hall/CRC, 2011.
- [9] Johan Andersson, Terrain Rendering in Frostbite using Procedural Shader Splatting, Siggraph, 2007.
- [10] Flora Ponjou Tasse, Distributed Texture-based Terrain Synthesis, Master Thesis, University of Cape Town, South Africa, 2011.
- [11] Paul Mather, Brandt Tso, Classification Methods for Remotely Sensed Data, 2<sup>nd</sup>, CRC Press, May 12, 2009.
- [12] Z HANG , H., O UYANG , D., L IN , H., AND G UAN, Texture synthesis based on terrain featurerecognition, International Conference on Computer Science and Software Engineering. Vol. 2. IEEE Computer Society, Washington, DC, USA, 1158–1161, 2008.
- [13] Long Thanh Ngo, Dinh Sinh Mai, Witold Pedrycz, Semi-supervising Interval Type-2 Fuzzy C-Means clustering with spatial information for multi-spectral satellite image classification and change detection, Computers & Geosciences 83 (2015).
- [14] Nguyễn Trung Kiên, Lê Anh, Nguyễn Huy Liêm, Học viện Kỹ thuật Quân sự, Mô phỏng trường quan sát của người lính trong môi trường tác chiến ảo, Tạp chí Khoa học và Kỹ thuật – Học viện Kỹ thuật Quân sự số 157 (ISSN-1859-0209), 10/2013.
- [15] Trịnh Thị Hoài Thu, Lê Thị Thu Hà, Phạm Thị Lân, So sánh phương pháp phân loại dựa vào điểm ảnh và phân loại định hướng đối tượng chiết xuất thông tin lớp phủ bề mặt từ ảnh độ phân giải cao, Trường Đại học Mô - Địa chất, Tạp chí KHKT Mô - Địa chất, số 39, 7/2012.

## PROCESSING AND RENDERING TEXTURE LAYERS IN A THREE-DIMENSIONAL DIGITAL TERRAIN

Le Anh, Nguyen Thi Lan, Nguyen Trung Kien

**ABSTRACT**—Three-dimensional digital terrain plays an important role in the military simulation applications. Visual images of the terrain supports commanders to evaluate the characteristics and topography values correctly, as the basis for the preparation of combat plans. Three-dimensional digital terrain typically consists of a set of triangular surfaces and textures. The whole surface is covered by the texture in rendering (each pixel is covered with a certain terrain area). Therefore, the closer the surface is, the fewer the number of pixels we can observe. To improve the rendering quality of terrain surfaces, while maintaining the rendering speed in real time, an approach is proposed to classify terrain data and to process texture layers based on GPU architecture.

**Keywords**—Simulation, three-dimentional digital terrain, texture, GPU programming, shader.