Tạp chí Khoa học ĐHQGHN, Khoa học Tự nhiên và Công nghệ, Tập 29, Số 2 (2013) 31-37

Một cải tiến cho kỹ thuật giấu LSB trên dữ liệu audio

Nguyễn Xuân Huy[1], Huỳnh Bá Diệu [2,*], Võ Thị Thanh[2]

[1]*Viện Công nghệ thông tin, Viện Khoa học Công nghệ Việt Nam, 18 Hoàng Quốc Việt, Hà Nội*
[2]*Khoa Đào tạo Quốc tế, Trường Đại học Duy Tân, 182 Nguyễn Văn Linh, Đà Nẵng*

Nhận ngày 01 tháng 3 năm 2013
Chỉnh sửa ngày 08 tháng 4 năm 2013; chấp nhận đăng ngày 07 tháng 5 năm 2013

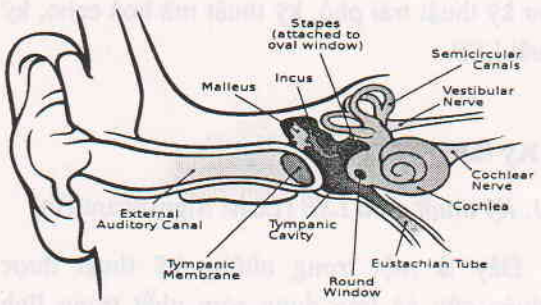
Tóm tắt. Trong bài báo này chúng tôi trình bày một cải tiến cho kỹ thuật giấu tin trong audio dựa vào LSB. Việc cải tiến được thực hiện bằng cách chọn các bit và các đoạn dữ liệu khác nhau để giấu tin, nhờ vào bộ sinh số ngẫu nhiên. Bên cạnh đó chúng tôi cũng đề xuất các cách điều chỉnh bit để làm giảm độ sai khác dữ liệu gốc và dữ liệu chứa tin giấu.

Từ khóa: Bộ sinh số ngẫu nhiên, kỹ thuật LSB.

1. Giới thiệu

Để bảo vệ thông tin ta có thể thực hiện theo cách mã hoá hoặc giấu tin. Mã hoá sẽ dùng khoá (bí mật hoặc công khai) biến bản rõ thành bản mã. Để lấy lại tin ta dùng khoá để giải mã. Giấu thông tin sẽ dùng đối tượng chứa để giấu thông tin cần giấu. Việc giấu tin vào đối tượng chứa có thể bảo vệ cho tin giấu hoặc có thể bảo vệ cho đối tượng chứa. Có nhiều kiểu dữ liệu chứa có thể chọn để giấu tin như ảnh, âm thanh, video, text [1]. Ngay từ những ngày đầu phát triển, các thuật toán giấu tin hầu hết chỉ tập trung nghiên cứu nhiều lĩnh vực ảnh và video. Thuật toán giấu tin trên âm thanh chỉ thực sự phát triển trong khoảng hơn thập niên trở lại đây, do đó mức độ hiệu quả của các phương pháp giấu tin trên âm thanh số khó có thể so sánh được với các kết quả đã đạt được như trên ảnh và video. Tuy nhiên, với tốc độ phát triển của hướng nghiên cứu này ngày càng được chú ý [2].

Đặc điểm chung của hướng nghiên cứu "Giấu tin trên âm thanh số" là tập trung khai thác khả năng cảm nhận của hệ thính giác người.



Hình 1. Hệ thống thính giác con người.

Theo các nghiên cứu về sinh học, hệ thính giác người nhạy cảm hơn nhiều so với thị giác. Vì vậy các kỹ thuật giấu tin trong dữ liệu audio gặp khó khăn hơn[3,4].

* Tác giả liên hệ. ĐT: 84-914146868.
E-mail: dieuhb@gmail.com

Các kỹ thuật giấu tin trên âm thanh có thể được chia thành hai nhóm chính. Đó là nhóm có sử dụng tín hiệu gốc trong quá trình rút trích thông tin và nhóm không cần đến tín hiệu gốc trong quá trình rút trích thông tin.

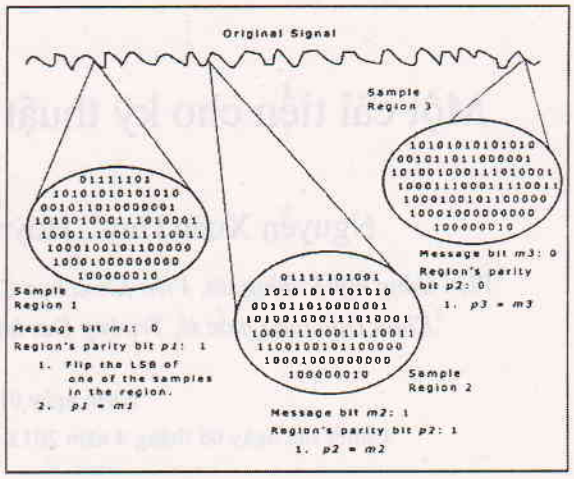
Nhóm sử dụng tín hiệu gốc hay nhóm giao thoa với tín hiệu gốc cần đến thông tin gốc khi rút trích thông tin. Trong các ứng dụng thực tế, nhóm phương pháp này tỏ ra không hiệu quả vì cần gấp đôi bộ nhớ để lưu trữ cùng một thông tin, và đôi khi ta không thể có được tín hiệu gốc sử dụng cho quá trình rút trích thông tin.Tuy nhiên, trong một số trường hợp như trong việc chứng thực bản quyền, nhóm phương pháp này lại tỏ ra rất có hiệu quả, thậm chí là chúng rất cần thiết. Một số phương pháp thuộc nhóm này bao gồm: mã hoá pha và điều chế pha.

Nhóm phương pháp không cần đến tín hiệu gốc trong quá trình rút trích thông tin chỉ cần file tín hiệu âm thanh có chứa tin giấu và có thể là khoá để rút trích thông tin mật. Nhóm phương pháp này được chia thành bốn nhóm nhỏ: nhóm các phương pháp trải phổ, nhóm các phương pháp tập đôi, nhóm các phương pháp bản sao và nhóm các phương pháp tự đánh dấu. Một số phương pháp giấu tin thuộc nhóm này như kỹ thuật trải phổ, kỹ thuật mã hoá echo, kỹ thuật LSB…

2. Kỹ thuật giấu LSB

2.1. Kỹ thuật giấu LSB (Least Significant Bit)

Đây là một trong những kỹ thuật được nghiên cứu và ứng dụng sớm nhất trong lĩnh vực giấu dữ liệu trên âm thanh, cũng như trên các định dạng dữ liệu khác. Bộ mã hoá sử dụng một tập các mẫu tín hiệu gốc x được chọn ra theo một khoá mật nào đó. Sau đó thực hiện thao tác thay thế trên các bit ít quan trọng nhất để biểu diễn thông tin giấu.



Hình 2. Chọn mẫu dữ liệu để giấu tin.

Các bit được chọn để giấu tin thường là từ bit 1 đến bit 3. Việc thay đổi giá trị các bit này làm thay đổi ít giá trị của mẫu dữ liệu gốc nên ít ảnh hưởng đến hệ thống thính giác, gây nghi ngờ cho những người thám tin.

Để giấu 1 bit thông tin, ta sẽ thay bit được chọn bằng bit thông tin đem giấu. Ví dụ ta cần giấu bit 1 vào mẫu dữ liệu 8 bit sau:

0	1	0	1	1	1	1	0

Sau khi giấu bít 1 sẽ như sau:

0	1	0	1	1	1	1	1

Hình 3. Minh hoạ kỹ thuật giấu LSB.

Ưu điểm của phương pháp này là khả năng lưu trữ lớn, có thể chứa được nhiều thông tin mật. Khuyết điểm của nó là dễ bị tấn công, có tính bền vững thấp. Tuy nhiên, do không phải thực hiện nhiều phép toán phức tạp nên phương pháp này có thời gian thực hiện rất nhanh, có thể đáp ứng về mặt thời gian thực. Đây có thể được xem là thuật toán ẩn dữ liệu cơ bản nhất.

Có thể tăng thêm dữ liệu giấu bằng cách dùng hai bít LSB. Tuy nhiên cách này cũng làm tăng nhiễu trên đối tượng chứa dẫn đến đối

phương dễ phát hiện và thực hiện các tấn công. Vì vậy dữ liệu chứa cần phải được chọn trước khi giấu sử dụng phương pháp mã hóa LSB.

2.2. Tấn công đối với hệ giấu tin mật dùng LSB

Giấu tin trong âm thanh chịu một số tấn công như lấy lại mẫu, lọc thông, thêm nhiễu, biến đổi D/A A/D.... Đối với phương pháp giấu tin dùng LSB, hay được sử dụng để giấu tin mật thì tấn công được quan tâm đó là lấy lại tin mật. Bằng cách chọn các bit cuối của các mẫu dữ liệu, người thám tin ghép lại để lấy tin mật hay một phần của tin mật[5].

Để tăng độ an toàn cho kỹ thuật này, ta sẽ chọn các mẫu không liên tục và chèn vào các bit khác nhau chứ không cố định bit cuối. Cách này sẽ gây khó khăn cho người thám tin trong quá trình dò tin. Đây là cơ sở để thiết lập nên cải tiến cho phương pháp LSB.

3. Kỹ thuật đề xuất và các kết quả đạt được

Cải tiến đề xuất dưới đây sẽ dùng chuỗi giả ngẫu nhiên để xác định vị trí các mẫu dữ liệu chọn và vị trí bit sẽ dùng để giấu tin. Bộ sinh số ngẫu nhiên được chọn là bộ sinh số đồng dư cải tiến. Ký thuật điều chỉnh bit để giảm thiểu độ lệch so với dữ liệu gốc cũng được đề cập.

3.1. Bộ sinh đồng dư (congruential generator)

Trong giấu tin, ta thường cần đến các chuỗi ngẫu nhiên để chọn mẫu dữ liệu giấu hoặc điều chỉnh giá trị các mẫu theo chuỗi ngẫu nhiên này. Có nhiều kỹ thuật để sinh ra chuỗi giả ngẫu nhiên như kỹ thuật trung bình bình phương, bộ sinh fibonacci, bộ sinh đồng dư [6, p10]. Bài báo này dùng bộ sinh đồng dư cải tiến để sinh chuỗi ngẫu nhiên.

Bộ sinh đồng dư có dạng:

$$x_i = (ax_{i-1} + b) \bmod N \quad (1)$$

Trong đó a và b là hai số nguyên cho trước, trị x_0 ban đầu được gọi là "hạt giống" (seed) và số nguyên N là số xấp xỉ (hoặc bằng) với số nguyên lớn nhất trên máy tính. Nếu trị số x_0 được gán cố định thì việc sinh các dãy là giống nhau trong các lần chạy. Để khởi tạo x_0 sao cho các dãy sinh ra khác nhau, người ta thường lấy giờ được lưu trong CMOS để làm số mồi x_0. Chất lượng của bộ sinh ngẫu nhiên phụ thuộc vào việc chọn lựa giá trị a và b. Và trong bất kỳ trường hợp nào, số lần sử dụng bộ sinh này không thể vượt quá M (vì nếu không nó sẽ phát sinh những số lặp lại).

Bộ sinh đồng dư cải tiến dùng để phát sinh một số nguyên giả ngẫu nhiên trong miền [0..n-1]. Đây là một bộ sinh ngẫu nhiên khá thông dụng, đặc biệt là trong các ứng dụng mô phỏng tự nhiên như mô phỏng vũ trụ [7]. Bộ sinh số này sẽ dùng 2a thay cho b trong công thức (1) với a được chọn thường là số nguyên tố lớn. Công thức của bộ sinh này như sau :

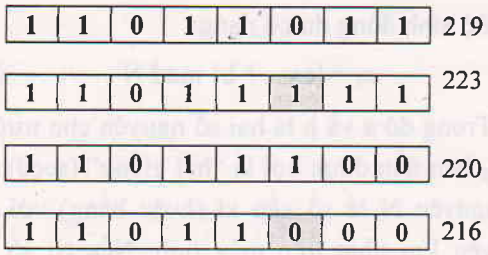
$$x_i = (ax_{i-1} + 2a) \bmod n \quad (2)$$

Việc chọn số mồi x_0 tương tự như bộ sinh số đồng dư.

Đối với kỹ thuật đề xuất, sẽ dùng khoá là cặp số (x,y) để sinh chuỗi ngẫu nhiên. Giá trị x sẽ dùng làm số mồi và y thay cho a trong công thức (2).

3.2. Điều chỉnh độ lệch bit

Đối với phương pháp LSB, ta có thay đổi từ bít 1 đến bít 3. Trong trường hợp thay đổi bít 3, độ lệch giữa mẫu gốc và mẫu sau khi thay đổi là 4 như hình 4.



Hình 4. Độ lệch giá trị khi điều chỉnh bit 3.

Để giảm độ lệch này, ta tiến hành như sau:

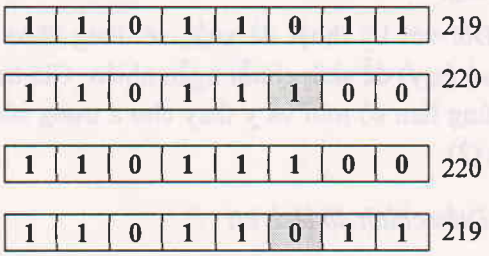
Đối với đổi bit 3 từ 0 thành 1, ta sẽ đổi các bít 2 và 1 thành 0 và đối với đổi bít 3 từ 1 thành 0 ta sẽ đổi các bit 2 và 1 thành 1.

Tương tự đối đối với đổi bit 2 từ 0 thành 1, ta sẽ đổi bit 1 thành 0 và đối với đổi bít 2 từ 1 thành 0 ta sẽ đổi bit 1 thành 1.

Thủ tục điều chỉnh bit thứ I chứa giá trị k như sau:

```
PROC DIEUCHINH(i,k)
    SET(i,k);
    If(i>1) SET(1,1-k);
    If(i>2) SET(2,1-k);
END
```

Nếu tiến hành điều chỉnh theo phương pháp đề xuất thì độ giá trị chỉ còn 1.



Hình 5. Độ lệch giá trị khi điều chỉnh bit 3.

3.3. Quá trình giấu tin

Đầu vào:

File âm thanh A, chuỗi bit M có độ dài L, khoá K gồm hai số x,y.

Đầu ra:

File âm thanh A' có chứa chuỗi bit M

Bước 1: Chuẩn bị

Dựa vào khoá (x,y) sinh ra mảng S[] cho biết mẫu dữ liệu được chọn để giấu tin và mảng V[] cho biết vị trí bit sẽ được giấu trong mỗi mẫu.

Bước 2: Giấu tin

Đọc file dữ liệu âm thanh, dựa vào mảng S[] để chọn mẫu cần giấu.

Dựa vào mảng V[] để biết vị trí bit giấu.

Nếu bit thứ V_i của mẫu chọn thứ i khác với M_i thì thực hiện DIEUCHINH(V_i, M_i)

Sau khi giấu xong, ta dùng các ký hiệu đánh giấu kết thúc tin giấu nhằm phục vụ cho quá trình giải tin.

Việc sinh mảng S[] và V[] được thực hiện như sau:

Dựa vào khoá K(x,y), dùng công thức (2) của bộ sinh số đồng dư cải tiến ta sinh được chuỗi giả ngẫu nhiên SRN[].

Tìm S[] dùng công thức:

$$S[i] = (SRN[i] \bmod 9) \bmod 4 + 1$$

Tìm V[] dùng công thức:

$$S[i] = (SRN[i] \bmod 3) + 1$$

Ví dụ, khoá K là K(7,9137), chọn N=10000, theo bộ sinh số đồng dư, ta cần sinh chuỗi số ngẫu nhiên gồm 16 số như sau:

SRN={82233, 81195, 96989, 6767, 48353, 19635, 23269, 27127, 77673, 16475, 50349, 57087, 22193, 95715, 66229, 52647}

S ={1, 3, 2, 1, 2, 3, 1, 2, 4, 2, 4, 1, 1, 1, 4, 3}

V={1, 1, 3, 3, 3, 1, 2, 2, 1, 3, 1, 1, 3, 1, 2, 1}

Mảng S sẽ cho ta biết mẫu được chọn giấu là 1, 4, 6, 7, 9, 12, 13, 15, 19, 21, 25, 26, 27, 28, 32, 35. Điều này có nghĩa là S[i] cho biết mẫu kế tiếp sẽ được chọn giấu tin so với mẫu đã chọn để giấu bít thứ i-1.

Mảng V sẽ cho ta biết vị trí bit để giấu bít thứ i trong mẫu được chọn thứ I là V[i].

Để giấu được chuỗi bit độ dài L vào trong file dữ liệu A thì file A phải có ít nhất 4.L mẫu.

3.4. Quá trình giải tin

Đầu vào:

File âm thanh A' chứa tin giấu, khoá K gồm hai số (x,y).

Đầu ra:

Chuỗi bit M được giấu.

Bước 1: Chuẩn bị

Dựa vào khoá (x,y) sinh ra mảng S[] cho biết mẫu dữ liệu được chọn để giấu tin và mảng V[] cho biết vị trí bit sẽ được giấu.

Bước 2: Giải tin

Đọc file dữ liệu âm thanh A', dựa vào mảng S[] để chọn mẫu có chứa bit thông tin giấu, dựa vào mảng V[] để lấy bít thứ i của mẫu tin giấu.

3.5. So sánh với phương pháp LSB nguyên thuỷ

3.5.1. Chi phí thời gian giấu tin và giải tin

Phương pháp LSB nguyên thuỷ sẽ chọn 1 vị trí bit cố định và các đoạn tuần tự để giấu tin.

So với kỹ thuật LSB nguyên thuỷ, phương pháp này cần phải tốn thời gian cho việc sinh chuỗi giả ngẫu nhiên, chuỗi S[] và chuỗi V[].

Bên cạnh đó ta phải tốn chi phí thời gian cho việc điều chỉnh bit để giảm độ chênh lệch giá trị trước và sau khi giấu.

Trong quá trình giải tin chúng ta cũng tốn chi phí cho sinh chuỗi giả ngẫu nhiên để biết vị trí mẫu và bit chứa tin giấu.

3.5.2. Tỉ lệ dữ liệu giấu

Phương pháp LSB nguyên thuỷ thực hiện giấu bit tin trên 1 mẫu dữ liệu. Phương pháp này dùng mảng độ lệch để xác định mẫu giấu. Trường hợp xấu nhất, các mẫu này đều có độ lệch 4 nên cần ít nhất là 4*L mẫu để có thể giấu chuỗi bit độ dài L. Vì vậy so với phương pháp LSB nguyên thuỷ này có tỉ lệ dữ liệu giấu thấp hơn, chỉ bằng ¼ .

3.5.3. Độ mật của kỹ thuật

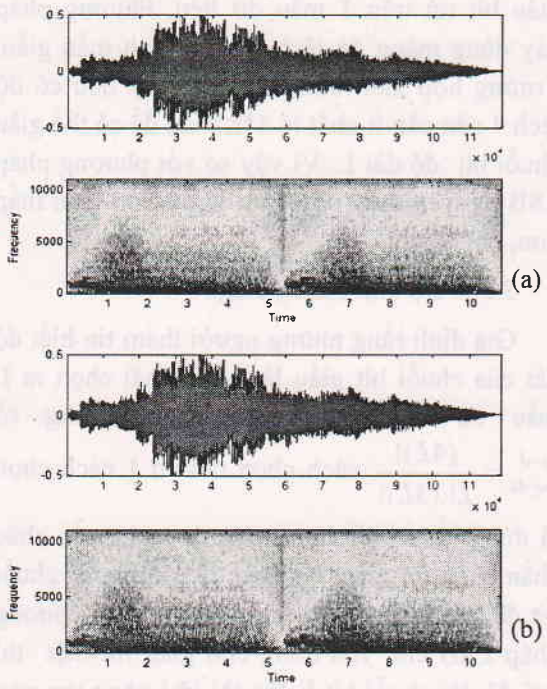
Giả định rằng những người thám tin biết độ dài của chuỗi bít giấu là L, họ phải chọn ra L mẫu dữ liệu trong 4*L mẫu. Trong số $C_{4L}^{L} = \frac{(4L)!}{L!.(3L)!}$ cách chọn chỉ có 1 cách chọn là đúng. Sau khi chọn xong được L mẫu chắc chắn chứa tin giấu, họ phải dò 1 trong 3^n chuỗi bit để xác định chuỗi tin giấu. Đối với phương pháp LSB chủ yếu dùng cho giấu tin mật thì với độ dài chuỗi bit L lớn thì khả năng tìm này gần như không thể.

Trong trường hợp những người thám tin có được file âm thanh gốc A và file chứa tin giấu A' thì họ không thể dùng phương pháp trừ để lấy được chuỗi tin giấu như phương pháp LSB nguyên thuỷ vì trong phương pháp cải tiến chúng ta có thể thay đổi nhiều hơn 1 bit chỉ để giấu 1 bit tin.

Bên cạnh đó, việc chọn bộ sinh số ngẫu nhiên với các tham số khác nhau cũng gây khó khăn cho những người thám tin khi họ muốn lấy tin giấu.

Bù lại chi phí thời gian cho quá trình giấu tin, giải tin, chúng ta có được một kỹ thuật an toàn hơn so với cách dùng LSB nguyên thuỷ.

Các thử nghiệm dưới đây dùng file WindowsLogOn.wav và chuỗi bit có độ dài 1024. Thử nghiệm cho thấy rằng ti lệ sai khác giữa file chứa tin giấu và không chứa tin giấu không thể phân biệt khi nghe.



Hình 6. Phổ biên độ và phổ pha của file chứa trước (a) và sau khi giấu chuỗi bit (b).

4. Kết luận

Với việc chọn ngẫu nhiên các mẫu và các bít dữ liệu để giấu tin sẽ làm cho người thám tin kích thông tin. Kỹ thuật điều chỉnh bit cũng là giảm độ sai khác giữa file chứa thông tin và file gốc. Với việc cải tiến này chúng ta có thể sử dụng để giấu tin mật, phục vụ cho việc bảo vệ thông tin. Do đặc tính chung của phương pháp LSB là không bền vững nên phương pháp cải tiến này chỉ có thể chịu tấn công lấy lại tin chứ không chịu được các tấn công khác như thay đổi tin, thêm nhiễu v.v ..Để tránh trường hợp nhận tin bị sai, có thể sử dụng các mã hỗ trợ xác thực tin giấu[3].

Tài liệu tham khảo

[1] Maha Bellaaj, Comparative analysis of audio watermarking technique in MDCT domain with other references in spectral domain ,Intenational Multi-Conference on Systems, Signals and Devices, 2012.

[2] Fatiha Djebbar, Beghdad Ayady, Habib Hamamzand Karim Abed-Meraim, A view on latest audio steganography techniques, 2011 International Conference on Innovations in Information Technology, 2011.

[3] Nguyễn Xuân Huy, Huỳnh Bá Diệu, Nghiên cứu kỹ thuật giấu tin trong audio hỗ trợ xác thực, Tạp chí Khoa học ĐHQGHN, Khoa học Tự nhiên và Công nghệ 25, 2008.

[4] Sridevi, A damodaram, Avl.narasimham, Efficient method of audio steganography by modified lsb algorithm and strong encryption key with enhanced security, Journal of Theoretical and Applied Information Technology, 2005.

[5] Min Wu, Scott A, Craver, Edward W Felten, Bede Liu, Analysis Of Attacks On Sdmi Audiowatermarks, Princeton University, 2003.

[6] D. E. Knuth. *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*, Addison-Wesley, 1997.

[7] http://en.wikipedia.org/wiki/Linear_congruential_generator.

An improved technique for hiding data in audio using LSB

Nguyễn Xuân Huy[1], Huỳnh Bá Diệu[2], Võ Thị Thanh[2]

[1]*Institute of Infomation Technology ,Vietnamese Academy of Science and Technology , 18 Hoàng Quốc Việt Str., Cầu Giấy Dist., Hanoi, Vietnam*

[2]*Internation School, DuyTan University, 182 Nguyễn Văn Linh Str., Da Nang, Vietnam*

Abstract. This paper presents an improved technique to hide information in audio based on the LSB. The improvement is done by using the random number generator to select the samples data and bit postion for hiding. We also suggest a solution for modifying the set of bits to reduce the difference between host file and file contains hiding data.

Keywords: random number generator, Least Significant Bit.