

MỘT THUẬT TOÁN RÚT GỌN BỀ MẶT BIỂU DIỄN MÔ HÌNH 3D

ĐỖ NĂNG TOÀN, NGUYỄN VĂN HUÂN

1. GIỚI THIỆU

Xây dựng mô hình là một khâu quan trọng trong các hệ thống thực tại ảo. Mô hình được xây dựng không những phải đảm bảo về chất lượng mà còn phải đảm bảo về yêu cầu giảm thiểu không gian bộ nhớ, rút ngắn thời gian tính toán, nhằm phục vụ cho các yêu cầu điều khiển sau đó. Điều này, đặc biệt quan trọng khi sử dụng các thiết bị thu nhận như máy quét 3D, mô hình thu được thường có kết quả quá "nặng" [4, 15].

Để giảm thiểu không gian bộ nhớ của mô hình, một trong những cách tiếp cận là giảm thiểu số bề mặt biểu diễn mô hình. Các thuật toán "giảm thiểu bề mặt" biểu diễn một vật thể trong không gian ba chiều hiện nay là không nhiều đối với đa số các ứng dụng thực tại ảo. Trong thực tế, không phải lúc nào chúng ta cũng cần biểu diễn một vật thể trong không gian ba chiều với chất lượng như thật. Hơn nữa, mức độ chất lượng hình ảnh của cùng một vật thể khi hiển thị cũng có thể thay đổi theo cách nhìn (view). Ví dụ, khi quan sát cận cảnh một vật thể, chúng ta cần số lượng mặt nhiều hơn để việc mô hình hoá có thể đạt đến một chất lượng như thật [8]. Tuy nhiên, khi khoảng cách tăng lên, vật thể càng nhỏ đi đối với mắt người quan sát thì số lượng mặt biểu diễn có thể giảm đi mà vẫn đảm bảo cho người quan sát vẫn có thể hình dung ra được vật thể. Kỹ thuật hiển thị này đã được tích hợp vào nhiều hệ thống đồ họa 3 chiều và thực tại ảo với tên gọi là LOD (Level Of Detail) [12, 14].

Để thực hiện được kỹ thuật hiển thị LOD về cơ bản người ta phải sinh ra các mô hình ở các mức độ chi tiết khác nhau và kỹ thuật sẽ sử dụng mô hình phù hợp tại các tình huống hiển thị khác nhau. Việc sinh ra các mô hình này thường dựa trên các kỹ thuật rút gọn bề mặt hoặc trường hợp mô hình được biểu diễn bởi các bề mặt có phương trình tham số, thì chúng sẽ được sinh ra bởi các tham số phù hợp [1, 2]. Hình 1 dưới đây minh họa cho điều này.



Hình 1. Minh họa về việc giảm thiểu bề mặt theo các mức khác nhau

Như vậy, trong các ứng dụng đồ họa ba chiều nảy sinh một vấn đề là nghiên cứu một lớp các thuật toán nhằm giảm bớt số mặt của một đa diện biểu diễn vật thể sao cho vẫn giữ lại được một cách tương đối hình dạng ban đầu của vật thể đó. Tùy theo cách phân loại, các tiêu chí phân loại có thể kể tới là:

- *Dựa trên ứng dụng*: Một số thuật toán giới hạn các đối tượng vật thể của mình, ở một số dạng nhất định tùy theo ứng dụng của nó. Ví dụ: có một loạt thuật toán chỉ nhằm giảm thiểu mặt của địa hình được dùng trong các ứng dụng GIS hay các bài toán mô phỏng địa hình. Một số khác được dành cho các ứng dụng tổng quát nên các ràng buộc vào dữ liệu đầu vào ít hơn.

- *Dựa trên các ràng buộc ban đầu*: Có những thuật toán chú trọng tới việc giữ lại hình dạng vật thể (TPA). Tuy nhiên, cũng có những thuật toán chấp nhận mất mát bớt đi một phần thông tin để có thể đạt được tốc độ xử lý cao.

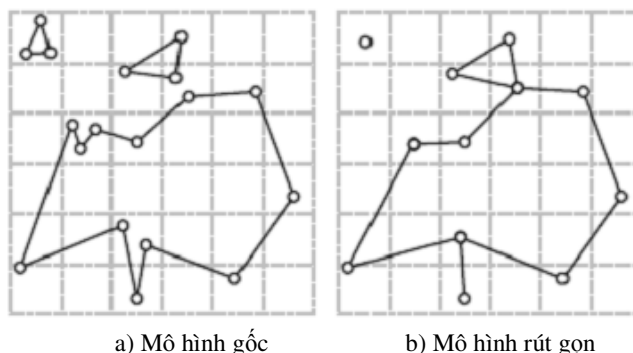
- *Dựa trên phương pháp đánh giá sai số*: Một trong những điều quan trọng nhất của các thuật toán “giảm thiểu bề mặt” là cách thức đánh giá sai số của vật thể mới tạo ra so với vật thể ban đầu, qua đó xác định phương án giảm thiểu nào là tốt, phương án nào là kém.

Bài báo này đề cập một kĩ thuật rút gọn số lượng bề mặt biểu diễn mô hình theo cách tiếp cận dựa trên ứng dụng, thuật toán áp dụng cho các mô hình có các bề mặt gần nhau là tương đối “bằng phẳng”, trên cơ sở mở rộng thuật toán rút gọn số lượng điểm biểu diễn Angles [13,19] trong không gian 2D. Phần còn lại của bài báo được thể hiện như sau: Phần 2 trình bày một số thuật toán rút gọn số lượng bề mặt biểu diễn mô hình 3D đã được phát triển. Phần 3 đề cập một thuật toán rút gọn số lượng bề mặt biểu diễn mô hình 3D dựa trên cơ sở thuật toán Angles. Tiếp theo là thử nghiệm và cuối cùng là kết luận về kĩ thuật đề xuất.

2. MỘT SỐ NGHIÊN CỨU LIÊN QUAN

Trong một vài năm gần đây, bài toán rút gọn số lượng bề mặt biểu diễn mô hình theo cách tiếp cận dựa trên ứng dụng đã nhận được nhiều sự quan tâm nghiên cứu và luôn là vấn đề được đặt ra khi mà trong thực tế vẫn thường xuyên có những yêu cầu cụ thể. Tùy thuộc vào từng ứng dụng cụ thể với những yêu cầu cụ thể mà hiện nay người ta có thể thực hiện việc rút gọn mô hình dựa vào những đặc trưng khác nhau của từng mô hình.

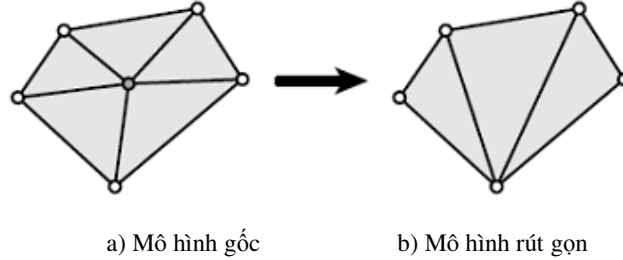
Với những ứng dụng mà mô hình gồm có những bề mặt rời rạc gần nhau hoặc có tồn tại những cặp (v_i, v_j) không phải là cạnh thì người ta áp dụng giải thuật bó đỉnh hoặc giải thuật NEC (*Non-Edge Contraction*) [5, 9].



Hình 2. Minh họa giải thuật bó đỉnh hoặc NEC

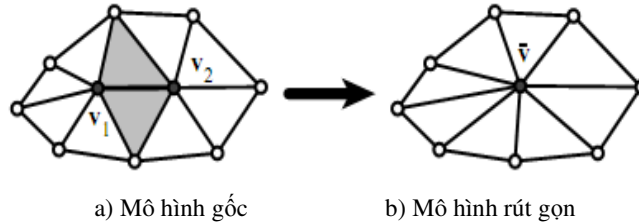
Với giải thuật bó đỉnh cho phép thực hiện bó các đỉnh gần nhau thành một đỉnh đơn (hình 2b), với giải thuật này cho kết quả tối và không hỗ trợ điều khiển mô hình xấp xỉ.

Với những ứng dụng mà mô hình gồm có những bề mặt cong, lồi lõm, có đỉnh không quan trọng thì người ta áp dụng giải thuật chọn đỉnh và xóa bỏ đỉnh này đi, rồi xóa bỏ các bề mặt gần kề. Nhưng giải thuật này không phù hợp cho các mô hình đa tạp [5, 9].

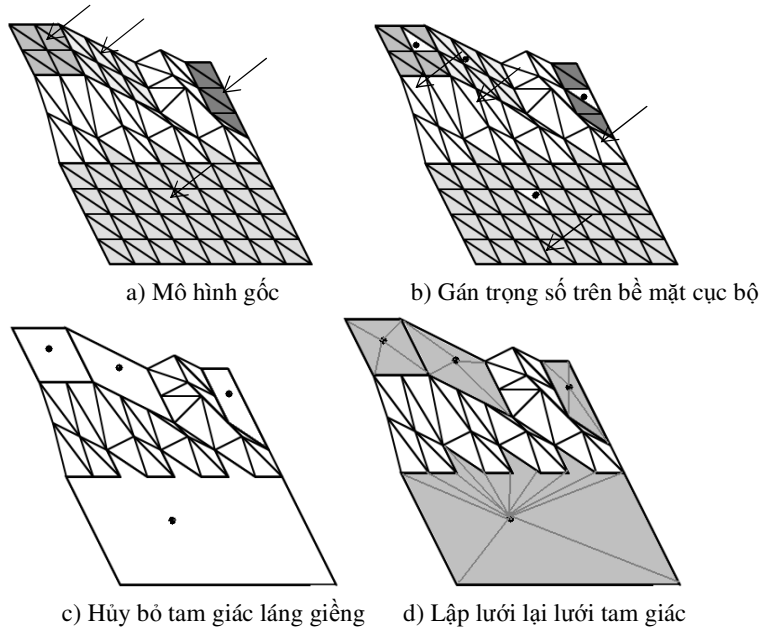


Hình 3. Minh họa giải thuật xoá đỉnh và lập lại lưới tam giác

Với những ứng dụng mà mô hình gồm có những bề mặt mà có các cạnh v_i, v_j với $\|v_i - v_j\| < t$ (t là ngưỡng) thì người ta áp dụng giải thuật chụm cạnh EC (*Edge Contraction*), rồi xoá bỏ những bề mặt có cạnh chụm [5, 6, 7, 9].



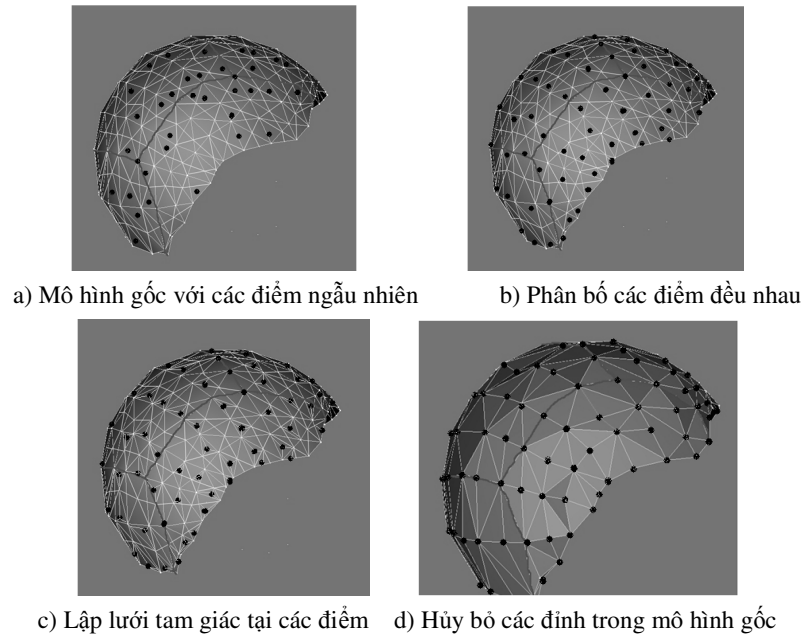
Hình 4. Minh họa giải thuật chụm cạnh ($v_1; v_2$) và huỷ bỏ các mặt có chung cạnh



Hình 5. Minh họa giải thuật cắt bỏ hình học

Với những ứng dụng mà mô hình gồm có các vùng bề mặt có độ cong cao và độ cong thấp thì người ta áp dụng giải thuật cắt bỏ hình học bằng cách gán trọng số cho những bề mặt trong mô hình gốc theo độ cong cục bộ [17].

Với những ứng dụng mà mô hình gồm có những bề mặt cong cao hơn so với những đối tượng chứa những điểm gián đoạn (không liên tục) có cạnh nhọn (có độ cong, đường cong đột ngột) thì người ta áp dụng giải thuật Re-Tiling. Ví dụ: Rút gọn bề mặt một con thỏ, lông thỏ tốt, nhưng không tốt cho mô hình ngôi nhà (kết quả tồi). Với kĩ thuật này yêu cầu người dùng chỉ ra số đỉnh trong mô hình đầu ra [19].

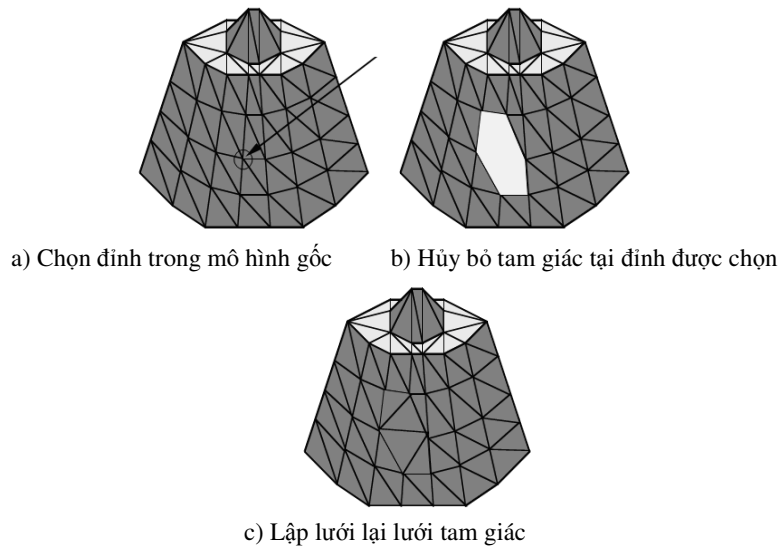


Hình 6. Minh họa giải thuật Re-Tiling

Với những ứng dụng mà xây dựng mô hình từ các hình khối MC (*Marching Cubes*) thì người ta áp dụng giải thuật DTM (*Decimation of Triangle Meshes*). Giải thuật thực hiện lựa chọn các đỉnh mà thỏa mãn một điều kiện sai số khoảng cách, rồi xóa bỏ đỉnh này, sau đó thực hiện lặp lập lưới tam giác [18].

Ngoài những ứng dụng với những mô hình có các bề mặt với những đặc trưng như trên, thì hiện nay vẫn còn có những ứng dụng với những mô hình với những đặc trưng khác mà cần phải được rút gọn số lượng bề mặt xuống mức tối thiểu sao cho chất lượng vẫn đảm bảo, chẳng hạn như những ứng dụng mà mô hình gồm có những bề mặt gần nhau là tương đối bằng phẳng.

Mục 3 dưới đây trình bày kĩ thuật rút gọn số lượng bề mặt biểu diễn mô hình 3D dựa trên ứng dụng, với các mô hình có các bề mặt gần nhau là tương đối bằng phẳng nhờ việc tính toán góc giữa các vectơ pháp tuyến bề mặt kề nhau, trên cơ sở mở rộng thuật toán Angles trong không gian 2D.

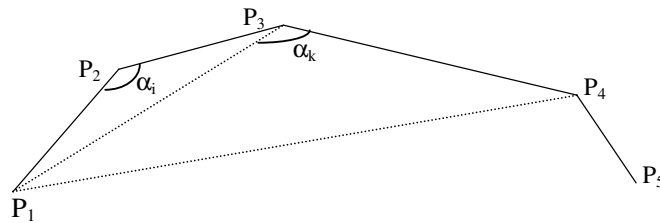


Hình 7. Minh họa giải thuật DTM

3. THUẬT TOÁN RÚT GỌN SỐ LƯỢNG BỀ MẶT BIỂU DIỄN MÔ HÌNH 3D

3.1. Thuật toán góc Angles[13, 16] trong không gian 2D

Thuật toán thực hiện việc rút gọn số lượng điểm biểu diễn cho một đường cong 2D. Thuật toán bắt đầu bằng việc xác định điểm đầu tiên trên đường cong và coi đó như là một điểm chốt (P1). Điểm thứ ba (P3) được coi là điểm động. Điểm giữa điểm chốt và điểm động (P2) là điểm trung gian. Ban đầu góc tại điểm trung gian được tạo bởi các cạnh nối điểm trung gian giữa điểm chốt và điểm động được tính toán và kiểm tra. Nếu giá trị tính được này lớn hơn một ngưỡng θ cho trước thì điểm trung gian có thể bỏ đi, tiến trình tiếp tục với điểm chốt là điểm chốt cũ, điểm trung gian là điểm động cũ và điểm động là điểm kế tiếp sau điểm động cũ. Trong trường hợp ngược lại, góc tính được nhỏ hơn ngưỡng θ cho trước thì điểm trung gian sẽ được giữ lại, tiến trình tiếp tục với điểm chốt là điểm trung gian, điểm trung gian là điểm động cũ và điểm động là điểm kế tiếp sau điểm động cũ. Tiến trình được lặp cho đến hết đường cong (hình 8 minh họa thuật toán Angles).



Hình 8. Đơn giản hóa đường cong với thuật toán Angles

3.2. Thuật toán loại bỏ các điểm góc Angles

Bước 1: Xác định điểm đầu tiên trên đường cong và coi đó như là một điểm chốt (P_1). Điểm thứ ba (P_3) được coi là điểm động. Điểm giữa điểm chốt và điểm động (P_2) là điểm trung gian.

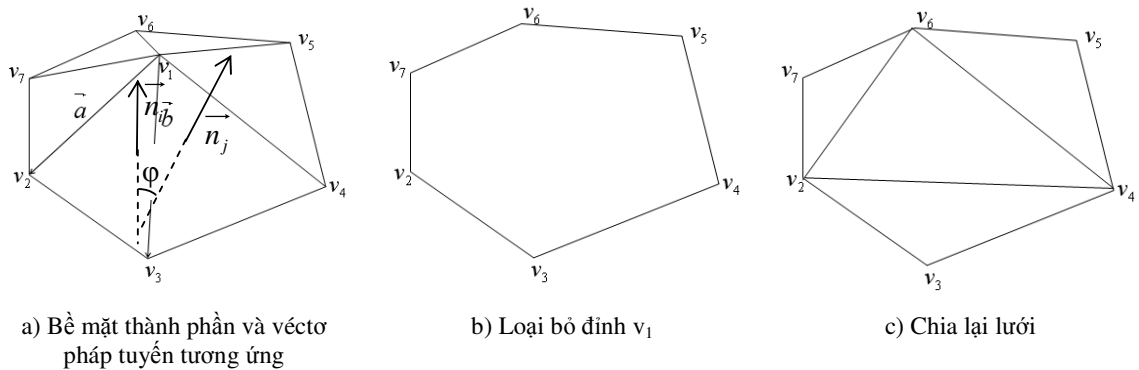
Bước 2: Tính góc tại điểm trung gian tạo bởi các cạnh nối điểm trung gian với điểm chốt và điểm động.

Bước 3: Kiểm tra góc tìm được nếu lớn hơn một ngưỡng θ cho trước thì điểm trung gian có thể bỏ đi. Trong trường hợp ngược lại điểm chốt chuyển đến điểm trung gian.

Bước 4: Chu trình được lặp lại cho đến hết đường cong với điểm trung gian được chuyển đến điểm động và điểm kế tiếp sau điểm động được chỉ định làm điểm động mới.

3.3. Thuật toán rút gọn số lượng mặt biểu diễn mở rộng HS

Thuật toán được thực hiện như là việc mở rộng của thuật toán Angles sang 3 chiều với việc thay tính góc tạo bởi các cạnh tại một đỉnh bởi tính góc tạo bởi vectơ pháp tuyến của các mặt có cùng một đỉnh. Xuất phát từ việc che phủ đối tượng có thể hình dung như sau: Coi ảnh che phủ như một miếng cao su, sau đó đóng đinh các điểm trên ảnh tại các vị trí tương ứng của lưới. Ảnh sẽ bị căng ra và che phủ toàn bộ đối tượng. Tại các điểm chốt mà vectơ pháp tuyến tại các mặt có chứa điểm này lệch nhau một góc nhỏ hơn ngưỡng θ cho trước thì điểm đó có thể bỏ đi, đa giác được tạo bởi các điểm đỉnh có cạnh nối điểm được bỏ đi sẽ được chia lưới lại theo các thuật toán về chia lưới tam giác (hình 9).



Hình 9. Vectơ pháp tuyến của các bề mặt chung đỉnh và sự loại bỏ đỉnh

Do có thể biểu diễn bề mặt đa giác thành các tam giác thành phần, nên không làm mất tính tổng quát ta có thể giả thiết rằng một đối tượng bề mặt lưới trong không gian ba chiều M chỉ bao gồm các mặt tam giác và được biểu thị bởi cặp danh sách (V, F) trong đó:

$V = (v_1, v_2, v_3, \dots, v_r)$: là danh sách gồm một dãy thứ tự các đỉnh, mỗi một đỉnh là một vectơ cột $v_i = [x_i, y_i, z_i]^T$; ($i = 1, 2, 3$) trong không gian R^3 .

$F = (f_1, f_2, f_3, \dots, f_n)$: là danh sách dãy có thứ tự các mặt tam giác, mỗi một tam giác f_i là một bộ ba đỉnh (v_i, v_k, v_l) .

Khi đó vectơ pháp tuyến của bề mặt tam giác f_i được tính bởi tích có hướng của hai vectơ $\vec{a} = (a_1, a_2, a_3)$ và $\vec{b} = (b_1, b_2, b_3)$ nằm trong bề mặt:

$$\vec{n}_i = \vec{a} \Lambda \vec{b} = \left(\begin{array}{c|c|c} a_2 & a_3 & a_1 \\ b_2 & b_3 & b_1 \end{array} \middle| \begin{array}{c} a_1 \\ a_2 \end{array} \right) = (a_2 b_3 - a_3 b_2, a_3 b_1 - a_1 b_3, a_1 b_2 - a_2 b_1). \quad (*)$$

Góc φ giữa hai pháp tuyến \vec{n}_i và \vec{n}_j của các bề mặt tam giác f_i, f_j được xác định bởi công thức:

$$\cos \varphi = \frac{\vec{n}_i \cdot \vec{n}_j}{\|\vec{n}_i\| \|\vec{n}_j\|} = \frac{a_1 a_2 + b_1 b_2 + c_1 c_2}{\sqrt{a_1^2 + b_1^2 + c_1^2} \cdot \sqrt{a_2^2 + b_2^2 + c_2^2}}. \quad (**)$$

Do đó, việc rút gọn số lượng bề mặt biểu diễn được thực hiện lặp thông qua 4 bước cơ bản.

3.4. Thuật toán rút gọn bề mặt biểu diễn HS

Bước 1: Khởi tạo

- Lựa chọn ngưỡng θ ;
- Tính vectơ pháp tuyến \vec{n}_i của các bề mặt tam giác $f_i \in F, i = 1, \dots, n$.

Bước 2: Với mỗi đỉnh $v_i \in V$ tính các góc được tạo bởi các cặp vectơ pháp tuyến của các bề mặt tam giác chứa đỉnh v_i

Bước 3: Nếu các góc tính được ở bước 2 đều nhỏ hơn ngưỡng θ cho trước, tức là các bề mặt tương đối phẳng tại đỉnh v_i , thì điểm đỉnh v_i có thể bỏ đi, thuật toán chuyển qua bước 4. Trong trường hợp ngược lại đỉnh được giữ lại, thuật toán quay lại bước 2.

Bước 4: Chia đa giác được tạo bởi các điểm đỉnh có cạnh nối điểm được bỏ đi thành lưới tam giác và cập nhật lại danh sách F

3.5. Độ phức tạp của thuật toán HS

Để đánh giá hiệu quả của thuật toán HS, chúng tôi dựa vào độ phức tạp của thuật toán.

Giả sử có mô hình tam giác M với v đỉnh và x bề mặt, n vectơ pháp tuyến được tạo bởi từ x bề mặt có chung đỉnh v .

Tại bước (1): Vì (*) có độ phức tạp tính toán là $O(1)$, mỗi vòng lặp tính toán một \vec{n} . Vì vậy có i vòng lặp thì chúng ta cần tính \vec{n}_i , do đó nếu có n vectơ pháp tuyến (tương ứng n bề mặt) thì có độ phức tạp là $O(n)$.

Tại bước (2)-(3), ta có 2 vectơ pháp tuyến tạo thành một góc cần tính. Nếu đỉnh v có x mặt phẳng chứa nó (x vectơ pháp tuyến) thì ta có:

$$C_x^2 = \frac{x!}{2!(x-2)!} = \frac{1}{2} (n-1)(n-2) \Rightarrow O(n^2).$$

Tại bước (4): Để chia được thành lưới các điểm thì thuật toán chia cần phải duyệt các điểm đã được bỏ đi. Giả sử độ phức tạp của thuật toán chia là $O(\log n)$. Cập nhật lại danh sách cần thực hiện tối thiểu n phép tính, do đó độ phức tạp là $O(n)$

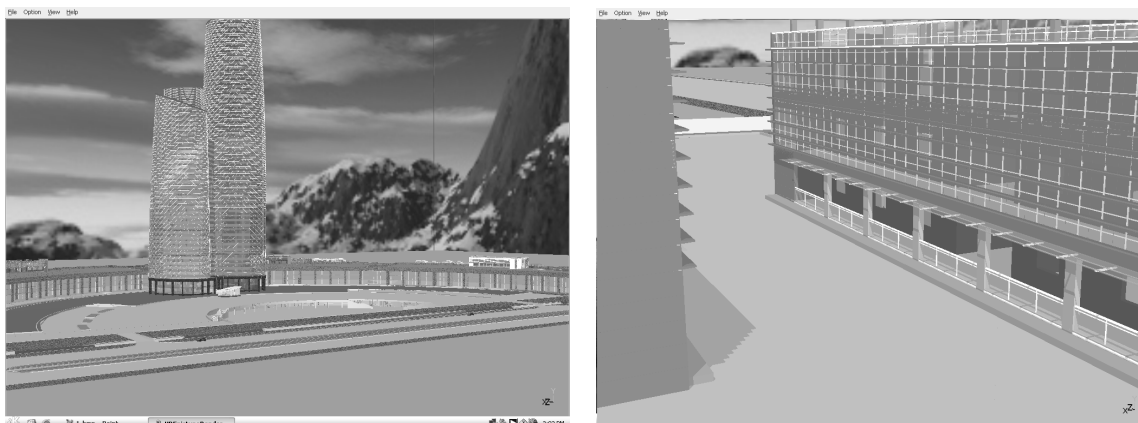
Vậy độ phức tạp của thuật toán HS là:

$$\text{Max}(O(n), O(n^2), O(n), O(\log n)) = O(n^2)$$

3.6. Thử nghiệm

Chúng tôi đã cài đặt thuật toán đề xuất ở trên và rút gọn số lượng bề mặt biểu diễn trong xây dựng mô hình có bề mặt tương đối bằng phẳng. Khi bề mặt đối tượng "bằng phẳng" thì sẽ có nhiều bề mặt tạo nên lưới đối tượng có cùng một điểm đỉnh có vectơ pháp tuyến "song song" với nhau do đó, góc tạo giữa chúng sẽ nhỏ. Do vậy, số điểm đỉnh có thể bỏ đi nhiều hơn dẫn đến số lượng bề mặt lưới sẽ được rút bỏ nhiều hơn.

Hình 10 dưới đây minh họa cho việc rút gọn số lượng bề mặt biểu diễn của thuật toán HS. Hình 10.a, 10.b là mô hình tổng thể và một phần của khu đô thị mới Nghĩa Tân. Số lượng bề mặt ban đầu của hệ thống là hơn 80.000.000 bề mặt. Với số lượng bề mặt lớn thế này, việc load và điều khiển mô hình là rất khó khăn. Với ngưỡng góc giữa 2 vectơ pháp tuyến $\leq 5^\circ$, chúng tôi đã rút gọn được xuống còn dưới 54.000.000 bề mặt. Nhờ đó chương trình trình diễn công trình kiến trúc VREpistypeRender [11] có thể điều khiển khá linh hoạt công trình kiến trúc này.

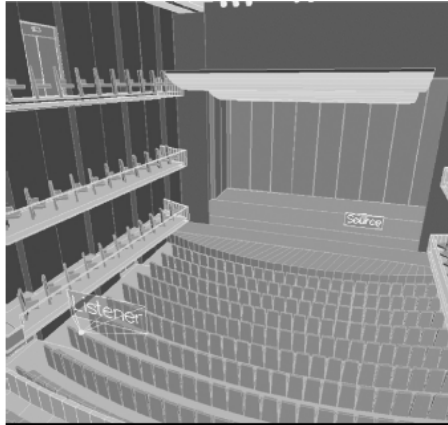


a) Mô hình tổng thể khu đô thị mới Nghĩa Tân

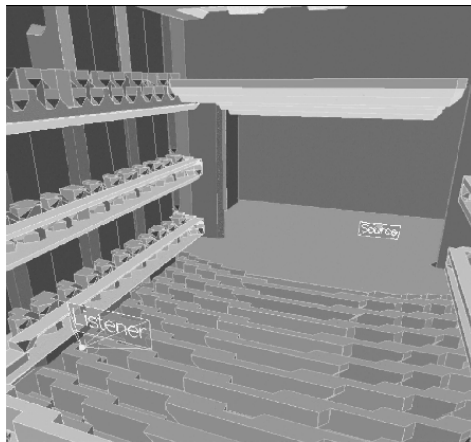
b) Một phần khu đô thị

Hình 10. Mô hình khu đô thị mới Nghĩa Tân, Cầu Giấy, Hà Nội

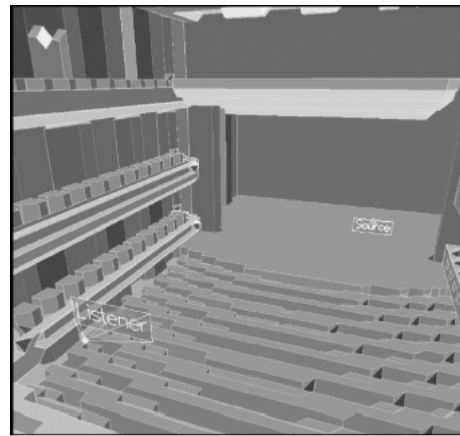
Hình 11 dưới đây minh họa cho việc rút gọn số lượng bề mặt biểu diễn của thuật toán HS. Hình 11.a là mô hình tổng thể của một hội trường với số lượng bề mặt là 73.400 bề mặt. Áp dụng thuật toán HS với ngưỡng góc giữa 2 vectơ pháp tuyến $\leq 5^\circ$, chúng tôi đã rút gọn được xuống số lượng bề mặt khác nhau (kết quả hình 11.c, 11.e), còn thuật toán EC [10, 18] thực hiện rút gọn bằng cách chụm cạnh (kết quả hình 11.b, 11.d). So sánh kết quả rút gọn của HS với kết quả rút gọn của EC thì thấy HS cho kết quả mô hình có chất lượng tốt hơn với bề mặt mịn hơn, còn EC cho kết quả mô hình tồi hơn với bề mặt thô hơn.



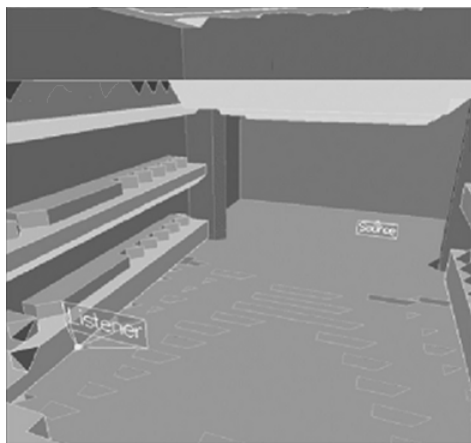
a) Mô hình gốc với 73.400 bề mặt



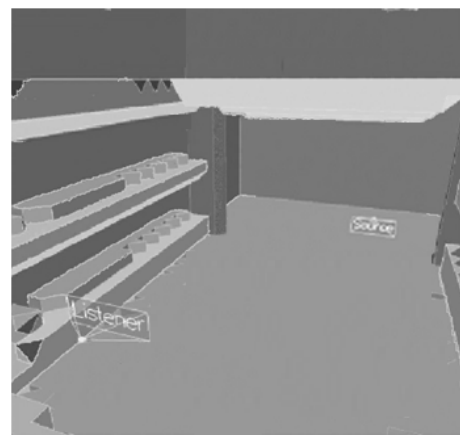
b) Mô hình rút gọn còn 45.000 bề mặt đối với EC



c) Mô hình rút gọn còn 45.000 bề mặt đối với HS



d) Mô hình rút gọn còn 10.000 bề mặt đối với EC



e) Mô hình rút gọn còn 10.000 bề mặt đối với HS

Hình 11. Mô hình một công trình kiến trúc với các mức độ rút gọn khác nhau của thuật toán EC và thuật toán đề xuất HS

4. KẾT LUẬN

Hai khâu quan trọng trong việc xây dựng một hệ thống thực tại ảo (virtual reality system) là xây dựng mô hình (modelling) và điều khiển mô hình (rendering). Trong xây dựng mô hình, ngoài vấn đề chất lượng còn đòi hỏi yêu cầu về tính đơn giản nhằm giảm thiểu không gian lưu trữ, phục vụ cho quá trình điều khiển sau này.

Bài báo đề xuất một kĩ thuật rút gọn số lượng bề mặt biểu diễn mô hình 3D nhằm giảm thiểu không gian lưu trữ. Kĩ thuật đề xuất được thực hiện nhờ việc tính toán góc giữa các véctơ pháp tuyến bề mặt kề nhau, trên cơ sở mở rộng thuật toán Angles trong không gian 2D. Qua thực nghiệm kĩ thuật đề xuất tỏ ra phù hợp với các bề mặt gần nhau là tương đối bằng phẳng.

Lời cảm ơn. Các tác giả xin chân thành cảm ơn sự hỗ trợ của đề tài KC.01.17/06-10 "Nghiên cứu phát triển và ứng dụng các giải pháp công nghệ thông tin hiện đại tái tạo ảnh mặt người 3 chiều từ dữ liệu hình thái xương sọ phục vụ điều tra hình sự và an sinh xã hội"

TÀI LIỆU THAM KHẢO

1. Đỗ Năng Toàn, Nguyễn Văn Huân - A hair material simulation approach in Virtual reality and application, Proceedings of The 7th Asian Control Conference, Hong Kong, China, 2009, pp. 541-546. <http://iee.org>
2. Đỗ Năng Toàn, Nguyễn Văn Huân - Morphing and Repairing a 3D Scalp Geometry in Building Hair Models and Simulation, Proceedings of 2nd International Congress on Image and Signal Processing, Tianjin, China, 2009, pp.705-710. <http://iee.org>
3. Eric Landreneau and Scott Schaefer - Simplification of Articulated Meshes, Journal compilation of The Eurographics Association and Blackwell Publishing Ltd. Eurographics **28** (2009).
4. Hoppe, S. Krömker - Adaptive Meshing And Detail-Reduction Of 3d-Point Clouds From Laser Scans, Proceedings of the ISPRS Workshop '3D-ARCH, 2009.
5. Mridulika - Surface Model Simplification Method Based On Quadric Error Metrics, Guru Gobind Singh Indraprastha University, Delhi 06, Proceedings of 2nd National Conference on Challenges & Opportunities in Information Technology (COIT-2008), RIMT-IET, Mandi Gobindgarh, March 29, 2008.
6. Frédéric Payan, Stefanie Hahmann and Georges-Pierre Bonneau - Deforming surface simplification based on dynamic geometry sampling, Proceedings of International Conference on Shape Modeling and Applications, SMI'07, Lyon – France, 2007.
7. Zhang S., Wu E. - Deforming surface simplification based on feature preservation, In ICEC, 2007, pp. 139-149.
8. Connie Phong - An Introduction to 3D Geometry Compression and Surface Simplification, CSC/Math 870, 2007.
9. Jia-xin CHEN, Hai-he HU - One Mesh Model Simplification Method Based on Shape Transform of Triangles, IEEE Computer Society, 2006.
<http://ieeexplore.ieee.org/iel5/4089190/4089191/04089302.pdf>
10. S. Siltanen, T. Lokki, and L. Savioja - Geometry Reduction In Room Acoustics Modeling, Proceedings of the Institute of Acoustics, Vol. 28, Pt.2, 2006.

11. Đỗ Năng Toàn - Ứng dụng công nghệ thực tại ảo trong bảo tàng các di sản, đề tài cấp Viện Khoa học và Công nghệ Việt Nam, 2004-2006, 2006.
12. Röttger S. Terrain LOD Implementations-libMini, 2004. <http://www.vterrain.org/LOD/Implementations/>. [capture on 26/03/04]
13. Đỗ Năng Toàn, Ngô Quốc Tạo - Đơn giản hoá đường cong có hiệu chỉnh trong quá trình vectơ hoá, chuyên san Các công trình nghiên cứu và triển khai Công nghệ thông tin và viễn thông, Tạp chí Bưu chính viễn thông (7) (2002) 75-81.
14. Zhao, Youbing, Ji Zhou, Jiaoying Shi, and Zhigeng Pan - A Fast Algorithm for Large Scale Terrain Walkthrough, CAD/Graphics, 2001.
15. Levoy M., Pulli K., Curless B., Rusinkiewicz S., Koller D., Pereira L. - The digital michelangelo project: 3d scanning of large statues. In SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques, New York, NY, USA, 2000, pp. 131-144.
16. Mapscan, <http://www.un.org/Depts/unsd/softproj/software/mapscan.htm>
<http://www.ons.dz/unfpa/software/mapscan.htm>;
<http://www.aig.asn.au/utilities.htm>
17. Hamann Bernd - A Data Reduction Scheme for Triangulated Surfaces, Computer Aided Geometric Design **11** (2) (1994) 197-214.
18. Schroeder, William J., Jonathan A. Zarge, and William E. Lorensen - Decimation of Triangle Meshes, Computer Graphics, SIGGRAPH '92 Proceedings **26** (2) (1992) 65-70.
19. Turk Greg - Re-Tiling Polygonal Surfaces, Computer Graphics, SIGGRAPH '92 Proceedings **26** (2) (1992) 55-64.

SUMMARY

AN ALGORITHM TO REDUCE FACES IN 3D MODELS

This paper deals with a technique to reduce faces in 3D models based on Angles algorithm in 2D, by calculating angle between two normal faces, that have the same edge. The given technique seems be effective for approximately flat surfaces.

Địa chỉ:

Nhận bài ngày 12 tháng 8 năm 2008

Đỗ Năng Toàn,
 Viện Công nghệ thông tin, VAST.
 Nguyễn Văn Huân,
 Khoa Công nghệ thông tin,
 Trường Đại học Thái Nguyên.