LỜI CẢM ƠN

Trước hết, em xin gửi lời cảm ơn sâu sắc tới TS. Hồ Văn Canh, người đã gợi mở và hướng dẫn em đi vào tìm hiểu đề tài giấu tin mật và thuỷ vân ảnh. Người đã hết lòng giúp đỡ, tạo điều kiện cho em hoàn thành khoá luận này.

Em xin cảm ơn các thầy, cô trong trường Đại học Dân lập Hải Phòng đã dậy dỗ chúng em, giúp đỡ động viên chúng em từ những ngày đầu chập chững bước chân vào cánh cổng trường Đại học. Thầy cô đã tạo cho chúng em môi trường học tập, những điều kiện thuận lợi cho chúng em được học tập tốt, trang bị cho chúng em những kiến thức quý báu giúp chúng em có thể vững bước trong tương lai.

Cảm ơn các bạn đã giúp đỡ, cùng nghiên cứu và chia sẻ trong suốt 4 năm học Đại học.

Hà Nội, 2009

Lê Thị Hải Yến

Mục lục

LỜI MỞ ĐẦU

Các kỹ thuật đảm bảo an toàn thông tin cho thông tin liên lạc số được chia thành 3 loại (Categories). Đó là mật mã (Cryptography), giấu tin mật (Steganography) và thủy vân số (watermarking). Mỗi loại có những ứng dụng và mục tiêu khác nhau nhưng đều đảm bảo an toàn cho việc truyền tin mật trên kênh không an toàn.

Các kỹ thuật Cryptography và steganography nói chung được dùng để truyền những thông tin nhạy cảm (confidential infomation) giữa hai hay nhiều thực thể trong cùng một nhóm với nhau. Tuy nhiên giữa chúng có những sự khác nhau.

Cryptography sử dụng những phép biến đổi toán học để mã hóa bản thông điệp, biến một thông điệp đọc được có nghĩa thành một dãy giả ngẫu nhiên, mà người ta gọi là bản mã, để truyền trên mạng công cộng đến người nhận có chủ đích. Đó là khi hai người chẳng hạn là Alice và Bob liên lạc mật với nhau thì mặc dù Wendy không đọc được nội dung thông tin nhưng Wendy rõ ràng là biết được giữa Alice và Bob đang có ý đồ "đen tối" nào đó.

Ngược lại, với steganography thì Wendy không thể biết được giữa Alice và Bob đang có sự liên lạc truyền thông tin mật cho nhau. Để đảm bảo được điều này, Alice và Bob sử dụng một vật trung gian số ở đây là audio, video, hoặc images…

Trong phạm vi nghiên cứu, ta giả thiết vật trung gian ở đây là ảnh số(ảnh đen trắng, ảnh màu hoặc ảnh đa cấp xám). Người ta đã lợi dụng độ "dư thừa" trong ảnh để nhúng (embedding) các bít thông điệp mật vào đó, do sự "dư thừa" này có thay đổi chút ít sẽ không làm thay đổi đến ảnh. Để đảm bảo bí mật tuyệt đối người ta sẽ mã hóa bức thông điệp trước khi thực hiện nhúng chúng vào ảnh.

Còn thủy vân số(watermarking) về nguyên lý tương tự như steganography nhưng có khác nhau về mục đích ứng dụng. Mục tiêu của watermarking là những thông tin được nhúng trong ảnh phải đảm bảo sao cho watermark không thể bị dịch chuyển mà không phá hủy chính ảnh mang tin đó. Watermaking thường được ứng dụng trong các lĩnh vực như bảo vệ bản quyền.

Hiện nay ngoài mật mã học, steganography và watermaking đang phát triển rất mạnh. Trên thế giới cho đến nay đã có nhiều công trình nghiên cứu vấn đề này và đang trở thành một hướng đi mới trong lĩnh vực An toàn thông tin, chống giả mạo. Ở trong nước thì đây là một lĩnh vực mới được nghiên cứu trong những năm gần đây của thế kỷ 21, và cũng mới đươc quan tâm chủ yếu ở một số viện nghiên cứu khoa học, và một số trường Đại học lớn như viện công nghệ thông tin, trường Đại Học Công nghệ thuộc Đại Học Quốc Gia Hà Nội và TP. HCM, Đại Học Đà Nẵng.

Tin rằng lĩnh vực nghiên cứu này có nhiều hứa hẹn trong tương lai gần và dần trở thành một hướng đi mới trong lĩnh vực Bảo Đảm An toàn thông tin rất có hiệu quả. Chính vì vậy, em đã chọn đề tài : « ìm hiểu kỹ thuật giấu tin mật và thuỷ vân ảnh » làm đồ án tốt nghiệp của mình. Do đây là hướng mới của an toàn thông tin với lại do trình độ của em có phần hạn chế nên kết quả của nó chắc còn nhiều thiếu sót, em kính mong được sự góp ý, chỉ bảo của thầy (cô).

CHƯƠNG 1: NHỮNG KHÁI NIỆM CƠ BẢN

.1 Mở đầu

Giấu tin mật là một khoa học về liên lạc "không nhìn thấy được". Nó khác với khoa học về mật mã là ở chỗ: Trong khoa học mật mã người ta tìm cách biến đổi bản thông điệp có ý nghĩa thành một dãy giả ngẫu nhiên để liên lạc với nhau trên mạng công cộng mà người ngoài cuộc (người không được phép chia sẻ thông tin trong thông điệp đó) có thể thu được sự hiện hữu của dãy ngẫu nhiên đó nhưng khó lòng chuyển dãy đó thành bản thông điệp ban đầu nếu không có "khóa" trong tay. Trong lúc đó kỹ thuật giấu tin mật(steganography) lại tìm cách ẩn giấu thông điệp đó vào trong một phương tiện số khác (như audio, video, images…) mà người ngoài cuộc khó có thể phát hiện được sự hiện hữu của thông điệp trong phương tiện số đó, mặc dù người ta có thể có phương tiện đó trong tay. Phương tiện được dùng để giấu tin trong đó được gọi là phương tiện gốc (Cover-objects). Còn phương tiện gốc đó đã được chứa thông tin cần giấu trong đó được gọi là phương tiện mang tin (Stego-Objects).

Việc giấu thông tin mật có ý nghĩa quan trọng đối với an ninh, thông tin có tính chất Quốc gia. Hiện nay bọn khủng bố Quốc tế cũng như các cơ quan tình báo các nước đã và đang ứng dụng thành công kỹ thuật này để phục vụ mục tiêu của họ.

Một hướng phát triển của kỹ thuật này là Thủy vân số (Watermaking). Hướng nghiên cứu này phát triển rất nhanh, chủ yếu phục vụ cho kinh tế-xã hội (như để bảo vệ bản quyền…). Do mục tiêu của hai kỹ thuật này khác nhau nên yêu cầu của chúng cũng khác nhau.

Trong đề tài luận văn, cả hai kỹ thuật này (steganography và watermarking) đều được tập trung nghiên cứu tìm hiểu. Hiện nay cả hai

trường hợp steganography và watermarking đều phát triển rất mạnh trên thế giới. Tuy nhiên trong phạm vi đồ án tốt nghiệp, em chỉ tập trung tìm hiểu các kỹ thuật giấu tin trong ảnh tĩnh cho cả hai trường hợp là steganography và watermarking.

1.2 Những khái niệm cơ bản

1.2.1 Những quy ước.

Ảnh môi trường hay đôi khi còn gọi là ảnh gốc (cover image) là ảnh (đối tượng) chứa mang thông tin nhúng trong đó. Nó có thể là ảnh đen trắng, ảnh màu hoặc ảnh đa cấp xám. Trong nghiên cứu này ảnh môi trường sẽ được ký hiệu là C. Nếu có nhiều ảnh môi trường, chúng là sẽ ký hiệu là C_1, C_2…

Ảnh stego (stego image) là ảnh có chứa thông tin mật trong đó. Ta thường gọi là ảnh có chứa thông tin ẩn và được ký hiệu là S. Nếu có nhiều stego image thì ta ký hiệu là S_1, S_2…

Để tiện cho việc trình bày, ta gọi hai người liên lạc với nhau là Alice và Bob còn người thứ 3 Wendy không biết được sự hiện hữu của thông điệp trong ảnh mà Alice và Bob trao đổi với nhau.

1.2.2 Những tính chất cơ bản của steganography và watermarking

1.2.2.1 steganography

- Khả năng không thể nhận biết (impercetibility).
- Khả năng chứa được nhiều thông tin (capacity).
- Khả năng không thể dò-tìm.

Khả năng không thể nhận biết được, có nghĩa là với người quan sát bằng mắt thường không thể phát hiện được ảnh có chứa thông tin ẩn trong đó. Đây là một tính chất cực kỳ quan trọng đối với kỹ thuật steganography.

Khả năng chứa được nhiều thông tin cũng là một tính chất quan trọng đối với kỹ thuật steganography. Tính chất capacity có nghĩa là lượng thông tin cần nhúng càng nhiều càng tốt nhưng không được vi phạm tính chất khác của kỹ thuật steganography.

Cuối cùng tính chất không thể dò tìm được hiểu ở đây là khả năng chống lại việc xác định ảnh đó có hay không có thông tin ẩn bằng các kỹ thuật thống kê toán học thông thường.

Tính chất này cùng với tính chất "không thể nhận biết được" và độ dài thông điệp cần giấu đóng một vai trò quan trọng và cần thiết trong kỹ thuật steganography.

Ngoài ra, tốc độ giấu cũng được tính đến mặc dù nó không phải là tính chất cần có.

1.2.2.2 Watermarking.

Do yêu cầu bảo vệ bản quyền, xác thực… nên giấu tin thủy vân có yêu cầu khác với giấu tin bí mật. Yêu cầu đầu tiên là các dấu hiệu thủy vân phải đủ bền vững trước những tấn công vô tình hay cố ý gỡ bỏ nó. Thêm vào đó các dấu hiệu thủy vân phải có ảnh hưởng tối thiểu(về mặt cảm nhận) đối với các phương tiện chứa. Vậy các thông tin cần giấu sẽ càng nhỏ càng tốt.

Trọng tâm của khóa luận là nghiên cứu các kỹ thuật giấu tin bí mật, nhưng để có cái nhìn đầy đủ hơn về các lĩnh vực giấu tin, trong phần tổng quan này chúng tối giới thiệu sơ lược về thủy vân, một lĩnh vực hiện nay đang được nghiên cứu phát triển mạnh và có nhiều ứng dụng trong thực tế.

Phân biệt giấu thông tin mật và thủy vân có thể mô tả tóm lược trong bảng sau:

	Giấu thông tin mật	Thủy vân số
Mục tiêu	Tàng hình các phiên liên lạc để bảo mật thông tin Dùng trong các liên lạc xác định	Chủ yếu phục vụ cho mục đích bảo vệ bản quyền Chủ yếu dùng trong các hoạt động xuất bản
Cách thực hiện	Không làm thay đổi phương tiện chứa	Có thể thay đổi nhỏ về cảm nhận tới phương tiện chứa
Yêu cầu	Giấu được nhiều thông tin nhất Không cần quan tâm tới độ bền của phương tiện chứa Không thể quan sát được việc nhúng thông tin Không kiểm tra được nếu không có khóa thích hợp	Chỉ cần nhúng ít dữ liệu Dữ liệu nhúng cần phải mạnh Đảm bảo trước các phương pháp nén dữ liệu Dữ liệu nhúng có thể nhận thấy hay không nhận thấy Không kiểm tra được nếu không có khóa thích hợp

1.3 Một số ứng dụng và xu hướng phát triển

Che giấu thông tin nói chung có rất nhiều ứng dụng tùy theo từng hoàn cảnh cụ thể. Giấu thông tin bí mật góp phần "tàng hình" các phiên liên lạc, một sự bổ sung lý tưởng cho công tác bảo mật thông tin. Ngoài ra cũng với hình thức dùng vỏ bọc ngụy trang che giấu thông tin này, các hacker có thể thực hiện việc phát tán các vi rút, các Trojan vào các máy tính để phục vụ cho các yêu cầu của mình. Do tính chất dễ sao chép sửa đổi của các loại dữ liệu kỹ thuật số, các kỹ thuật che giấu thông tin còn được áp dụng trong việc bảo vệ bản quyền, chống lại các sao chép bất hợp pháp, các sửa đổi thay đổi làm sai lệch nội dung thông tin, đây là các ứng dụng chính và rất quan trọng của các kỹ thuật giấu thông tin thủy vân. Ngoài ra còn có rất nhiều các ứng dụng khác như tự động kiểm tra bản quyền theo các mã quy định, điều khiển sao chép…

CHƯƠNG 2: STEGANOGRAPHY SECURITY (MỨC ĐỘ AN TOÀN CỦA GIẤU TIN MẬT)

2.1 Khái quát chung

Để đánh giá một thuật toán giấu tin nào đó có đạt các yêu cầu đặt ra hay không chúng ta cần đưa ra độ đo (measure) chất lượng của thuật toán đó.

Ta ký hiệu phân bố xác suất của ảnh C là P_c và phân bố xác suất của ảnh stego S là tương ứng với một thuật toán được sử dụng nào đó là P_s. Khi đó khả năng phát hiện hệ stego dựa trên entropy giữa phân bố xác suất của ảnh môi trường C và ảnh stego S được "đo" dựa trên công thức:

$$D(P_c||P_s) = \int Pc \log \frac{Pc}{Ps} \qquad (1)$$

Từ phương trình này, chúng ta thấy rằng $D(P_c||P_s)$ tăng theo sự tăng của tỷ số $\frac{Pc}{Ps}$ và do đó, độ tin cậy của việc phát hiện cũng tăng. Vì vậy, kỹ thuật stego được gọi là an toàn tuyệt đối nếu $D(P_c||P_s)=0$ (tức $P_c=P_s$) và nó được gọi là ε -an toàn nếu $D(P_c||P_s) \leq \varepsilon$

Về lý thuyết, người ta đã chứng tỏ được rằng có tồn tại thuật toán an toàn tuyệt đối mặc dù chúng không xẩy ra trong thực hành.

Ở đây chúng ta giả thiết rằng cover image và stego image là những vectơ ngẫu nhiên, độc lập cùng phân bố (independent, identically distributed-iid).

Như vậy để kiểm tra sự khác nhau giữa ảnh gốc và ảnh có giấu tin tương ứng, người ta (Wendy) sẽ kiểm tra sự khác biệt giữa tỷ số $\frac{Pc}{Ps}$.

Trong quá trình kiểm tra này, Wendy sẽ mắc phải hai sai lầm loại 1 (type – I error) và sai lầm loại 2 (type –II error) với xác xuất lần lượt là α và β ($0<\alpha, \beta<1$).

Sai lầm loại một là sai lầm sảy ra khi giả thiết là đúng nhưng anh ta lại bác bỏ nó. Còn sai lầm loại hai là sai lầm sảy ra do chấp nhận giả thiết sai(tức giả thiết là sai nhưng Wendy lại chấp nhận nó).

Chúng ta không thể đồng thời cực tiểu hóa cả hai sai lầm. Thông thường người ta cho cố định xác suất sai lầm loại một và xây dựng bài toán làm cực tiểu hóa sai lầm loại hai (ở đây là cực tiểu hóa β).

Ta ký hiệu $d(\alpha,\beta)=\alpha \log\frac{\alpha}{1-\beta}+(1-\alpha)\log\frac{1-\alpha}{\beta}$ (3)

Khi đó, Entropy giữa hai phân bố P_c và P_s và entropy tương đối của hai phân bố với tham số $(\alpha,1-\alpha)$ và $(\beta,1-\beta)$ cần thỏa mãn:

$$d(\alpha,\beta) \le D(P_c||P_s) \quad (2)$$

Như vậy, đối với hệ thống ε-an toàn chúng ta có:

$$d(\alpha,\beta) \le \varepsilon \quad (4)$$

Khi ε=0 hệ thống steganography được gọi là an toàn tuyệt đối.

2.2 Dung lượng chứa thông tin ẩn(steganography capacity**).**

Để đảm bảo tính chất không thể cảm nhận được (impercepcity) mỗi pixel ảnh không được giấu quá một bit dữ liệu. Như vậy tỷ lệ giữa độ dài (quy ra bit) bức thông điệp cần nhúng với số các pixel ảnh môi trường là $\frac{1}{8}$=12,5%. Nếu độ dài thông điệp cần nhúng so với số pixel ảnh môi trường mà vượt quá con số này thì khả năng vi phạm tiêu chuẩn "imperceptibility" là rất lớn.

2.3 Các kỹ thuật giấu tin mật trong ảnh (image steganography)

Hiện nay, đã có một số thuật toán giấu đã được đề nghị. Các thuật toán này được nhúng trong hai miền: miền không gian (Spatial Domain) và miền biến đổi (Transform Domain).

2.3.1 Nhúng tin trong miền không gian (Spatial Domain Embedding)

Việc nhúng tin trong miền không gian thường được thực hiện với kỹ thuật giấu tin mật. Các thuật toán giấu tin mật nổi tiếng nhất hiện nay đều dựa trên sự thay đổi các bít ít ý nghĩa nhất (Least Significant Bit-LSB) của các điểm ảnh và được gọi là kỹ thuật LSB. Kỹ thuật LSB có nhược điểm là dễ bị phát hiện khi truyền trên băng thông thấp. Tuy nhiên nó có nhiều ưu điểm là nhúng được nhiều lượng thông tin và ít bị mất thông tin trong quá trình truyền.

Chính vì vậy, các thuật toán steganography đều thực hiện trên miền không gian là chủ yếu

2.3.2 Nhúng thông tin trong miền biến đổi(Transform Domain Embedding).

Những thuật toán nhúng trong miền biến đổi đều tận dụng độ dư thừa trong miền DCT (Diserete Cosine Transformatio), chủ yếu đối với ảnh nén JPEG.

Việc nhúng tin trong miền DCT được thực hiện bằng cách thay đổi các hệ số, chẳng hạn như là thay đổi bit ít ý nghĩa nhất của mỗi hệ số.

Một trong những hạn chế của việc nhúng thông tin trong miền DCT là ở chỗ có 64 hệ số bằng 0 (zero) và việc thay đổi hai hay nhiều con zero thành hệ số khác không sẽ ảnh hưởng đến tỷ lệ nén. Vì vậy, số các bit 1 cần thiết nhúng trong miền DCT sẽ ít hơn nhiều so với số các bit 1 được nhúng bằng phương pháp LSB. Do đó khả năng chứa dữ liệu ẩn đối với miền DCT phụ

thuộc vào dạng ảnh được sử dụng trong trường hợp nhúng DCT, bởi vì nó phụ thuộc vào cấu trúc (texture) của ảnh do số các hệ số DCT khác 0 sẽ thay đổi.

Mặc dù việc thay đổi các hệ số DCT sẽ tạo nên những tiểu xảo (artifices) không thể nhận thấy được nhưng nó gây ra sự thay đổi mà các kỹ thuật thống kê có thể dò tìm được.

CHƯƠNG 3: GIẤU TIN TRÊN ẢNH TĨNH

3.1 Giấu tin trong ảnh những đặc trưng và tính chất

Như đã được trình bày ở trên, giấu tin trong ảnh chiếm vị trí chủ yếu trong các kỹ thuật giấu tin chính vì vậy các kỹ thuật giấu tin phần lớn cũng tập trung vào các kỹ thuật giấu tin trong ảnh. Các phương tiện chứa khác nhau thì cũng sẽ có các kỹ thuật giấu khác nhau. Đối tượng ảnh là một đối tượng dữ liệu được tri giác tĩnh có nghĩa là dữ liệu tri giác không biến đổi theo thời gian(không giống như audio và video) và có nhiều định dạng cũng như tính chất của các ảnh khác nhau nên các kỹ thuật giấu tin trong ảnh phải chủ ý những đặc trưng và tính chất cơ bản sau đây:

3.1.1 Phương tiện chứa có giữ liệu tri giác tĩnh

Dữ liệu gốc ở đây là ảnh tĩnh, dù đã giấu thông tin vào trong ảnh hay chưa thì khi ta xem ảnh bằng thị giác, dữ liệu ảnh không thay đổi theo thời gian, điều này khác với dữ liệu audio hay video và khi ta nghe hay xem dữ liệu gốc sẽ thay đổi liên tục với tri giác con người theo các đoạn hay các bài, các ảnh…Sự khác biệt này sẽ ảnh hưởng lớn đối với các kỹ thuật giấu thông tin trong ảnh với kỹ thuật giấu thông tin trong video hay audio.

3.1.2 Kỹ thuật giấu phụ thuộc vào ảnh

Kỹ thuật giấu tin phụ thuộc vào các loại ảnh khác nhau. Chẳng hạn như đối với ảnh đen trắng, ảnh xám hay ảnh màu đều đòi hỏi những kỹ thuật riêng, ảnh nén hay ảnh không nén cũng có những kỹ thuật giấu khác nhau vì ảnh nén có thể mất mát thông tin ảnh do nén ảnh…

3.1.3 Kỹ thuật giấu tin lợi dụng tính chất hệ thống thị giác của con người (HSV)

Giấu tin trong ảnh ít nhiều cũng gây ra những thay đổi trên dữ liệu ảnh gốc. Dữ liệu ảnh được quan sát bằng hệ thống thị giác(HSV_ Human Vision System)của con người nên các kỹ thuật giấu tin phải đảm bảo một yêu cầu cơ bản những thay đổi trên ảnh là rất nhỏ sao cho bằng mắt thường không thể nhận biết được sự khác biệt vì có như thế thì mới đảm bảo độ an toàn cho thông tin giấu. Rất nhiều kỹ thuật đã lợi dụng tính chất của hệ thống thị giác để giấu tin như chẳng hạn mắt người cảm nhận về độ xám kém hơn sự biến đổi về màu hay sự cảm nhận của mắt về màu xanh da trời (Blue) là kém nhất trong ba màu cơ bản RGB.

3.1.4 Giấu thông tin trong ảnh tác động lên dữ liệu ảnh nhưng không thay đổi kích thước của ảnh.

Các phép toán thực hiện việc giấu thông tin sẽ được thao tác trên dữ liệu của ảnh. Dữ liệu ảnh bao gồm cả phần thông tin ảnh(header), bảng màu(có thể có) và dữ liệu ảnh. Khi giấu thông tin, các phương pháp giấu để biến đổi các giá trị của các bit trong dữ liệu ảnh chứ không thêm vào hay bớt đi dữ liệu ảnh. Do vậy mà kích thước ảnh trước hay khi giấu thông tin là như nhau.

3.1.5 Đảm bảo yêu cầu chất lượng ảnh sau khi giấu thông tin.

Đây là một yêu cầu quan trọng đối với giấu thông tin trong ảnh. Sau khi giấu thông tin bên trong, ảnh phải đảm bảo yêu cầu không bị biến đổi để có thể bị phát hiện dễ dàng so với ảnh gốc. Yêu cầu này giường như khá đơn giản đối với ảnh màu hoặc ảnh xám bởi mỗi pixel ảnh được biểu diễn bởi nhiều bit, nhiều giá trị và khi ta thay đổi một giá trị nào đó thì chất lượng ảnh không thay đổi, thông tin giấu khó bị phát hiện, nhưng đối với ảnh đen trắng thì việc giấu thông tin phức tạp hơn nhiều, vì ảnh đen trắng mỗi pixel ảnh chỉ gồm hai giá trị hoặc trắng hoặc đen, và nếu ta biến đổi một bít từ đen thành trắng mà không khéo rất dễ bị phát hiện. Do đó, đối với yêu cầu các thuật toán

giấu thông tin trong ảnh màu hay ảnh xám và giấu thông tin trong ảnh đen trắng là khác nhau. Trong khi đối với ảnh màu thì các thuật toán chú trọng vào việc làm sao cho giấu được càng nhiều thông tin càng tốt thì các thuật toán áp dụng cho ảnh đen trắng thì lại tập trung vào làm thế nào để thông tin giấu khó bị phát hiện

3.1.6 Thông tin trong ảnh sẽ bị biến đổi nếu có bất cứ một biến đổi nào trên ảnh

Vì phương pháp giấu tin trên ảnh dựa trên việc điều chỉnh giá trị của các bit theo một quy tắc nào đó và khi giải mã sẽ theo các giá trị đó để tìm được thông tin giấu. Theo đó, nếu một phép biến đổi nào đó trên ảnh làm thay đổi giá trị của các bit thì sẽ làm cho thông tin giấu sẽ bị sai lệch. Chính đặc điểm này mà giấu thông tin trong ảnh có tác dụng nhận thực và xuyên tạc thông tin.

3.1.7 Cần thiết ảnh gốc khi giải mã ảnh?

Các kỹ thuật giấu tin phải phân biệt rõ ràng quá trình giải mã ảnh để lấy thông tin giấu có cần ảnh gốc hay không. Đa số các kỹ thuật giấu tin mật thì không cần ảnh gốc khi giải mã. Thông tin được giấu trong ảnh sẽ được mang cùng với dữ liệu ảnh, khi giải mã chỉ cần ảnh đã mang thông tin giấu và khóa để trích chọn thông tin ẩn mà không cần dùng đến ảnh gốc để so sánh đối chiếu.

Tuy nhiên, nhiều kỹ thuật giấu tin cũng sử dụng ảnh gốc khi mang giải mã ảnh, phương pháp này giúp cho việc đồng bộ hóa ảnh giấu và ảnh gốc. Điều này rất cần thiết khi phải xử lý đối với các tấn công trên ảnh. Giả sử như phép tấn công xoay ảnh chẳng hạn, nhờ có ảnh gốc ta so sánh và đồng bộ hóa và khôi phục dạng ban đầu của ảnh thì có thể khôi phục lại tin đã giấu. Nhưng phương pháp này cũng gặp khó khăn khi dữ liệu gốc lớn. Ví dụ như giấu tin trong video, với lượng dữ liệu lớn nếu như để giải mã mà dùng

phương pháp này thì khối lượng thao tác quá nhiều và không thể áp dụng được.

Trên đây là những tính chất và những đặc điểm cơ bản chung của giấu tin trong ảnh. Riêng đối với ứng dụng giấu tin mật(steganography)thì các tính chất ẩn, lượng thông tin giấu và độ an toàn là ba tính chất quan trọng nhất.

3.2 Giấu thông tin trong ảnh đen trắng, ảnh màu và ảnh đa cấp xám

Khởi nguồn của giấu thông tin trong ảnh là thông tin được giấu trong các ảnh màu hoặc ảnh xám trong đó mỗi pixel ảnh mang nhiều giá trị, được biểu diễn bằng nhiều bit. Với những ảnh đó thì việc thay đổi một giá trị nhỏ ở một pixel thì chất lượng ảnh giường như vẫn không thay đổi và khả năng bị phát hiện là rất thấp dưới sự quan sát của mắt thường, do đó hệ thống thị giác của con người cũng đóng một vai trò quan trọng trong việc đảm bảo tính ẩn của thông tin giấu trên ảnh. Với những ảnh mà mỗi điểm ảnh chỉ mang một giới hạn nhỏ các giá trị thì việc giấu thông tin trong ảnh đảm bảo tính ẩn của thông tin giấu là một công việc khó khăn hơn nhiều đặc biệt đối với ảnh đen trắng, mỗi điểm ảnh chỉ mang một trong hai giá trị trắng hoặc đen. Vậy thì khi thay đổi giá trị một pixel từ đen thành trắng hoặc ngược lại thì rất dễ bị phát hiện. Và do đó với ảnh đen trắng thì số lượng thuật toán không nhiều và vẫn chưa đạt được kết qua mong muốn. Có thuật toán giấu được nhiều thông tin vào ảnh thì chất lượng ảnh lại kém và rất dễ bị phát hiện. Một số thuật toán khi giấu chất lượng ảnh tốt hơn nhưng lượng thông tin giấu được ít và quá đơn giản không đam bảo được độ an toàn thông tin. Bảng sau sẽ liên kết sự khác nhau cơ bản giữa giấu thông tin trong ảnh đen trắng và ảnh màu.

Giấu thông tin trong ảnh đen trắng	Giấu thông tin trong ảnh màu hoặc ảnh xám
Thông tin giấu it hơn đối với ảnh có cùng kích cỡ với ảnh màu	Thông tin giấu nhiều hơn.
Khả năng bị phát hiện trong ảnh có giấu thông tin cao hơn ảnh màu	Khả năng bị phát hiện thấp
Độ an toàn thông tin thấp do dễ bị phát hiện có thông tin chứa bên trong	Độ an toàn cao
Các thuật toán giấu ít, phức tạp	Nhiều thuật toán và có nhiều hướng mở rộng phát triển. Như áp dụng giải thuật di truyền

Sự khác nhau giữa giấu thông tin trong ảnh đen trắng và ảnh màu

3.3 Cấu trúc ảnh BITMAP

Các kỹ thuật giấu tin ở phần sau được thực hiện trên ảnh Bitmap, trên phần này chúng ta cùng tìm hiểu cấu trúc ảnh để hỗ trợ cho việc cài đặt các kỹ thuật giấu tin.

Ảnh BMP (Bitmap) được phát triển bởi Mcrosoft Corporation, được lưu trữ dưới dạng thiết bị độc lập cho phép Window hiển thị dữ liệu không phụ thuộc vào khung chỉ định màu trên bất kỳ phần cứng nào. Tên file mở rộng mặc định của một file ảnh là BMP. ảnh BMP được sử dụng trên Mcrosoft Window và các ứng dụng chạy trên Windows từ version 3.0 trở lên.

Mỗi file ảnh BMP gồm 3 phần:

-BitmapHearder.

-Palette màu.

-Bitmapdata

Bitmapheader(54byte)
Bảng màu (có thể có hoặc không)
Thông tin ảnh (Bitmap Data)

Cấu trúc ảnh Bitmap

Cấu trúc cụ thể của ảnh Bitmap như sau:

Bitmapheader: được cho trong bảng sau:

Byte	Đặt tên	Ý nghĩa	Giá trị
1-2	ID	Nhận dạng file	'BM' hay 19778
3-6	File_Size	Kích thước file	Kiểu long trong Tuebo C
7-10	Reserved	Dành riêng	Mang giá trị 0
11-14	OffsetBit	Byte bắt đầu vùng dữ liệu	Offset của byte bắt đầu vùng dữ liệu
15-18	ISize	Số byte cho vùng info	40 byte
19-22	Width	Chiều rộng ảnh BMP	Tính bằng pixel
23-26	Height	Chiều cao ảnh BMP	Tính bằng pixel
27-28	Planes	Số planes màu	Cố định là 1
29-30	bitCount	Số bit cho một pixel	Có thể là 1,4,8,16,24
31-34	Compression	Kiểu nén dữ liệu	0: Không nén 1:Nén runlength 8bit/pixel 2:Nén runlength 4bit/pixel
35-38	ImageSize	Kích thước ảnh	Tính bằng byte
39-42	XPelsPerMeter	Độ phân giải ngang	Tính bằng pixel/meter
43-46	YPelsPerMeter	Độ phân giải ngang	Tính bằng pixel/meter
47-50	ColorUsed	Số màu sử dụng trong ảnh	
51-54	ColorsImportant	Số màu sử dụng trong khi hiện ảnh	

Cấu trúc header của file ảnh Bitmap

Palette màu: bảng màu của ảnh, chỉ những ảnh nhỏ hơn hoặc bằng 8 bit màu mới có palette màu.

Bitmap Data: Phần này nằm sau phần palette màu của BMP. Đây là phần chứa giá trị màu của điểm ảnh trong BMP. Các dòng ảnh được lưu từ dưới lên trên, các điểm ảnh được lưu từ trái sang phải. Giá trị của mỗi điểm ảnh là một số trỏ tới phần tử màu tương ứng của palette màu.

Thành phần bitCount của cấu trúc BitmapHeader cho biết số bit dành cho điểm ảnh và số lượng màu lớn nhất của ảnh. BitCount có thể nhận các giá trị sau

- 1: Bitmap là ảnh đen trắng, mỗi bít biểu diễn một điểm ảnh. Nếu bít mang giá trị 0 thì điểm ảnh là đen, bit mang giá trị 1 thì điểm ảnh là trắng.
- 4: Bitmap là ảnh 16 màu, mối điểm ảnh được biểu diễn bởi 8 bit.
- 8: Bitmap là ảnh 256 màu, mỗi điểm ảnh được biểu diễn bởi 1byte
- 16: Bitmap là ảnh high color, mỗi dãy 2 byte liên tiếp trong bitmap biểu diễn cường độ tương đối của màu đỏ, xanh lá cây, xanh lơ của một điểm ảnh.
- 24: Bitmap là ảnh true color 2^{24} màu, mỗi dãy 3 byte liên tiếp trong bitmap biểu diễn cường độ tương đối của màu đỏ, xanh lá cây, xanh lơ của một điểm ảnh.

Thành phần ColorUsed của cấu trúc BitmapHeader xác định số lượng màu của palette màu thực sự được sử dụng để hiển thị Bitmap. Nếu thành phần này được đặt là 0, Bitmap sử dụng số màu lớn nhất tương ứng với giá trị của BitCount.

3.4 Một số kỹ năng xử lý ảnh trong kỹ thuật giấu tin.

Trong phần này chúng ta sẽ mô tả một số kỹ năng xử lý ảnh sử dụng trong kỹ thuật giấu tin. Các kỹ thuật được giới thiệu và mô tả bởi câu lệnh của ngôn ngữ lập trình C.

Đọc Hearder ảnh Bitmap

Để đọc được dữ liệu ảnh, chúng ta cần đọc được Hearder ảnh để biết thông tin ảnh và đọc bảng màu(nếu có của ảnh).

Ta khai báo cấu trúc của ảnh bitmap như sau, cấu trúc này tương ứng với những thành phần có trong hearder của ảnh bitmap đã giới thiệu ở trên.

```
typedef struct{
    unsigned int ID;
long File_size; //kích thước toàn tệp ảnh(bytes)
long Reserved;
long OffsetBit;
long Isize, Width,Heingt;
ungsigned int Planes,Bitcount;
long Comperession, Imagesize;
long XpelsPerMerter,YpelsPerMerter;
long ColorsUsed, ColorsImportant;
}BitmapHearder;
```

Khi đọc Hearder ảnh của ảnh ta chỉ việc dùng câu lệnh đọc một cấu trúc fread:

```
fread(&bmh,sizeof(bmh),1,pictute);
```

-bmh là một biến có kiểu là kiểu cấu trúc hearder đã định nghĩa.

-picture là biến file ảnh.

Sau khi đọc xong hearder của ảnh thì những thông tin về tính chất ảnh sau đây là cần thiết.

-bmh.Height: Chiều cao của ảnh.

-bmh.Bitcount: Số bít cho mỗi điểm ảnh (nhờ thông số này ta biết ảnh có bảng màu hay không). Để đọc dữ liệu ảnh tiếp theo

Đọc bảng màu của ảnh

Chỉ có ảnh mà mỗi điểm ảnh biểu diễn bởi số không lớn hơn 8 bit thì có bảng màu, trong trường hợp đó ta khai báo một cấu trúc màu và đọc bảng màu như sau:

typedef struct{singed char Red, Green, Blue,Reserved;}TRGB;

Cấu trúc bảng màu gồm bốn thành phần: Red, Green, Blue,Reserved, mỗi thành phần là 1 byte. Khi đó kích thước của bảng màu sẽ được tính bằng công thức:

Tablesize=power2(bmh.Bitcount)*sizeof(TRGB)

Với power2(x) là hàm tính 2^x, hàm power2(bmp.Bitcount)sẽ cho ta số màu để biểu diễn điểm ảnh. Chẳng hạn nếu mỗi điểm ảnh được biểu diễn bởi 8 bit thì màu biểu diễn ảnh là 2^8=256 màu. Mỗi màu được biểu diễn bằng 1 byte như cấu trúc ở trên nên kích thước bảng màu là 256*4=1024, nếu có 16 màu thì mảng có 16 phần tử hay nếu ảnh đen trắng thì mảng có hai phần tử. Tổng quát ta khai báo mảng màu như sau:

TRGB *color;

Cuối cùng, ta đọc bảng màu của ảnh:

TabelSize=power2(bmh*Bitcount)*sizeof(TRGB);

fread(color, TableSize,1,pictute);

Chia ảnh thành các khối mxn

Trong nhiều kỹ thuật giấu tin, ảnh ban đầu thường được chia nhỏ thành các khối có kích thước mxn. Đây chính là phần xử lý dữ liệu sau khi ta đã đọc một hearder và bảng màu.

Ta định nghĩa khối ảnh mxn là một ma trận hai chiều, kích thước mxn, mỗi phần tử của mảng hai chiều có giá trị tương ứng là giá trị các điểm ảnh.

Các kỹ thuật giấu tin thường chia nhỏ ảnh ra thành các khối, sau đó giấu tin vào các khối, cuối cùng ghép các khối để thu được ảnh ban đầu.

Để lấy ma trận điểm ảnh ta nên dùng một ma trận hai chiều chứa toàn bộ điểm ảnh sau đó lấy từng khối nhỏ ra một cách dễ dàng. Nhưng mỗi ảnh có kích thước khác nhau và ta chưa biết trước kích thước ảnh nên nếu ta dùng mảng hai chiều có kích thước cố định thì sẽ có thể thừa hoặc thiếu không gian nhớ để chứa vì vậy ta nên dùng biến kiểu con trỏ của con trỏ. Sau đó, cần bao nhiêu không gian nhớ thì xin cấp phát bấy nhiêu. Phần sau đây sẽ trình bày khai báo và thủ tục cấp phát không gian nhớ cho mảng hai chiều.

- Khai báo dữ liệu ảnh:

  Byte ** imagedata; // nếu là ảnh 256

  Hoặc long ** imagedata' // nếu là ảnh lớn hơn 256

- Thủ tục xin cấp phát không gian nhớ khi biết kích thước ảnh

  //thủ tục xin cấp phát không gian nhớ cho ma trận hai chiều khi biết số hàng và số cột

```
byte**alloc_grays(int cols , int rows)
{
    byte**p;
```

```
int i;
p=(byte**)malloc(rows*sizeof(byte*));
if(!p){
#ifdef DEBUG
fprintf(stderr, "alloc_gray()failed\n");
exit(1);
#else
return NULL;
#endif
}
p[0]=(BYTE*)malloc(rows*cols*sizeof(Byte));
if(!p[0])
{
#ifdefDEBUG
        fprintf(stderr, "alloc_grays()faild\n");
exit(1);
#else
        free(p);
return NULL;
#endif
}
for(I =0;i<rows;i++)
{
p[i]=&(p[0][i*cols]);
}
return p;
}
```

Sau khi lấy dữ liệu ảnh, ta sử dụng các ma trận có kích thước m x n để đọc dữ liệu ảnh từ ma trận ảnh hai chiều imagedata.

- **Kỹ thuật ghép ảnh mới.**

Sau khi giấu dữ liệu xong ta ghép thành ảnh mới theo thứ tự sau:

-Ghi Hearder ảnh mới lấy từ ảnh cũ.

-Ghi bảng màu vào ảnh mới sang ảnh cũ.

-Đưa dữ liệu ảnh mới (đã giấu tin) vào trong ảnh

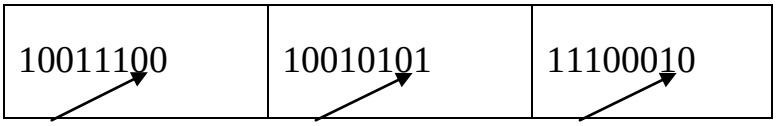
Các kỹ thuật xử lý điểm ảnh

Xử lý điểm ảnh là một kỹ thuật đươc sử dụng thường xuyên trong các kỹ thuật giấu tin trong ảnh. Các giá trị điểm ảnh được lấy ra rồi biến đổi theo thuật toán giấu tin. Tuy nhiên, miền giá trị của các điểm ảnh lại khác nhau phụ thuộc vào các loại ảnh, chính vì thế ta cần dùng đến kỹ thuật tách bit thông tin từ giá trị điểm ảnh.

Kiểu ảnh	Đen trắng	Ảnh 16 màu	Ảnh 256 màu	Ảnh Hing color	Ảnh true color(hơn 16 triệu màu)
Số bit/pixel	1	4	8	16	24

Kỹ thuật này được sử dụng nhiều trong kỹ thuật giấu tin sử dụng các bit ít quan trọng nhất của điểm ảnh(LBS_Least Significant Bit). Kỹ thuật LSB là kỹ thuật sử dụng các bít ít quan trọng về thị giác nhất trong các bit mang giá trị điểm ảnh để giấu tin. Ví dụ với ảnh 256 màu thì bit cuối cùng trong 8 bit biểu diễn một điểm ảnh được coi là bit ít quan trọng nhất theo nghĩa là nếu thay đổi bit này thì ảnh hưởng ít nhất đến cảm nhận của mắt người về điểm ảnh. Hay đối với ảnh 16 bit thì 15 bit biểu diễn ba màu RGB của điểm ảnh còn bit cuối cùng không dùng đến thì ta sẽ tách bit này ra ở mỗi điểm ảnh để giấu tin…Như vậy, kỹ thuật tách bít trong xử lý điểm ảnh được xử dụng rất nhiều trong quy trình giấu tin, sau đây ta sẽ khảo sát một số kỹ thuật tách bit it quan trọng trên một số loại ảnh phổ biến.

Tách bit cuối cùng trong 8 bit biểu diễn mỗi điểm ảnh của ảnh 256 màu.



Trong phép tách này ta coi bít cuối cùng là bit it quan trọng nhất, thay đổi giá trị của bit này thì sẽ thay đổi giá trị của điểm ảnh lên hoặc xuống đúng một đơn vị, ví dụ như giá trị điểm ảnh là 234 thì khi thay đổi bit cuối cùng nó có thể mang giá trị mới là 235 nếu đổi bit cuối cùng từ 0 thành 1. Với sự thay đổi đó ta hy vọng là cấp độ màu của điểm ảnh sẽ không bị thay đổi nhiều.

Ta thực hiện tách như sau:

//đọc một giá trị của điểm ảnh

c=getc(picture);

c1=c&1;

Sau hai câu lệnh này thì c1 sẽ mang giá trị là 0 hoặc 1 ứng với bit cuối cùng của biến c. Thật vậy, c là giá trị của một điểm ảnh nào đó, giả sử c=01100101(đối với ảnh 8 bit), c1=c&1 đây là phép toán nhân theo bit thông thường.

01100101

& 00000001(giá trị của 1 lưu trên một byte 8 bit)

c1=00000001(c1 bằng giá trị của bit cuối cùng)

Tách phần Blue trong RGB

Đối với ảnh 24 bit màu, mỗi màu được biểu diễn bởi 8 bit theo thứ tự R, G, B thì người ta thường dùng kỹ thuật tách thành phần Blue (B) trong RGB để giấu tin vì mắt người cảm nhận thành phần blue kém hơn so với hai màu còn lại. Kỹ thuật này được thực hiện như sau:

Mỗi lần đọc điểm ảnh, ta đọc vào một cấu trúc bản ghi gồm ba thành phần R,G,B sau đó sẽ sử dụng thành phần B và có thể lại sử dụng kỹ thuật tách bit ít quan trọng đối với thành phần B.

```
typedef struct{
byte R,G,B;
}pixel;
```

Biến đổi không gian màu cho ảnh 24 bit màu

Đối với ảnh 24 bit màu, người ta còn hay sử dụng một kỹ thuật nữa là kỹ thuật biến đổi không gian màu ví dụ như từ RGB sang YIQ, hay từ RGB sang HSV… ở phần này, chỉ giới thiệu một biến đổi RGB sang YIQ và ngược lại vì đây là kỹ thuật thường được dùng nhất trong kỹ thuật giấu tin trong ảnh với ảnh 24 bit màu. Cũng giống như tách thành phần B trong tổ hợp RGB của điểm ảnh, người ta thường dùng biến đổi này để lợi dụng tính chất của thị giác, trong ba thành phần Y,I,Q có thành phần Y biểu diễn độ chói của ảnh, hệ thông mắt người cảm nhận về độ chói kém hơn cảm nhận về màu chính vì thế kỹ thuật giấu tin thường biến đổi không gian màu từ RGB sang YIQ rồi lấy ra thành phần Y để giấu tin.

Công thức biến đổi không gian màu từ hệ RGB sang YIQ

$$\begin{cases} Y = 0.299*R + 0.587*G + 0.144*B \\ I = 0.596*R - 0.247*G - 0.322*B \\ Q = 0.211*R - 0.523*G + 0.321*B \end{cases}$$

Công thức biến đổi ngược từ hệ YIQ sang RGB

$$\begin{cases} R = 1.0*Y + 0.956*I + 0.262*Q \\ G = 1.0*Y - 0.272*I - 0.647*Q \\ B = 1.0*Y - 1.106*I + 1.703*Q \end{cases}$$

Trong công thức trên các thành phần RGB và YIQ nhận các giá trị thực chứ không phải giá trị nguyên như đã khai báo. Điều này được hiểu như sau:

ba màu cơ bản RGB được dung để tạo nên các màu khác nhau nhờ sự phối hợp tỷ lệ trong mỗi thành phần RGB. Ví dụ, để biểu diễn màu trắng tỉ lệ phối màu của ba màu tương ứng là (255,255,255), màu đen là (0,0,0)…

CHƯƠNG 4: MỘT SỐ KỸ THUẬT GIẤU TIN TRONG ẢNH ĐEN TRẮNG VÀ ẢNH MÀU

4.1 Một kỹ thuật giấu tin đơn giản

4.1.1 Ý tưởng

Đây có lẽ là kỹ thuật đơn giản nhất trong các kỹ thuật giấu tin. Ý tưởng cơ bản của thuật toán là chia một ảnh thành các khối nhỏ và với mỗi khối nhỏ đó sẽ giấu 1 bit thông tin. Thuật toán này dùng cho cả ảnh xám, ảnh màu và ảnh đen trắng nhưng để dễ trình bày thuật toán ta sẽ sử dụng ảnh đen trắng, phần sau sẽ trình bày phương pháp áp dụng thuật toán vào ảnh màu và ảnh xám.

4.1.2 Thuật toán giấu tin

Input:

- Một file ảnh Bitmap đen trắng F
- Một file thông tin cần giấu.

Output:

- Một file ảnh đã giấu tin F'
- Một khóa để giấu và giải tin K

Cách thực hiện:

Tiền xử lý:

- Chuyển file thông tin cần giấu P sang dạng nhị phân.

- Đọc header của ảnh để lấy thông tin ảnh, đọc bảng màu. Sau đó đọc toàn bộ dữ liệu ảnh vào một mảng hai chiều để sử dụng cho việc giấu tin.

Quá trình thực hiện giấu tin:

Chia phần thông tin ảnh (ma trận hai chiều điểm ảnh) thành các khối nhỏ có kích thước mxn. Giả xử ảnh gốc ban đầu có kích thước là MxN, khi đó, tổng số các khối nhỏ sẽ là (MxN)/(mxn) khối. Vì ảnh sử dụng là ảnh đen trắng nên mỗi khối là một ma trận hai chiều chứa các giá trị 0 và 1 như hình vẽ.

1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1
0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0
1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0
1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1
0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0
1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0
1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1
0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0
1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0

Một ví dụ về dữ liệu ảnh được chia thành các khối kích thước 4x4

Sau khi phân thành các khối nhỏ ta chọn các khối để giấu tin, ta có thể chọn ngẫu nhiên các khối nhưng để cho đơn giản ta coi như các khối được chọn tuần tự từ khối đầu tiên cho đến khi hết thông tin giấu.

Mỗi khối nhỏ này sẽ được dùng để giấu một bít thông tin theo quy tắc sau: sau khi giấu thì tổng số bit 1 trong khối và bít thông tin cần giấu sẽ có

cùng tính chẵn lẻ. Nghĩa là, nếu giấu bit 1 vào một khối thì khối đó phải thỏa mãn tính chất tổng số bit 1 trong khối là số lẻ. Ngược lại nếu cần giấu bit 0 vào một khối thì khối đó phải thỏa mãn tính chất tổng số bít 1 trong khối là chẵn.

Như vậy, mỗi lần giấu 1 bit ta có hai trường hợp sảy ra sau đây:

-Khối đã thỏa mãn tính chất để giấu một bít thông tin: chẳng hạn như khi cần giấu bit 1 thì tổng số bit 1 đã là lẻ rồi, hoặc khi cần giấu bit 0 vào khối thì khối đó đã có tổng số bit 1 là chẵn. Trong những trường hợp như thế, ta không cần thay đổi và xem như một bit thông tin đã được giấu.

-Trong trường hợp ngược lại, tính chất của khối không thỏa mãn yêu cầu để giấu tin nghĩa là khi cần giấu bit 1 thì tổng số bit 1 trong khối là chẵn hoặc khi cần giấu bit 0 thì tổng số bit 1 trong khối là lẻ. Trong những trường hợp như thế thì ta cần phải thay đổi khối đó sao cho thỏa mãn điều kiện bằng cách đảo trị ngẫu nhiên một bit(từ 0 sang 1 hoặc từ 1 sang 0).

Giả sử ta phải giấu bit 1 vào khối B sau:

1	0	1	1
0	1	0	0
0	0	1	0
1	1	1	0

Ta đếm số bít 1 trong khối: trường hợp ở trên khối B có 8 bit 1, như vậy khối B không thỏa mãn yêu cầu để giấu bit 1, để giấu bit 1 vào khối này ta cần phải thay đổi khối bằng cách chuyển một bit bất kỳ và đổi từ 0 sang 1 hoặc từ 1 sang 0. Giả sử ta đổi như hình vẽ sau:

Bít bị thay đổi từ 0 sang 1 →

1	0	1	1
0	0	0	0
0	0	1	0
1	1	1	0

Còn nếu, cũng với khối này mà ta cần phải giấu bit 0 thì ta không phải làm gì hết, vì ban đầu khối này đã thỏa mãn tính chất để giấu bit 0.

Mỗi lần giấu 1 bit ta lại lấy một khối để giấu theo quy tắc trên cho đến hết lượng thông tin cần giấu. Sau khi giấu xong ta được một ma trận dữ liệu hai chiều ảnh mới. Bước tiếp theo, ta xây dựng ảnh mới bằng cách: Chép header ảnh gốc đã đọc ra từ lúc đầu vào file ảnh mới, chép bảng màu đã đọc vào file ảnh mới, cuối cùng chép nốt dữ liệu ảnh mới sau khi đã giấu thông tin vào ảnh ta sẽ thu được ảnh mới sau khi giấu tin.

Trong thuật toán giấu tin này khóa đơn giản chỉ là kích thước của khối thì dễ dàng giải mã theo quy tắc sau:

-Đọc header của ảnh và bảng màu của ảnh để biết các thông tin của ảnh

-Lấy phần dữ liệu ảnh vào mảng hai chiều.

Các bước này giống như quá trình giấu tin. Sau khi đã có được dữ liệu ảnh ta lại chia ảnh thành các khối có kích thước khối giống như khi giấu, đây chính là khóa để giải mã. Chọn ra các khối đã giấu và giải tin theo quy tắc: đếm số bit 1 trong khối, nếu tổng số bit 1 là lẻ thì thu được bit 1, ngược lại thu được bit 0. Và cứ tiếp tục cho đến khi hết các khối đã giấu tin.

Như vậy, sau khi hết các khối đã giấu, ta thu được một chuỗi bít đã đem giấu. Bước tiếp theo ta chuyển từ file nhị phân sang file văn bản.

4.1.3 Phân tích thuật toán.

Đây là thuật toán rất đơn giản thực hiện một cách thức giấu tin trong ảnh, sau khi nghiên cứu thuật toán này ta có thể đưa ra một số bình luận và đánh giá sau đây:

-Việc chọn kích thước để giấu tin tùy thuộc vào kích thước của ảnh và khối lượng thông tin cần giấu sao cho giấu giàn trải trên toàn ảnh. Ví dụ, nếu ta có một ảnh kích thước 512x512 pixel và có một lượng thông tin cần giấu là 100 ký tự. Như vậy, file nhị phân thông tin cần giấu sẽ là 100*8=800 bit 0/1 vì mỗi ký tự mã ASCII biểu diễn bởi một byte. Ta có thể thấy để giấu được hết thông tin thì cần ít nhất 800 khối như vậy thì ta nên chia khối như thế nào để đủ khối giấu và giàn trải rộng trên ảnh. Lấy 512*512 /800=327 dư 544. Với kết quả này, kích thước khối tối đa là 327 vậy thì ta có thể chọn các kích thước phù hợp với con số này (phù hợp theo nghĩa đủ lớn và không vượt quá 327) chẳng hạn như 20x15, 16x16…

-Sở dĩ ta nên chọn khối có kích thước lớn vì như vậy nếu như trong trường hợp các khối bị thay đổi sẽ xa nhau (thưa) làm cho ảnh sau khi giấu khó bị nhận biết hơn.

-Với thuật toán này việc chọn khối khá đơn giản, ta bắt đầu từ khối đầu tiên và những khối liên tiếp phía sau tuần tự. Tuy nhiên, ta có thể làm khó thuật toán hơn bằng cách chọn ngẫu nhiên một khối chưa giấu ở mỗi lần giấu. Khi đó, ta đã làm tăng được độ an toàn của thuật toán vì khóa bây giờ còn thêm cả chỉ số khối đã giấu tin cho từng bit. Hoặc ta có thể thay đổi kích thước khối ở mỗi lần giấu, chẳng hạn như lần một có kích thước khối là 8*8, lần 2 là 8*12…trong trường hợp này thì khóa sẽ là kích thước khối của mỗi lần giấu.

-Một nhận xét quan trọng nữa thông qua thuật toán này là ta phải hiểu được bản chất của giấu tin được thực hiện trong kỹ thuật này. Bản chất ở đây là cách thức giấu chẳng qua chỉ là quy ước nào đó, nếu thỏa mãn thì giấu bít 1, ngược lại thì giấu bít 0. Điều này khác hẳn với giấu cái bút bi trong cái bàn vì thực tế là ta có cái bút bi thực sự và phải giấu nó đâu đó trong cái bàn còn

xét trong kỹ thuật giấu tin thì bản chất là ta không có cái bút bi nào hết mà chỉ là thông tin về bút bi.

4.1.4 Cài đặt

Để thực hiện thuật toán trên ta cần các kỹ thuật sau đây:

1) Đọc header ảnh

2) Đọc bảng màu của ảnh

3) Đọc dữ liệu vào mảng hai chiều

4) Tách ảnh thành các khối nhỏ

5) Giấu tin trong một khối

6) Chuyển file văn bản sang file nhị phân

7) Chuyển file nhị phân về file văn bản

8) Kỹ thuật giải tin

9) Kỹ thuật so sánh hai file để xem file giấu tin vào và file thông tin được gỡ ra có giống nhau hay không.

Kỹ thuật giấu tin vào một khối

Đây là một kỹ thuật đơn giản, ta chỉ việc duyệt khối nhỏ và đếm tổng số bit 1 và kiểm tra điều kiện để giấu tin. Nếu như cần phải thay đổi một bít nào đó ta dùng lệnh sau:

x=x-1;

Với x là giá trị của một phần tử bất kỳ của mảng hai chiều khối điểm ảnh, sau câu lệnh này bít x sẽ bị lật từ 0 thành 1 hoặc ngược lại từ 1 thành 0.

Kỹ thuật chuyển file văn bản sang file nhị phân

Ta lấy mỗi ký tự của văn bản và chuyển sang nhị phân theo thủ tục sau:

```
void Convert2Bin(char *text, char *bintext)
{
    FILE *t,*bin;
    int c,b,i;
    if((t=fopen(text, "r"))==NULL)
    {
        printf("Khong mo duoc file %s",text);
        gertch();
        exit();
    }
    if((bin=fopen(bintext, "wb"))==NULL)
    {
        printf("Khong mo duoc file %s",bintext);
        gertch();
        exit();
    }
    while((c=getc(t))=EOF)
        for(i=7;i>=0;i--)
        {
            b=(c>>i)&1; //dich trai de lay tung bit cua byte
            putc(b,bim);  //dua vao file nhi phan
        }
    fclose (t);
    fclose(bin);
}
```

Trong đó text là tên file văn bản, bintext là tên file nhị phân của văn bản.

Kỹ thuật chuyển đổi ngược từ file nhị phân sang file văn bản.

Sau khi thu được file nhị phân, công việc chuyển ngược sang file văn bản được thực hiện qua thủ tục sau:

```
void Convert2text(char *bintext ,char *text)
{
    FILE *bt,*kq;
    int c,b,i;
    if((bt=fopen(bintext, "rb"))==NULL)
    {
```

```
            printf("Khong mo duoc file %s", bintext );
            gertch();
            exit();
        }
        if((kq=fopen(text, "w"))==NULL)
        {
            printf("Khong mo duoc file %s",text);
            gertch();
            exit();
        }
        B=0;i=7;
        while((c=getc(bt))!=EOF)
                if (c) b<=(c<<i);
                --i;
                if(i<0)
                {
                    putc(c,kq); b=0;
                    i=7;
                }
        fclose (bt);
        fclose(kq);
}
```

Trong đó bintext là tên file nhị phân, text là tên file văn bản.

```
power2(int x)
{
    int temp;
    temp=1<<x;  //dich trai bit 1 di x don vi nghia la 2^x
    return temp;
}
```

Kỹ thuật giải tin:

Kỹ thuật này đơn giản nó bao gồm các thủ tục như khi giấu tin nhưng chỉ khác nhau là khi lấy từng khối ảnh ra ta chỉ việc tính tổng số bit 1 rồi ghi lại kết quả.

Kỹ thuật so sánh hai file

Ta sử dụng thủ tục sau đây:

```
void compare (char * fn, char *gn)
{
      int cf,cg;
```

```
    FILE *f, *g ;
    long d=0;
    if((f=fopen(fn, "r"))==NULL)
    {
        printf("Khong mo duoc file %s", fn );
        gertch();
        exit();
    }
    if((g=fopen(gn, "rb"))==NULL)
    {
        printf("Khong mo duoc file %s", gn );
        gertch();
        exit();
    }
    while (1)
    {
        cf=getc(f) ;
        cg=getc(g) ;
        if (cf==EOF || cg==EOF) break;
        if (cf != cg) break;
        ++ d;
    }
    if (cf!=cg)
    printf ("\n sai o byte thu %d",d);
    else printf ("\n %s=%s\n",fn,fg);
    fclose(f);
    fclose(g);
    getch();
}
```

Trong đó: fn và gn là hai file cần so sánh, nếu hai file khác nhau thì thủ tục sẽ thông báo vị trí byte sai đầu tiên.

4.1.5 Vấn đề áp dụng thuật toán trong ảnh đen trắng và ảnh màu, ảnh đa cấp xám.

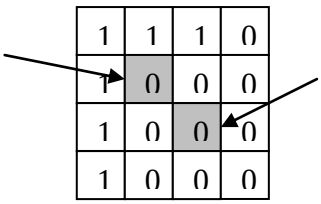
Thuật toán này mặc dù áp dụng cho ảnh đen trắng nhưng nó cũng có thể sử dụng cho ảnh màu hoặc ảnh đa cấp xám. Phần này chúng ta sẽ làm rõ việc áp dụng thuật toán vào các loại ảnh và những điều quan trọng khi áp dụng kỹ thuật cho từng ảnh.

Áp dụng thuật toán cho ảnh đen trắng

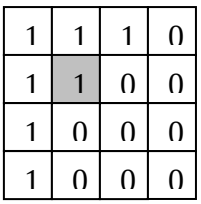
Thuật toán trên được trình bày cho ảnh đen trắng nên ta chỉ quan tâm đến những vấn đề cốt yếu khi áp dụng cho ảnh đen trắng. Như ta đã biết ảnh đen trắng khó giấu hơn do đặc điểm, mỗi điểm ảnh chỉ được biểu diễn bởi 1 bit hoặc 0 hoặc 1. Nếu như ta thay đổi bit 0 sang 1 hay ngược lại từ 1 sang 0 thì đều làm cho trên ảnh xuất hiện những điểm đen, điểm trắng lạ. Như vậy vấn đề cốt yếu ở đây là làm thế nào hạn chế tối đa các điểm đen điểm trắng lạ và làm thế nào để những bít bị thay đổi đó khó bị phát hiện nhất. Ta sẽ nghiên cứu một số kỹ thuật cải tiến dành cho ảnh đen trắng sau đây:

Ý tưởng của phần cải tiến này dựa vào một nhận xét: với các ảnh đen trắng thì việc thay đổi một giá trị một bit điểm ảnh từ trắng thành đen hoặc ngược lại thì rất dễ bị phát hiện (bị nhiễu). Nhưng nếu ta đảo bít ở trên viền ảnh giữa miền đen và miền trắng thì bít bị đảo sẽ khó bị nhận biết hơn.

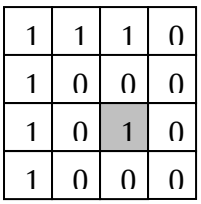
Ví dụ: Giả sử ta có một khối ảnh và các bit có thể đảo là hai bit được đánh giấu xám như trong hình vẽ dưới đây



a) Khối bít ban đầu



b)đảo ở vị trí 1



c)Đảo ở vị trí 2

mô tả các trường hợp thay đổi bit

Rõ ràng ta nhận thấy rằng nếu ta đảo bít như trong hình b thì bít bị đảo sẽ khó bị nhận biết hơn đảo bít như trong hình c.

Ý tưởng này đã được thực hiện nhờ một hệ số phân bố bit D. Hệ số phân bố bít D là một đại lượng đặc trưng cho mức độ rời rạc của các bít 0,1 trên một ma trận điểm ảnh và được tính theo công thức sau:

Giả sử ta có một ma trận A chứa các điểm ảnh 0,1, cỡ mxn;

$$\mathbf{D=D_h + D_v + D_c + D_a}$$

Trong đó:

D_h: là hệ số phân bố bít theo chiều ngang.

$$D_h=\sum_{i=1}^{m}\sum_{j=1}^{n-1} (A_{i,j} \text{ xor } A_{i,j+i})$$

D_v là hệ số phân bố theo chiều dọc.

$$D_v=\sum_{j=1}^{n}\sum_{i=1}^{m-11} (A_{i,j} \text{ xor } A_{i+1,j})$$

D_c là hệ số phân bố bit theo đường chéo 1.

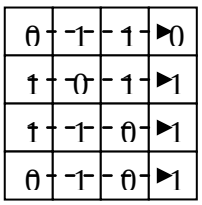
$$D_c=\sum_{i=2}^{m}\sum_{j=1}^{n-1} (A_{i,j} \text{ xor } A_{i-1,j+i})$$

D_a là hệ số phân bố bít đường chéo 2.

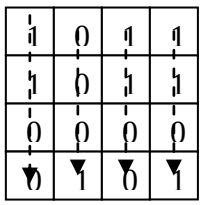
$$D_h=\sum_{i=1}^{m-1}\sum_{j=1}^{n-1} (A_{i,j} \text{ xor } A_{i+1,j+i})$$

Với xor là phép toán XOR logic x xor y=1 nếu x # y, ngược lại x xor y =0 nếu x=y.

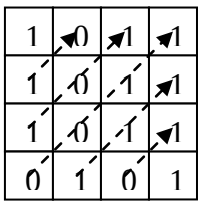
Thực chất nếu ta duyệt các phần tử của ma trận theo từng dòng và đếm số lần chuyển màu (từ 1 sang 0 hoặc từ 0 sang 1) thì phân bố ngang D_h chính là số lần chuyển màu tính theo các dòng, D_v là tổng số lần chuyển màu tính theo cột, D_c là tổng số lần chuyển màu tín theo đường chéo 1, D_v là tổng số lần chuyển màu tính theo đường chéo 2.



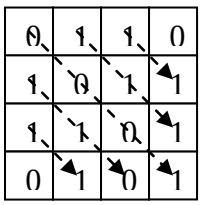
Số lần chuyển màu tính theo hàng D_h



Số lần chuyển màu tính theo cột D_v



Số lần chuyển màu tính theo đường chéo1



Số lần chuyển màu tính theo đường chéo2

Ví dụ: cho một ma trận nhị phân B 4x4 như sau:

1	1	0	1
0	0	1	0
1	0	1	0
1	0	1	1

Khi đó ta có các hệ số phân bố theo các chiều là:

- D_h=2 + 2 + 3 + 2 =9
- D_v=2 + 1 + 1 + 2 =6
- D_c= 1 + 2 + 2 +1 + 1 =7
- D_a =1 + 1 + 2 + 1 = 5

Hế số phân bố bit trên B là

$$D= D_h + D_v + D_c + D_a = 9+ 6 + 7 + 5 = 27$$

Nếu D càng nhỏ thì mức độ rời rạc càng thấp tức là độ kết dính giữa các bít 0,1 càng lớn. Và áp dụng trong thuật toán này ta sẽ chọn cách chọn đảo bít nào có D nhỏ nhất.

Một phần cải tiến nữa của thuật toán là hạn chế các khối giấu tin vì những khối có tỉ lệ bít đen rất thấp hoặc rất cao thì khi giấu thông tin thì rất ít khả năng đảo bít. Bít đảo sẽ rất dễ bị phát hiện. Và trong một số trường hợp trên ảnh có những khối toàn trắng hoặc toàn đen thì không nên giấu thông tin vào các khối đó. Trong thuật toán này đã dùng hai biến để chặn cận tỉ lệ bít đen trên một khối là MinBlack và MaxBlack.

Áp dụng thuật toán cho ảnh màu và ảnh đa cấp xám

Thuật toán ở trên hoàn toàn có thể áp dụng cho ảnh màu và ảnh đa cấp xám. Các loại ảnh này có giá trị của mỗi điểm ảnh được biểu diễn bằng

nhiều bít. Vậy làm thế nào để có được một ma trận điểm ảnh 0,1 để thực hiện việc giấu tin như thuật toán ở trên? Ta chỉ việc chọn từ mỗi điểm ảnh đúng một bít và lưu vào ma trận hai chiều các bít 0,1. Việc chọn này thực hiện chọn theo quy tắc chọn bit ít quan trọng nhất LSB (Least Significant Bit)

Đối với ảnh màu và ảnh đa cấp xám ta không cần quan tâm nhiều đến việc chọn điểm cần giấu vì ta đã dùng những bít ít quan trọng nhất để giấu rồi. Do vậy, tại mỗi bước giấu ta có thể chọn một bít bất kỳ để thay đổi.

4.2 Kỹ thuật giấu WU_LEE

Mục này giới thiệu chi tiết một thuật toán giấu tin trong ảnh đen trắng. Thuật toán này được đưa ra bởi M.Y. Wu và J.H. Lee trong proceedings of international Symposium on Multimedia Information Proscessing 1999.

Gọi a, b là hai bit tùy ý, phép toán nhân bit AND, ký hiệu là $\wedge$ trên hai bit a và b cho giá trị 1 khi và chỉ khi a=b=1, trong các trường hợp còn lại bằng 0.

Phép toán cộng loại trừ (còn gọi là phép toán so khác) XOR, ký hiệu là $\oplus$ trên hai bit a và b cho ta giá trị 1 nếu a khác b và giá trị 0 nếu a=b.

a	b	$a \wedge b$	$a \oplus b$
1	1	1	0
1	0	0	1
0	1	0	1
0	0	0	0

Cho A và B là hai ma trận bit cùng cấp. Ta phát triển các phép toán $\wedge$ và $\oplus$ trên các bít tương ứng của A và B như sau:

Nếu $A = (a_{ij}), B = (b_{ij}), C = (c_{ij}), D = (d_{ij})$

Thì $A \wedge B = C$ với $c_{ij} = a_{ij} \wedge b_{ij}$

Và $A \oplus B = D$ với $d_{ij} = a_{jj} \oplus b_{ij}$

Thí dụ:

Nếu cho A=

1	0	1	1
1	1	0	1
0	1	0	0
1	0	1	1

Nếu cho B=

0	0	1	0
0	1	1	1
1	0	1	0
1	1	0	1

Thì C=

0	0	1	0
0	1	0	1
0	0	0	0
1	0	0	1

D=

1	0	0	1
1	0	1	0
1	1	1	0
0	1	1	0

Ngoài ra, ta định nghĩa SUM (X) là tổng các giá trị 1 trên ma trận X. Ta có, theo thí dụ trên SUM(A) =10, SUM(B)=9, SUM(C)=5, SUM(D)=9. Để ý rằng nếu X là một ma trận bit thì SUM(X) chính là tổng số bít 1 trong X.

Quay trở lại thuật toán, giả sử ta có một ảnh đen trắng, ta sẽ coi ảnh như một ma trận điểm ảnh gồm những điểm 0,1. Với hai ma trận có cùng kích cỡ B1,B2. Ta ký hiệu:

- $B1 \wedge B2$ là phép toán AND giữa các cặp bit của hai ma trận
- $B1 \oplus B2$ là phép toán XOR giữa các cặp bít của hai ma trận
- Với một ma trận số nguyên thì B[ij] là phần tử của ma trận B nằm ở dòng i cột j.
- SUM (B) là tổng của tất cả các phần tử của B.

Thuật toán giấu thông tin sẽ được thực hiện như sau. Chúng ta có một ảnh gốc nhị phân F, một khóa bí mật K và một số các bít dữ liệu cần giấu.

Khóa K là một ma trận nhị phân có kích thước mxn. Để cho đơn giản chúng ta coi kích cỡ của ảnh F là bội của mxn. Việc nhúng thông tin giấu vào trong ảnh sẽ được thực hiện bằng cách thay đổi một số bít của ảnh F theo quy tắc:

S1: Chia ảnh F thành các khối nhỏ, mỗi khối có kích thước là mxn.

S2: Với mỗi khối ảnh nhỏ Fi thu được từ bước S1, ta kiểm tra điều kiện.

$$0<SUM(Fi \wedge K) <SUM(K)$$

Nếu đúng thì chuyển đến bước 3 để giấu thông tin vào trong khối Fi, còn nếu không thì không giấu dữ liệu vào trong khối Fi, khối Fi đươc dữ nguyên.

S3: gọi bit cần giấu vào trong khối Fi là b, thực hiện các bước sau để thay đổi Fi.

if(SUM(FiK)mod 2 =b)then
giữ nguyên Fi
else if (SUM (Fi∧K) =1) then
chọn ngẫu nhiên một bit (j,k) thỏa mãn đồng thời
$[Fi]_{jk}$ = 0 và $[K]_{jk}$ = 1 sau đó chuyển giá trị của bít $[Fi]_{jk}$ thành 1
else if (SUM (Fi∧K) = SUM (K)-1) then
chọn ngẫu nhiên một bit (j,k) thỏa mãn đồng thời
$[Fi]_{jk}$ =1 và $[K]_{jk}$ =1 sau đó chuyển giá trị của bít $[F]_{jk}$ thành 0;
else
chọn ngẫu nhiên một bit mà $[K]_{jk}$ =1 chuyển giá trị của bit $[Fi]_{jk}$ từ 0 trở thành 1, hoặc từ 1 trở thành 0.
end if

Ta tìm hiểu sơ bộ ý tưởng của thuật toán Wu_Lee

-Việc chọn khóa K nhằm làm tăng độ mật của thuật toán. Nếu trước đây kích thước khối là mxn thì đối phương rất dễ khai thác được bản tin mật, nay ngoài kích thước này còn phải biết giá trị cụ thể của khóa K

-Phép toán Fi ∧K quy định thuật toán chỉ được phép sửa các bít trong khối Fi ứng với bit 1 trong khóa K. Như vậy, khóa K được xem như một mặt nạ, tạo ra khung nhìn cho thuật toán.

Dĩ nhiên ta có thể thay phép toán ∧ bằng 1 phép toán khác, chẳng hạn như phép XOR như các thuật toán được cải tiến sẽ được trình bày ở phần sau.

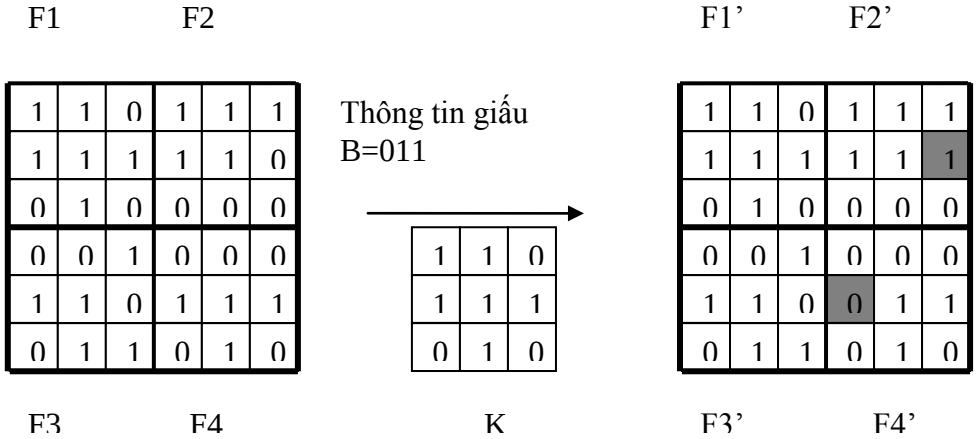
-Điều kiện 0< SUM(Fi∧K)<SUM(K) quy định nếu Fi∧ K toàn 0 hoặc giống như khóa K thì không được giấu tin để tránh bị lộ.

-Trong bước S3 ta thực hiện tối đa một phép đảo một bit của Fi để thu được khối Fi'nhằm đảm bảo tính bất biến.

SUM (Fi'∧K) mod 2 =b

-Việc chọn bít nào trong F để đảo cần tuân thủ theo nguyên tắc: Nếu Fi∧K có nhiều bit 1 thì chọn bit 1, ngược lại có quá ít bít 1 thì chọn bít 0. Nguyên tắc này làm giảm khả năng bit đảo bị phát hiện. Có thể dễ dàng kiểm chứng rằng sau bước S3 ta thu được bất biến trên. Nhờ bất biến đó, ta dễ dàng giải mã để lấy lại thông tin đã giấu như sau: Duyệt lần lượt các khối Fi' của ảnh đích F'. Nếu Fi' thỏa mãn điều kiện 0< SUM (Fi'∧K)<SUM (K) thì tính bit b đã được giấu vào trong khối bằng công thức tính b=SUM (Fi∧K) mod 2.

Vì khóa K là bí mật nên thông tin đã được nhúng là bí mật. Thuật toán này làm thay đổi nhiều nhất một bit của khối Fi khi giấu 1 bit thông tin vào trong khối nên với một khối có kích thước mxn đủ lớn thì sự thay đổi của Fi là nhỏ.



Mô tả quá trình đảo bít để giấu tin của thuật toán trên 4 khối

Chúng ta sẽ lấy một ví dụ cho thuật toán trên. Giả sử một ảnh F có kích thước 6x6 và một mà trận khóa K có kích thước 3x3 như trong hình vẽ. Ta chia F thành 4 khối nhỏ mỗi khối sẽ có kích thước là 3x3 ta thu được F1, F2, F3, F4.

-Vì SUM (F1 $\wedge$ K) = SUM(K) =5 nên không cần giấu dữ liệu vào trong F1.

-Vì SUM (F2 $\wedge$ K) = 3 nên một bít sẽ được giấu vào khối 2. Theo ví dụ trên thì bít đầu tiên được giấu là bit 0. Nên theo S3 ta sẽ chọn một bít $[F2]_{ij}$ =0 và $[K]_{ij}$ = 1 và đổi giá trị $[F2]_{ij}$ thành 1, F2 chuyển thành F2' như trên hình vẽ.

-Với F3, SUM (F3 $\wedge$ K) =3 nhưng bít cần giấu là bít 1 nên theo S1 ta giữ nguyên F3 nhưng thực tế F3 vẫn giấu được một bít 1.

-Tương tự đối với F4, SUM(F4 $\wedge$ K)=4, và bít cần giấu là bít 1 nên theo S3 ta chọn một bit $[F4]_{iJ}$ =1 và $[K]_{iJ}$ =1 rồi chuyển $[F]_{ij}$ =0.

4.2.2 Phân tích thuật toán

-Thứ nhất, vì phép toán AND được sử dụng để tính Fi $\wedge$ K, nên giá trị lớn nhất của SUM(Fi $\wedge$ K) không thể vượt quá SUM (K) và do tính chất của phép toán AND, nếu có một khối nào thay đổi thì vị trí thay đổi chỉ xảy ra ở phần tử có giá trị 1 trong khóa K. Vì thế, nếu một ảnh F hoàn toàn trắng nào đó được truyền đi thì kẻ thù khi bắt được thông tin sẽ dễ dàng tìm ra được vị trí 1 của khóa K, đó là lý do mà ta không dùng trường hợp SUM(Fi $\wedge$ K) = 0. Đây là một kẽ hở của thuật toán đối với khóa.

-Thứ hai, Với trường hợp SUM (Fi $\wedge$ K) = SUM (K) cũng tương tự nếu F hoàn toàn đen thì vị trí của bít thay đổi cũng là vị trí mà bít tương ứng ở khóa là 1.

Để tránh những trường hợp trên thuật toán đã đưa ra phụ thuộc 0<SUM (Fi∧K) <SUM (K). Nhưng cho dù như thế đi chăng nữa thì vị trí tương ứng với bít bị thay đổi cũng tương ứng với bit ở vị trí đó trong khóa K có giá trị 1, và bít không bao giờ bị thay đổi tưng ứng sẽ là bít 0 ở vị trí đó trong khóa K. Và như thế việc chọn khóa K như thế nào là một công việc hết sức quan trọng.

-Thứ ba, nếu ảnh F được lựa chọn để giấu thông tin có quá nhiều điểm trắng hoặc có quá nhiều điểm đen thì tỉ lệ bít giấu đươc sẽ rất thấp.

Nói chung thuật toán này vẫn chưa đạt được những yêu cầu cần thiết về khả năng giấu, độ an toàn thông tin cũng như chất lượng ảnh. Tuy nhiên, đó là áp dụng đối với ảnh đen trắng, nếu ta áp dụng kỹ thuật này cho ảnh màu thì cũng thu được kết quả khả quan.

4.2.3 Cài đặt

Để hiện thực hóa thuật toán này chúng ta cần một số kỹ thuật sau đây:

1. Kỹ thuật đọc ảnh (đọc header đọc bảng màu).
2. Kỹ thuật tách bít ít quan trọng (dùng trong trường hợp ảnh màu hoặc ảnh đa cấp xàm)
3. Kỹ thuật đọc dữ liệu ảnh
4. Kỹ thuật tính hệ số phân bố bit D áp dụng để nâng cao chất lượng ảnh.
5. Kỹ thuật chuyển file văn bản sang file nhị phân và ngược lại
6. Kỹ thuật tính toán trên ma trận hai chiều
7. Kỹ thuật giấu tin sử dụng thuật toán

8. Kỹ thuật giải tin

Hệ số phân bố bít D được tính theo thủ tục sau đây

```
int getscore (int matrix[m][n])
{
    int blachkbitcount =0,i,j;
    int sum;
    //Dem tong so bit 1 trong ma tran hai chieu
    for ( i=0; i< m; i++ )
        for (j =0 ; j< n; j ++)
            if (matrix[i][j]==1) blackbitcount +=1;
            /*nếu khối ảnh không phù hợp ví dụ như có quá nhiều điểm
            trắng hoặc có qua nhiều điểm đen thì ta phải cho nó số điểm
            cao để lại trừ khối này ra không giấu nữa*/
    if((blackbitcount < = MinBlackBit)|| (blackbitcount > =
        MaxBlackBit))
        return MAXINT ;
    else
        {
            for ( i=0 ;i< m;i++)
              for (j=0; j<n-1;j++)
                  if (matrix[i][j] != matrix[i][j+1])    sum+=1;
            for ( i=0 ;i< n;i++)
              for (j=0; j<m-1;j++)
                  if (matrix[i][j] != matrix[i+1][j])    sum+=1;
            //tinh diem cho truong hop duong cheo
            for ( i=0 ;i< mi++)
              for (j=0; j<n1;j++)
                  if (matrix[i][j] != matrix[i-][j+1    sum+=1;
            for ( i=0 ;i<m-1;i++)
              for (j=0; j<n-1;j++)
                  if (matrix[i][j] != matrix[i+1][j+1])    sum+=1;
        }
        return  sum;
}
```

4.3 Kỹ thuật giấu tin CHEN_PAN_TSENG(CPT)

Phần sau đây trình bày một thuật toán giấu thông tin trong ảnh đen trắng được phát triển từ thuật toán giấu tin của Yu_Yuan, Hsiang_kuang Pan va Yu__Chee Tseng, Khoa công nghệ thông tin và Khoa học máy tính thuộc trường Đại Học quốc gia Đài Loan… Phương pháp ban đầu này sử dụng một

ma trận khóa và một ma trận trọng số để giấu thông tin trong ảnh bằng cách thay đổi nhiều nhất hai bít mỗi khối ảnh. Nhược điểm của phương pháp này là chất lượng ảnh chưa cao, dễ bị phát hiện. Đối với ảnh đen trắng thì thuật toán này chưa đáp ứng được các yêu cầu nêu trên. Thuật toán cải tiến sẽ cải thiện được rất nhiều chất lượng ảnh bằng kỹ thuật chọn hệ số phân bố bit đen trắng và số bit giấu tương đương. Trước khi đi vào phần chi tiết kỹ thuật ta định nghĩa một số khái niệm dùng trong thuật toán.

4.3.1 Một số khái niệm dùng trong thuật toán:

a)Khóa bí mật

Khóa là một ma trận nhị phân có cùng kích thước mxn với kích thước của khối ảnh. Khóa được dùng một cách bí mật giữa người gửi và người nhận.

b) Ma trận trọng số cấp r:

Ma trận trọng số W cấp r là một ma trận số nguyên có kích thước bằng kích thước của khối ảnh mxn và thỏa mãn điều kiện sau:

-W có các phần tử nằm trong khoảng giá trị $(0\ldots2^{r-1})$ với r cho trước thỏa mãn điều kiện $2^r <m*n$.

-Với mỗi phần tử có giá trị từ $1\ldots2^{r-1}$ xuất hiện ít nhất một lần.

Ví dụ: xây dựng một ma trận trọng số kích thước 4x4, ma trận này có các giá trị nằm trong khoảng 0…15 và mỗi giá trị từ 1 đến 2^3 =7 xuất hiện ít nhất một lần.

Một ví dụ ma trận trọng số W.

1	2	3	4
5	6	7	1
2	3	4	5
4	5	7	1

c) Phép đảo bít

Phép đảo bít là một phép biến đổi trên các bít nhị phân. Đảo bít b tương đương với thay b bởi 1-b, tức là nếu ban đầu b nhận giá trị 0 thì sau khi đảo nó sẽ nhận giá trị 1 và ngược lại, nếu ban đầu b có giá trị là 1 thì sau khi đảo nó sẽ mang giá trị 0.

d) Các phép toán trên ma trận dùng trong thuật toán.

Phép toán XOR hai ma trận $A \oplus B$

Phép toán nhân hai ma trận $A \wedge B$ là nhân các phần tử tương ứng của A và B với nhau.

Phép toán tính tổng giá trị trong ma trận SUM (X)

4.3.2 Thuật toán

Intput:

-F là một ma trận ảnh gốc dùng để giấu thông tin. F được chia thành các khối Fi, mỗi ma trận điểm ảnh Fi có kích thước là mxn, để cho đơn giản ta giả sử rằng F là bội của các Fi.

-K là một ma trận khóa cấp mxn.

-W là một ma trận trọng số cấp mxn.

-r: Số lượng bit sẽ nhúng trong mỗi một khối ảnh mxn.

-B : là lượng thông tin cần giấu gồm k*r bít, k sẽ là số khối ảnh giấu.

Output:

Ảnh đích F’ chứa B. F’ được tạo từ các khối Fi’, Mỗi Fi’ thu được từ khối Fi tương ứng sau khi đã giấu r bit thông tin từ B.

Thuật toán:

Thuật toán sẽ thực hiện việc biến đổi mỗi Fi thành Fi' sao cho luôn thỏa mãn điều kiện sau:

$$SUM (Fi' \oplus K) \otimes W) \equiv b_1b_2b_3\ldots b_r \bmod 2^r.$$

Trong đó $b_1b_2b_3\ldots b_r$ là dạng biểu diễn của một số nhị phân tạo từ dãy r bit liên tiếp trong B. Mỗi Fi bị biến đổi nhiều nhất là 2 bit. Quá trình biến đổi gồm 4 bước sau đây:

Bước 1: Tính ma trận $T=Fi \oplus K$

Tính ma trận $P = T \otimes W$

Bước 2: Tính tổng Sum = SUM(P)

Bước 3: Với ma trận T và với mọi $w=1,2,\ldots,2^{r-1}$ ta xác định tập hợp S_w như sau:

$$S_w=\{(j,k)|(W[i,j]=w \wedge T[i,j]=0) \vee (W[j,k] =2^r -w \wedge T[j,k]=1)\}$$

Dễ nhận thấy S_w là tập hợp các tọa độ (i,k) của ma trận Fi sao cho khi đảo bít Fi[i,j] thì Sum ở bước hai tăng lên w đơn vị. Thực vậy, ta có:

-Trường hợp 1: Nếu W[i,j]=w và T[i,j]=0

Khi đó đảo bit Fi[i,j] sẽ làm cho T[j,k]=1, do đó Sum tăng lên w.

-Trường hợp 2: Nếu $W[j,k] =2^r -w$ và T[j,k]=1

Khi đó đảo bit Fi[i,j] sẽ làm T[i,j]=0, do đó Sum sẽ giảm đi 2^{r-1}-w, tức là tăng lên w theo mod 2^r.

Từ định nghĩa của tập S_w ta có:

$$S_w' = S_w$$

Bước 4: Ký hiệu $d=(b_1b_2...b_r) - SUM(P)(mod 2^r)$.

Ta cần thực hiện việc đảo bit trên Fi để được Fi' sao cho tổng Sum tính được ở bước 2 khi thay Fi bởi Fi' sẽ tăng lên d.

-Nếu d=0, không cần thay đổi Fi

-Nếu $d \neq 0$ ta thực hiện các công việc sau:

1) Chọn h bất kỳ thuộc tập $\{1,2,3\ldots,2^{r-1}\}$ sao cho $S_{hd} \neq \Phi$ và $S_{-(h-1)d} \neq \Phi$

2)Chọn phần tử (j,k) bất kỳ thuộc S_{hd} và đảo bit Fi[j,k]

3)Chọn phần tử (u,v) bất kỳ thuộc $S_{-(h-1)d}$ và đảo bít Fi[u,v]

Rõ ràng là để tăng Sum lên d, ta có thể chọn hai tập khác trống S_{hd} và $S_{-(h-1)d}$. Thật vậy, hai tập này chứa các vị trí bit trong khối Fi mà ta có thể đảo để tăng Sum lên dh và –(h-1)d một cách tương ứng, kết quả cuối cùng là Sum sẽ tăng lên hd+(-(h-1)d) =d;

Tương tự như các tập S_w khác ta cũng có thể coi tập S_0 là tập chứa các vị trí mà khi đảo những bít có vị trí này trên Fi, thì sẽ tăng Sum lên 0. Kết quả này cũng đạt được nếu ta không đảo bất kỳ bit nào trên Fi. Vì vậy, ta có thể coi S_0 là tập trống và khi nói đảo 1 bit có vị trí thuộc tập S_0 có nghĩa là không cần làm gì cả.

Ví dụ:

Giả sử ta có một ma trận ảnh F 8x8 được chia thành 4 ma trận khối ảnh F1, F2, F3, F4 cùng cỡ 4x4, một ma trận khóa K 4x4 và một ma trận cùng cỡ như sau:

Ma trận ảnh F8x8

0	1	0	1	0	1	1	0
1	0	0	0	1	1	1	1
1	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0
1	1	1	0	1	0	0	0
0	0	1	1	1	0	1	0
1	1	0	1	0	1	1	1
1	0	1	1	0	1	1	1

K=

1	1	0	0
0	1	0	0
1	1	1	0
0	0	1	0

W=

1	2	3	4
5	6	7	1
2	3	4	5
6	7	1	2

Trong ví dụ này, ta đã chọn m = n = 4. chọn r=3 ta giấu 12 bit sau B=001010000001 vào trong ảnh F. Như vậy, đoạn bit 001 sẽ được giấu vào khối F1,010 sẽ được giấu vào khối F2, 000 sẽ được giấu vào khối F3, 001 trong F4.

Bước 1: Ta thực hiện phép toán XOR của Fi với K. Kết quả cho trong bảng sau:

F1⊕K **F2⊕K**

1	0	0	1	1	0	1	0
1	1	0	0	1	0	1	1
0	1	1	0	1	1	1	0
0	0	1	0	0	0	0	1
0	0	1	0	0	1	0	0
0	1	1	1	1	1	1	0
0	0	1	1	1	0	0	1
1	0	0	1	0	1	0	1

F3⊕K **F4⊕K**

Bước 2: Ta thực hiện phép nhân từng khối ma trận kết quả trên với ma trận trọng số:

F1⊕K⊗W **F2⊕K⊗W**

1	0	0	1	1	0	1	0
1	1	0	0	1	0	1	1
0	1	1	0	1	1	1	0
0	0	1	0	0	0	0	1
0	0	1	0	0	1	0	0
0	1	1	1	1	1	1	0
0	0	1	1	1	0	0	1
1	0	0	1	0	1	0	1

F3⊕K⊗W **F4⊕K⊗W**

Với F1: Lưu ý rằng ta có 2^3 =8

Tính SUM(**F1**⊕**K**⊗**W**)=0(mod 8). Vì chuỗi ba bit cần giấu đầu tiên là 001 nên ta sẽ phải thay đổi để tăng trọng số lên 1 (d = 1)

Ta xây dựng tập S1:

h=1 : S1 ={(2,2)} $\neq \phi$ ta chọn luôn ô này để đảo bít. Khi đó ma trận khối ảnh F1' là:

F1'

0	1	0	1
1	0	0	1
1	0	0	0
0	0	0	0

*Với F2':

Tính SUM(**F2**⊕**K**⊗**W**)=2(mod8). Và vì chuỗi 3 bit tiếp theo cần giấu là 010=2 nên không cần phải thay đổi F2 nữa.

*Với F3:

Tính SUM(**F3**⊕**K**⊗**W**)= 2(mod 8). Và vì 3 bit tiếp theo cần giấu là 000=0 nên ta cần thay đổi F3 để tăng trọng số lên d=6

Ta cần xây dựng tập S6

h=1: S6 ={(4,4)} $\neq \phi$ ta chọn luôn ô này để đảo bit. Khi đó ma trận khối ảnh là:

F3'

1	1	1	0
0	0	1	1
1	1	0	1
1	0	1	0

*Với F4:

Tính SUM(**F4**$\oplus$**K**$\otimes$**W**)=4 (mod 8). Vì ba bit cần giấu tiếp theo là 001=1 nên ta sẽ thay đổi để tăng trọng số lên 5, (d=5)

Ta xây dựng tập S5:

h=1 : S5=ϕ

h=2: S10=S2={(2,2)}. S(-5)={(1,3),(2,1),(3,2),(3,4)}

Ta chọn đảo bit ở hai ô $[F4]_{2,2}$ và $[F4]_{3,2}$. Khi đó ma trận khối ảnh là:

F4'

1	0	0	0
1	1	1	0
0	0	1	1
0	1	1	1

Ảnh tạo thành ma trận ghép 4 khối điểm ảnh F1', F2', F3', F4'

F1' **F2'**

0	1	0	1	0	1	1	0
1	0	0	1	1	1	1	1
1	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0
1	1	1	0	1	0	0	0
0	0	1	1	1	1	1	0
1	1	0	1	0	0	1	1
1	0	1	0	0	1	1	1

Như vậy ta đã giấ **F3'** in **F4'** ục khối theo thuật toán. Tiếp theo phần sau đây ta sẽ chứng minh tính đúng đắn của thuật toán.

4.3.3 Chứng minh tính đúng đắn của thuật toán

Để chứng minh tính đúng đắn của thuật toán trên, ta sử dụng các bổ đề sau:

Bổ đề 1:

Với mọi w=1,2,…2^r-1 thỏa mãn w$\neq 2^r$-1 ta có:

$$(S_w = \phi) \Rightarrow (S_{2^r-w} \neq \phi)$$

Chứng minh: với w $\neq 2^{r-1}$ giả sử $S_w = \phi$.

Từ định nghĩa của ma trận trọng số ta suy ra tồn tại ít nhất một phần tử W[j,k]=w, do đó ta phải có F_i[i,k]^ K[I,k]=1 vì nếu không khi ta đảo bit F_i[i,k] sẽ được giá trị 1 dẫn đến tổng SUM($\mathbf{F4} \oplus \mathbf{K} \otimes \mathbf{W}$) sẽ tăng lên w và do đó S_w không trống (trái giả thiết).

Do đó, F_i[i,k]^ K[I,k]=1. Nếu ta đảo giá trị của F_i[j,k] thì Sum sẽ giảm đi w hay tăng lên 2^r-w(mod 2^r). $\Rightarrow S_{2^r-w} \neq \phi$

Bổ đề được chứng minh.

Bổ đề 2:

Tập S_w^{r-1} khác rỗng.

Chứng minh:

Từ định nghĩa 1 ta suy ra tồn tại ít nhất một phần tử W[j,k] của ma trận W nhận giá trị 2^{r-1}. Mặt khác ta lại có $2^{r-1} \equiv -2^{r-1}$(mod 2^r) nên nếu ta đảo giá trị của F1[j,k] thì Sum sẽ tăng lên hay giảm đi 2^{r-1} do đó bổ đề được chứng minh.

Bổ đề 3:

Bước 4 luôn luôn được thực hiện và nhiều nhất hai bít của Fi bị đảo để giấu được r bit dữ liệu. Tức là luôn luôn tìm được h sao cho S_{hd} và $S_{-(h-1)d}$ khác trống với mọi d nhận giá trị từ 0 đến 2^{r-1}.

Chứng minh:

Áp dụng định lý số học với mọi cặp số nguyên tố cùng nhau d1, d2, mọi x=1,2…d2-1 luôn tồn tại m,n sao cho md1=nd2+x hay $md1 \equiv x(\text{mod } d2)$ ta suy ra tập hợp {d mod 2^r, 2d mod 2^r,…} chứa tất cả và chỉ chứa các bội nhỏ hơn 2^r của ước số chung lớn nhất của d và 2^r (với d=0,…,2^r-1). Mặt khác 2^{r-1} là bội của ước chung lớn nhất của d và 2^r nên tồn tại một số nguyên k sao cho $kd \equiv 2^{r-1} (\text{mod } 2^r)$ giả sử k là số nguyên nhỏ nhất thỏa mãn điều kiện này.

Ta tính h thỏa mãn $S_{hd} \neq \phi$ và $S_{-(h-1)d} \neq \phi$. Với h=1, nếu $S_d \neq \phi$ thì h=1 là lời giải và bước 4 được thực hiện, nếu không thì $S_{-d} \neq \phi$ (theo bổ đề 1). Với h=2 nếu $S_{2d} \neq \phi$ thì h=2 là lời giải, nếu không thì $S_{-d} \neq \phi$ (theo bổ đề 1)...Tiếp tục như vậy với h=3,4…k-1 nếu vẫn chưa tìm được h thỏa mãn thì ta có thể khẳng định là h=k là lời giải vì khi đó $S_{kd} = S_2^{r-1}$ khác rỗng.

Vậy bổ đề được chứng minh

4.2.4 Độ an toàn của thuật toán

Để đánh giá độ an toàn của kỹ thuật giấu thông tin trong ảnh trình bày ở trên ta giả sử rằng thuật toán là công khai, cũng giả sử thêm rằng ảnh chứa F, giá trị r, và kích thước khối mxn không còn là bí mật. Hơn nữa kẻ địch của ta còn có trong tay cả bản copy của ảnh kết quả F', nhưng chưa biết khóa và ma trận trọng số. Khi đó việc tìm ra thông tin mà ta giấu trong F bằng thuật toán đã trình bày với các tham số này vẫn gần như là không thể được. Thật vậy, ta có thới $t_1=2^{mn}$ khả năng lựa chọn k và gần

$$t_2 = C_{mn}^{2r-1} * (2^r-1)! * (2^r-1)^{mn-(-_2^{r}-1)}$$

Khả năng lựa chọn W và do đó có tới $t_1 * t_2$ cách kết hợp K với W (ta dùng từ gần bởi vì con số này vẫn chưa chính xác hoàn toàn) Khi mxn đủ lớn thì số lựa chọn này có thể làm nản lòng bất kỳ một kẻ tò mò nào.

Trong trường hợp mộtt phần thông tin đã bị lộ và kẻ địch đã biết được hai khối ảnh Fi và Fj và hai khối ảnh tương ứng sau khi đã lần lượt giấu Bi và Bj vào là Fi' và Fj' thì khả năng giải mã được thông tin là có thể xảy ra nếu có thêm một số điều kiện.

Nếu Fi=Fj thì sự khác nhau giữa Bi và Bj sẽ cho biết mối quan hệ của trọng số tại vị trí mà Fi khác Fi' và vị trí Fj khác Fj'. Hơn nữa nếu có thêm rằng Fi=Fi'=Fj và chỉ có một bít tại vị trí (a.b) trong Fj bị đảo, thì khi đó giá trị của W[a,b] là Bj-Bi (mod 2^r). Điều này có thể dễ dàng thấy được nếu ta đặt

d_i =Bi- SUM($\mathbf{Fi} \oplus \mathbf{K} \otimes \mathbf{W}$)(mod 2) =0

d_j =Bj- SUM($\mathbf{Fj} \oplus \mathbf{K} \otimes \mathbf{W}$)(mod 2) =0

Nếu mỗi phần tử của W đều có thể được xác định chỉ được nhận một trong hai giá trị như trên thì khả năng có thể cho W chỉ còn là 2^{mn}. Giảm đi đáng kể so với ban đầu.

Khi ma trận trọng số W đã được xác định thì việc tìm khóa K trở lên dễ hơn. Chẳng hạn như với giả thiết Fi=Fi'=Fj và Fj' khác Fj tại duy nhất một vị trí (a,b) khi đó K[a,b] có thể tính được bằng cách.

*Nếu Bj-Bi=W[a,b] $\neq 2^{r-1}$ thì (Fj $\oplus$ K)[a,b] =0 $\Rightarrow$ K[a,b] =Fj[a,b]

*Nếu không Bj-Bi=-W[a,b] $\neq 2^{r-1}$ thì (Fj $\oplus$ K)[a,b]=1 $\Rightarrow$ K[a,b] = 1-Fj[a,b]

Tóm lại, việc giải mã thông tin càng khó khăn khi kich thước khối mxn đủ lớn và khóa K, ma trận trọng số W được cất giữ an toàn. Nếu coi đây là

một hệ mã mật thì hệ mã mật này có khóa bí mật giống như những hệ mật mã cổ điển. Hiện nay người ta cũng đang nghiên cứu việc dùng khóa công khai.

4.3.5 Phân tích đánh giá thuật toán

Sau khi nghiên cứu thuật toán chúng ta có thể đưa ra một số tổng kết, bình luận và đánh giá sau đây:

-Thuật toán có thể giấu được r bit vào trong một khối mxn với điều kiện là $2^r<m*n$ mà chỉ cần thay đổi nhiều nhất là 2 bit trên một khối. Như vậy, thuật toán này đã có cải tiến rất lớn so với những thuật toán khác chỉ giấu được một bít vào mỗi khối.

-Độ an toàn của thuật toán cũng rất cao thông qua hai ma trận dùng làm khóa để giải tin đó là ma trận trọng số và ma trận khóa

-Đây là thuật toán giấu thông tin trong ảnh đen trắng nên ta có thể áp dụng cải tiến hệ số phân bố bít D để tăng chất lượng ảnh sau khi giấu. Thực tế là nếu kết hợp với phần cải tiến này chất lượng ảnh sẽ được tăng lên đáng kể. Có thể giấu trong các ảnh text.

-Thuật toán này đương nhiên có thể áp dụng cho ảnh màu và ảnh đa cấp xám. Ta cũng có thể sử dụng kỹ thuật chọn ra các bít ít quan trọng nhất của mỗi điểm ảnh để xây dựng ma trận hai chiều các bit 0,1 như trong thuật toán với ảnh đen trắng.

Nếu áp dụng tốt thuật toán này cho ảnh màu thì có thể nói thuật toán đã đạt được những điều cơ bản của một ứng dụng giấu tin mật (steganography) đó là đảm bảo tính ẩn của thông tin giấu, số lượng thông tin giấu cao và độ an toàn của thuật toán cũng cao.

CHƯƠNG 5: THỦY VÂN SỐ TRÊN ẢNH TĨNH

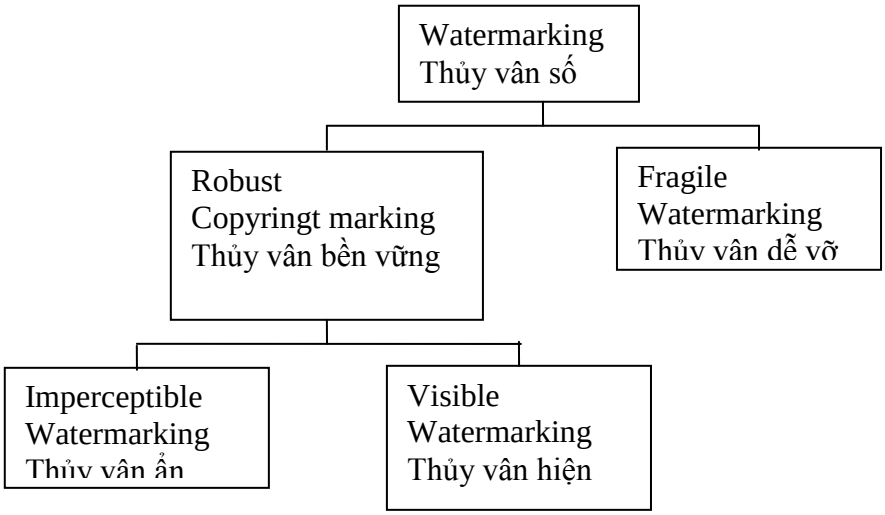
5.1 Giới thiệu chung về kỹ thuật thủy vân

5.1.1 Watermarking và Steganography

Tính năng an toàn và bảo mật thông tin của kỹ thuật giấu tin được thể hiện ở hai khía cạnh. Một là bảo vệ cho dữ liệu đem giấu và hai là bảo vệ cho chính đối tượng được sử dụng để giấu tin. Tương ứng với hai khía cạnh đó chúng ta có hai khuynh hướng kỹ thuật rõ ràng đó là giấu tin mật (steganography) và thủy vân số (watermarking)

Trong kỹ thuật giấu tin mật, thông tin cần giấu được gọi là thông điệp (message) còn trong kỹ thuật thủy vân số thì được gọi là thủy vân (watermark). Thủy vân số có thể là một chuỗi các ký tự, hay một hình ảnh, một logo nào đó.

Nói đến thủy vân số là nói đến kỹ thuật giấu tin nhắm đến những ứng dụng bảo đảm an toàn dữ liệu cho đối tượng được sử dụng để giấu tin như : bảo vệ bản quyền, chống xuyên tạc, nhận thực thông tin, điều khiển sao chép…Có thể thấy rõ là phần ứng dụng của thủy vân rất lớn, mỗi ứng dụng lại có những yêu cầu riêng và tính chất riêng, do đó các kỹ thuật thủy vân cũng có những đặc tính khác biệt tương ứng.



Các kỹ thuật thủy vân được phân biệt nhau bởi những đặc trưng tính chất của từng kỹ thuật và ứng dụng những kỹ thuật đó. Thủy vân "dễ vỡ" (fragile) là kỹ thuật nhúng thủy vân vào trong ảnh sao cho khi phân bố sản phẩm trong môi trường mở nếu có bất cứ một phép biến đổi nào làm thay đổi đối tượng sản phẩm gốc thì thủy vân đã được giấu trong đối tượng sẽ không còn nguyên vẹn như trước khi giấu nữa (dễ vỡ). Các kỹ thuật thủy vân có tính chất này được sử dụng trong các ứng dụng nhận thực thông tin (authentication) và phát hiện xuyên tạc thông tin (tamper detection). Rất rễ hiểu vì sao những ứng dụng này cần đến kỹ thuật thủy vân dễ vỡ. Ví dụ như đề bảo vệ chống xuyên tạc một ảnh nào đó ta nhúng một thủy vân vào trong ảnh và sau đó phân phối, quảng bá ảnh đó. Khi cần kiểm tra lại ảnh ta sử dụng hệ thống đọc thủy vân. Nếu không đọc được thủy vân hoặc thủy vân đã bị sai lệch nhiều so với thủy vân ban đầu đã nhúng vào ảnh thì có nghĩa là ảnh đó đã bị thay đổi. Cái khó ở đây là ta phải phân biệt giữa sai lệch thủy vân so xuyên tạc và sai lệch do lỗi đường truyền. Ngược lại, với kỹ thuật thủy vân dễ vỡ là kỹ thuật thủy vân bền vững (robust). Các kỹ thuật thủy vân bền vững thường được ứng dụng trong các ứng dụng bảo vệ bản quyền. Trong những ứng dụng đó, thủy vân đóng vai trò là thông tin sở hữu của người chủ hợp pháp. Thủy vân được nhúng trong sản phẩm như một hình thức dán tem bản quyền. Trong trường hợp như thế,

thủy vân phải tồn tại bền vững cùng với sản phẩm nhằm chống lại việc tẩy xóa, làm giả hay biến đổi phá hủy thủy vân. Một yêu cầu lý tưởng đối với thủy vân bền vững là nếu muốn loại bỏ thủy vân thì chỉ có một cách duy nhất là phá hủy sản phẩm.

Thủy vân bền vững lại được chia thành hai loại là thủy vân ẩn và thủy vân hiện. Thủy vân hiện là loại thủy vân được hiện ngay trên sản phẩm và người dùng có thể nhìn thấy được giống như các biểu tượng kênh chương trình vô tuyến mà chúng ta thường thấy VTV3, CCTV, TV5…Các thủy vân hiện trên ảnh thường dưới dạng chìm, mờ hoặc trong suốt để không gây ảnh hưởng đến chất lượng ảnh gốc. Đối với thủy vân hiện, thông tin bản quyền hiển thị ngay trên sản phẩm.

Còn đối với thủy vân ẩn thì cũng giống như giấu tin, bằng mắt thường không thể nhìn thấy thủy vân. Trong vấn đề bảo vệ bản quyền, thủy vân ẩn mang tính bất ngờ hơn trong việc phát hiện sản phẩm bị đánh cắp. Trong trường hợp này, người chủ sở hữu hợp pháp sẽ chỉ ra bằng chứng là thủy vân đã được nhúng trong sản phẩm bị đánh cắp.

5.1.2 Các yêu cầu cơ bản của hệ thủy vân trên ảnh

Hệ thủy vân số trên ảnh cũng là một hệ giấu tin nên cũng có một số đặc điểm và tính chất giống như giấu tin trong ảnh như:

-Phương tiện chứa là ảnh hai chiều tĩnh

-Thủy vân trên ảnh tác động lên dữ liệu ảnh nhưng không làm thay đổi kích thước ảnh.

-Kỹ thuật giấu phụ thuộc vào tính chất của hệ thống thị giác con người

-Khi giải tin có thể cần ảnh gốc.

Ngoài một số đặc điểm chung ra, kỹ thuật thủy vân số được phân biệt với kỹ thuật giấu tin mật ở những đặc trưng sau đây:

Thông tin trong ảnh sẽ bị biến đổi nếu có bất cứ một biến đổi nào trên ảnh.

Tính chất này có trong kỹ thuật giấu tin mật nhưng trong kỹ thuật thủy vân thì chỉ có trong thủy vân dễ vỡ. Còn đối với loại thủy vân bền vững thì ta yêu cầu chống lại được những phép biến đổi thông thường trên ảnh.

Thủy vân ẩn hay thủy vân hiện

Không giống như giấu tin mật với yêu cầu là thông điệp giấu phải ẩn bên trong ảnh sao cho mắt thường không nhìn thấy được thì kỹ thuật thủy vân số lại có hai loại là thủy vân ẩn và thủy vân hiện. Nghĩa là có loại thủy vân cho phép nhìn thấy được thông tin đem nhúng vào và có loại thủy vân không nhìn thấy. Loại thủy vân hiện được sử dụng cho mục đích công bố công khai về chủ quyền sở hữu, ngược lại, loại thủy vân ẩn được sử dụng với mục đích gài bí mật các thông tin xác nhận chủ quyền sở hữu.

Tính chất bền vững

Tính chất này là tính chất quan trọng nhất của một hệ thủy vân bền vững. Nghĩa là, hệ thủy vân phải chống lại được các phép biến đổi, hay các tấn công có chủ đích hoặc không có chủ đích lên thủy vân.

Thủy vân cái gì?

Một câu hỏi đầu tiên đối với một hệ thủy vân là thông tin gì sẽ được giấu vào bên trong ảnh? Kiểu thủy vân hay dùng nhất là một chuỗi ký tự, chuỗi ký tự được nhúng trực tiếp lên ảnh mang những thông tin như tác giả, tiêu đề hay ngày tháng, hay thông tin bản quyền…Tuy nhiên sử dụng chuỗi ký tự lại bị

một hạn chế, đó là mỗi ký tự được biểu diễn bằng nhiều bít nếu như vì một lý do nào đó một bit bị lỗi sẽ làm sai cả ký tự và do đó chỉ cần một phép biến đổi đơn giản như phép biến đổi DCT cũng có thể làm cho thủy vân bị sai lệch rất nhiều. Chúng ta cũng có thể dùng ảnh để giấu, khi đó ta sẽ có ảnh trong ảnh. Khi giải tin một số điểm ảnh có thể bị sai nhưng hình tổng thể có thể giữ nguyên. Trong những kỹ thuật gần đây, người ta sử dụng thủy vân là một chuỗi bit sinh ngẫu nhiên theo một luật phân phối xác suất nào đó. Và sau đó áp dụng các lý thuyết xác suất thống kê để chứng thực thủy vân.

Trong các loại thủy vân thì thủy vân ẩn và bền vững là loại được quan tâm nghiên cứu nhiều nhất vì ý nghĩa ứng dụng lớn của nó như đã nói ở phần trên. Do vậy, hai tính chất quan trọng của hệ thủy vân mà các nhà nghiên cứu đang cố gắng đạt được là thuộc tính ẩn và thuộc tính bền vững. Nhưng đây lại là mấu chốt của sự phức tạp vì hai thuộc tính này mâu thuẫn với nhau. Nếu như để đảm bảo được thuộc tính ẩn, thủy vân phải được giấu ở những vị trí có ít ý nghĩa tri giác nhất, ít bị chú ý nhất thì để đảm bảo thuộc tính bền vững, thủy vân phải chịu được những phép xử lý ảnh phổ biến như dịch chuyển hay nén JPEG. Ví dụ như nén JPEG loại bỏ ở ảnh những thông tin có ít ý nghĩa tri giác để làm giảm kích thước của ảnh mà vẫn đảm bảo được chất lượng của ảnh. Khi đó thì những dữ liệu của thủy vân nằm trong vùng này sẽ bị mất đi hoặc bị biến đổi sai lệch hoàn toàn. Với tính phức tạp của yêu cầu cho một hệ thủy vân, phần sau đây ta sẽ tìm hiểu những giải pháp kỹ thuật đã được đưa ra của các nhà khoa học trên thế giới.

5.1.3 Những tấn công trên hệ thủy vân

Phương pháp thủy vân nên chống lại được một số phép xử lý ảnh thông thường và một số tấn công có chủ đích. Cho đến nay vẫn chưa có một hệ thống hoàn hảo và cũng không rõ ràng việc liệu có tồn tại hay không một hệ thống thủy vân an toàn tuyệt đối. Vì vậy, trong thực tế thì thủy vân phải cân

nhắc giữa bền vững với các thuộc tính khác như lượng thông tin giấu, tính ẩn… Dựa vào yêu cầu của ứng dụng mà sẽ ảnh hưởng đến phương pháp thủy vân. Dựa vào những biến đổi có chủ đích hay không có chủ đích với hệ thủy vân mà ta có thể phân biệt thành hai nhóm xuyên tạc sau: một là biến đổi được xem như là nhiễu đối với dữ liệu, hai là làm mất tính đồng bộ để không thể lấy tin ra được.

-Biến đổi tín hiệu: làm sắc, biến đổi tương phản, màu, gamma…

-Nhiễu cộng, nhiễu nhân…

-Lọc tuyến tính

-Nén mất thông tin

-Biến đổi affine cục bộ hoặc toàn cục

-Giảm dữ liệu: cropping, sửa histogram

-Chuyển mã (gif→ jpeg)

-Chuyển đổi tương tự → số

-Thủy vân nhiều lần

Nguyên tắc cơ bản của phương pháp thủy vân là đảm bảo đủ tính bền vững sao cho các tấn công sẽ làm cho giá trị thương mại của ảnh gốc sẽ bị ảnh hưởng.

5.2 Những khuynh hướng tiếp cận thủy vân

5.2.1 Hướng tiếp cần dựa trên miềm không gian ảnh

Đây là hướng tiếp cận cơ bản và tự nhiên trong số các kỹ thuật thủy vân. Miền không gian ảnh là miền dữ liệu ảnh gốc, tác động lên miền không gian ảnh chính là tác động lên các điểm ảnh, thay đổi giá trị trực tiếp của điểm ảnh. Đây là hướng tiếp cận tự nhiên bởi lẽ khi nói đến việc giấu tin trong ảnh người ta thường nghĩ ngay đến việc thay đổi giá trị các điểm ảnh nguồn. Một phương pháp phổ biến của cách tiếp cận này là phương pháp thay thế bít ít quan trọng nhất của mỗi điểm ảnh.

Ý tưởng cơ bản của phương pháp thay thế bit ít quan trọng nhất LSB (Least Significant Bit) là chọn ra từ mỗi điểm ảnh các bít có ít ý nghĩa tri giác nhất để sử dụng cho việc giấu tin. Bít nào được coi là ít tri giác nhất và bao nhiêu bít có thể được lấy ra để thay thế thì phụ thuộc vào tính chất hệ thống thị giác con người và phụ thuộc vào nhu cầu chất lượng ảnh trong các ứng dụng. ví dụ, trong ảnh 24 bit màu, mỗi màu được biểu diễn bởi 24 bit tương ứng với 3 màu RGB, mỗi màu chiếm 1 byte. Người ta sử dụng một tính chất của mắt người là sự cảm nhận về màu B (blue) kém hơn so với hai màu RG. Chính vì thế mà người ta thường chọn bít cuối cùng trong 8 bit biểu diễn màu B của mỗi điểm ảnh để giấu tin. Thay đổi bít cuối cùng trong 8 bit biểu diễn màu B chỉ làm cho giá trị biểu diễn màu B tăng hoặc giảm đi 1 đơn vị. Do vậy, các bít ít quan trọng nhất trong trường hợp này là bít thứ 24 của mỗi điểm ảnh. Một thuật toán muốn giấu nhiều hơn và chất lượng ảnh thấp hơn một chút có thể sử dụng bít cuối cùng của mỗi byte biểu diễn mỗi màu RGB làm bít ít quan trọng nhất. Trong trường hợp này thì mỗi điểm ảnh sẽ chọn ra được 3 bit LSB.

Tuy nhiên, phương pháp này cũng có nhiều hạn chế như không đảm bảo được tính bền vững của thủy vân đối với các thao tác như quay ảnh hay nén ảnh jpeg chẳng hạn. Điều này là dễ hiểu vì các thao tác nói trên cũng loại bỏ hoặc làm sai lệch các bít ít quan trọng nhất.

5.2.2 Hướng tiếp cận dựa trên miền tần số của ảnh

Hướng tiếp cận dựa trên miền không gian ảnh như đã trình bày ở trên là cách tiến hành khảo sát tín hiệu và hệ thống rời rạc một cách trực tiếp trên miền giá trị rời rạc của các điểm ảnh gọi là trên miền biến số độc lập tự nhiên. Nhưng trong nhiều trường hợp, cách khảo sát trực tiếp này gặp phải những khó khăn nhất định hoặc rất phức tạp và hiệu quả không cao.

Ngoài phương pháp khảo sát trực tiếp này chúng ta có thể dùng nhiều phương pháp khảo sát gián tiếp khác thông qua các kỹ thuật biến đổi. Các biến đổi này làm nhiệm vụ chuyển miền biến số độc lập sang các miền khác và như vậy tín hiệu và hệ thống rời rạc sẽ được biểu diễn trong miền mới này với các biến số mới. Phương pháp biến đổi này cũng giống như phương pháp đổi biến trong tích phân hay phương pháp đổi hệ tọa độ trong toán giải tích của toán phổ thông quen thuộc.

Mỗi một cách biến đổi có những thuận lợi riêng, tùy từng trường hợp mà chúng ta dùng biến đổi nào. Sau khi khảo sát xong các tín hiệu và hệ thống rời rạc trong các miền biến số mới này nếu cần thiết chúng ta sẽ dùng các biến đổi ngược để đưa chúng về miền biến số độc lập cũ.

Phương pháp khảo sát gián tiếp này sẽ làm đơn giản rất nhiều công việc mà chúng ta gặp phải khi dùng phương pháp khảo sát trực tiếp trong miền biến số độc lập tự nhiên. Đối với chúng ta, hệ thống rời rạc cần khảo sát chính là miền không gian các điểm ảnh, có nhiều phép biến đổi cho dữ liệu ảnh trong đó có một số phương pháp biến đổi được sử dụng rất phổ biến như

Fourier, biến đổi cosin rời rạc, Wavelet…Đây là những phép biến đổi được sử dụng nhiều trong các kỹ thuật xử lý ảnh.

Trước hết ta khảo sát một số phép biến đổi đang được ứng dụng nhiều trong kỹ thuật thủy vân.

5.3 Một số kỹ thuật bổ trợ cho các kỹ thuật thủy vân số trên ảnh tĩnh

Phương pháp thủy vân số là một phương pháp mới và rất phức tạp, có thể nói việc nghiên cứu vẫn đang diễn ra và đang được các nhà nghiên cứu dần hình thành khung lý thuyết cho nó. Nhưng cho đến nay những kỹ thuật đưa ra cũng chỉ là những thử nghiệm, lúc thì người ta dùng các công cụ lý thuyết mật mã học, lúc thì kỹ thuật truyền thông, khi lại sử dụng lý thuyết thông tin…cho nên những kỹ thuật thủy vân cũng hết sức phong phú. Và như vậy, khi làm về thủy vân ta phải biết nhiều kỹ thuật ở nhiều lĩnh vực lý thuyết khác nhau. Tuy nhiên, qua khảo sát gần đây của giáo sư Deepa Kunder của trường đại học Toronto có hai khuynh hướng chủ yếu đã được hình thành đó là khuynh hướng sử dụng lý thuyết thông tin và lý thuyết truyền thông.

Theo giáo sư, khuynh hướng lý thuyết truyền thông thực tế hơn so với lý thuyết thông tin, và có thể sử dụng dễ dàng hơn trong thiết kế thuật toán. Có một vài sự khác biệt đặc trưng giữa hai khuynh hướng này. Kỹ thuật thủy vân dựa trên lý thuyết truyền thông thường sử dụng những cơ sở lý thuyết trong truyền thông để thiết kế như việc dùng lý thuyết phân tích thống kê để tạo thủy vân và kiểm định thủy vân lấy ra so với thủy vân được nhúng vào, kỹ thuật trải phổ tín hiệu để truyền tin hay kỹ thuật tạo nhiễu cộng và lọc nhiễu. Trong khi đó, khuynh hướng dùng lý thuyết thông tin lại sử dụng những cơ sở phân tích chung để phân tích làm sao thu được hiệu suất cao nhất, chiến lược tốt nhất cho một thuật toán cụ thể hay khả năng chịu tấn công đối với một kỹ thuật thủy vân. Một sự khác biệt nữa trong hai khuynh hướng kỹ thuật thủy

vân này là sự đánh giá hệ thống thủy vân. Đối với khuynh hướng sử dụng lý thuyết truyền thông thì thường nhận biết thủy vân và đánh giá hệ thống thủy vân thông qua độ bền vững của thủy vân trước và sau khi giấu bằng phép đo hệ số tương quan giữa thủy vân được nhúng vào và thủy vân được lấy ra hay tỉ lệ bit lỗi (BER-Bit Error Rate). Còn những kỹ thuật thủy vân theo khuynh hướng lý thuyết thông tin thì chủ yếu hệ thống được đánh giá thông qua khả năng giấu. Nghĩa là tổng số bít có thể được nhúng vào và được lấy ra một cách đáng tin cậy.

Để giúp cho nghiên cứu và và có thể cài đặt nhanh chóng các thuật toán thủy vân. Chúng ta sẽ tìm hiểu một số kỹ thuật thủy vân theo hai khuynh hướng trên.

5.3.1 Các phép biến đổi miền không gian ảnh sang miền tần số.

Để khảo sát hệ thống rời rạc, trong nhiều trường hợp, chúng ta thường biến đổi hệ thống rời rạc đó sang một miền biến số khác. Có nhiều phép biến đổi khác nhau như biến đổi tín hiệu và hệ thống rời rạc sang miền Z, biến đổi sang miền tần số liên tục hay sang miền tần số rời rạc. Mỗi phép biến đổi có những thuận lợi riêng, tùy theo yêu cầu khảo sát mà ta sẽ lựa chọn phép biến đổi phù hợp. Trong trường hợp khảo sát miền không gian ảnh người ta thường biến đổi miền không gian rời rạc tín hiệu điểm ảnh sang miền tần số rời rạc bằng các phép biến đổi như Fourier, Cosin rời rạc hay wavelet (sóng lăn)…Các phép biến đổi này khá phức tạp về ý nghĩa cũng như cài đặt.

5.3.1.1 Phép biến đổi Fourier rời rạc.

Phép biến đổi Fourier rời rạc viết tắt là DFT (Discrete Fourier Transform) là một công cụ toán học được dùng để chuyển cách biểu diễn tín hiệu và hệ thống rời rạc hoặc liên tục sang miền tần số rời rạc. Thực chất của cách biểu diễn này là lấy từng điểm rời rạc trên vòng tròn đơn vị trong mặt

phẳng Z để biểu diễn. Việc biểu diễn trong miền tần số rời rạc đặc biệt hiệu quả khi xuất hiện các thuật toán tính toán nhanh DFT ta gọi là phép biến đổi Fourier nhanh FFT(Fast Fourier Transform).

Định nghĩa phép biến đổi Fourier rời rạc cho tín hiệu hai chiều (ảnh số)

Biến đổi Fourier rời rạc của một ảnh MxN:{u(m,n)} được định nghĩa như sau:

$$v(k,l)= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} u((m,n) w_N^{km} w_N^{ln}$$

Với 0<=l, k<=N-1

Và biến đổi ngược:

$$u(m,n)= \frac{1}{N}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} v(k,l) w_N^{-km} w_N^{-ln}$$

Với 0<=m,n<=N-1

5.3.1.2 Phép biến đổi cosin rời rạc

Biến đổi cosin rời rạc viết tắt là DCT (Discrete Cosin Transform) được đưa ra bởi Ahmed và các đồng nghiệp của ông vào năm 1974. Từ đó cho đến nay, nó được sử dụng rất phổ biến trong nhiều kỹ thuật xử lý ảnh số nói riêng và xử lý tín hiệu số nói chung. Trong các kỹ thuật thủy vân ảnh dựa trên phép biến đổi dữ liệu ảnh sang miền tần số cho phép biến đổi DCT cũng được sử dụng nhiều nhất. Lý do ở đây là phép biến đổi DCT đã được dùng trong dạng chuẩn ảnh JPEG. Nếu áp dụng DCT thì cũng theo chuẩn của JPEG và do đó sẽ tránh được mất thủy vân do phép nén JPEG.

Định nghĩa biến đổi cosin rởi rạc hai chiều:

Biến đổi DCT hai chiều tổng quát là biến đổi trên khối hai chiều bất kỳ M*N, trong đó các khối kích thước 8x8, 16x16 được sử dụng nhiều nhất. Tuy nhiên, chúng ta sẽ tìm hiểu phép biến đổi DCT trên khối 8x8 được sử dụng trong chuẩn nén ảnh JPG.

Phép biến đổi thuận DCT 8x8 được định nghĩa như sau:

$$I(u,v) = \frac{\zeta(u)\zeta(v)}{4}\sum_{k=0}^{7}\sum_{l=0}^{7} X(k,l)\cos(\frac{(2k+1)u\pi}{16})sos\frac{(2l+1)v\pi}{16}$$

I(u,v) được gọi là hệ số DCT và là số thực

Còn biến đổi ngược IDCT được định nghĩa như sau:

$$X(k,l)= \frac{\zeta(u)\zeta(v)}{4}\sum_{u=0}^{7}\sum_{v=0}^{7}\frac{\zeta(u)\zeta(v)}{4} I(u,v)\cos(\frac{(2k+1)u\pi}{16})sos\frac{(2l+1)v\pi}{16}$$

Ở đây $0 \le k,l,u,v \le 7$ và $\zeta(u) = \begin{cases} \frac{1}{\sqrt{2}} & u > 0 \\ 1 & u = 0 \end{cases}$

$$\zeta(v) = \begin{cases} \frac{1}{\sqrt{2}} & v > 0 \\ 1 & v = 0 \end{cases}$$

Đặc điểm của phép biến đổi DCT trên ảnh hai chiều:

-Thể hiện về đặc tính nội dung về tần số của thông tin ảnh. Hệ số góc trên là số lớn đặc trưng cho giá trị trung bình, thành phần một chiều gọi là hệ số DC, còn các hệ số khác có giá trị nhỏ hơn biểu diễn cho các thành phần tần số cao theo hướng ngang và theo hướng thẳng đứng gọi là các hệ số AC.

-Bản thân biến đổi DCT không nén được dữ liệu vì cũng sinh ra 64 hệ số.

-Theo nguyên lý chung, khi biến đổi chi tiết giữa các điểm ảnh càng lớn theo một hướng nào đó trong khối các điểm ảnh, hướng ngang, hướng thẳng đứng hay theo hướng chéo, thì tương ứng theo các hướng đó, các hệ số biến đổi DCT cũng lớn.

-Tóm lại, DCT làm giảm độ tương quan không gian của thông tin trong khối ảnh. Điều đó cho phép biểu diễn thích hợp ở miền DCT có các hệ số DCT có xu hướng có phần dư thừa ít hơn. Hơn nữa, các hệ số DCT chứa thông tin về nội dung tần số không gian của thông tin trong khối. Nhờ các đặc tình tần số không gian của hệ thống nhìn của mắt người, các hệ số DCT có thể được mã hóa phù hợp, chỉ các hệ số DCT quan trọng nhất mới được mã hóa để truyền đi.

-Khối hệ số DCT có thể chia thành 3 miền, miền tần số thấp, chứa các thông tin quan trọng ảnh hưởng đến tri giác, miền tần số giữa và miền tần số cao. Các thông tin trong miền tần số cao thường không mang tính tri giác cao, khi nén JPEG thì thường loại bỏ thông tin trong miền này.

Trong các thuật toán thủy vân, miền hệ số DCT tần số cao thường không được sử dụng do nó thường không bền vững với các phép xử lý ảnh; hoặc nén ảnh JPEG. Miền tần số thấp cũng khó được sử dụng do một sự thay đổi dù nhỏ trong miền này cũng dẫn đến chất lượng tri giác của ảnh. Vì vậy, miền tần số ở giữa hay được sử dụng nhiều nhất và cũng cho kết quả tốt nhất. Trong thuật toán đề xuất cũng sử dụng miền tần số ở giữa.

5.3.1.3 Phép biến đổi sóng lăn (Wavelet)

Đây là phép biến đổi mới nhất áp dụng cho ảnh số. Ý tưởng của DWT cho tín hiệu một chiều như sau: Tín hiệu được chia thành hai phần, phần tần số cao và phần tần số thấp. Hầu hết năng lượng được tập trung ở phần góc cạnh hoặc có kết cấu và thuộc thành phần có tần số cao. Thành phần có tần số thấp lại được chia thành hai phần có tần số cao và tần số thấp. Với các bài toán nén và thủy vân ta chỉ cần áp dụng không quá 5 lần bước phân chia trên. Ngoài ra, từ các hệ số DWT, ta có thể tạo lại ảnh ban đầu bằng quá trình DWT ngược hay IDWT.

Ta có thể mô tả bằng toán học DWT và IDWT như sau:

$$H(\omega) = \sum_k h_k e^{-jkw}$$

$$\text{Và } G(\omega) = \sum_k g_k e^{-jkw}$$

Là lọc thông thấp và lọc thông cao tương ứng, mà thỏa mãn một vài điều kiện cho việc tái xây dựng ảnh ban đầu. Một tín hiệu F(n) có thể được phân tích đệ quy như sau:

$$f_{i-1}^{low}(k) = \sum_n h_{n-2k} f_j(n)$$

$$\text{Và } f_{i-1}^{high}(k) = \sum_n g_{n-2k} f_j(n)$$

Với j=J+1,J,…J_0 với $f_{J+1}(k)=F(j),k \in Z$, J+1 là chỉ số mức phân giải cao còn J_0 là chỉ số mức phân giải thấp. Các hệ số

$f_{j0}^{low}(k).\ f_{j0}^{high}(k).\ f_{j0+1}^{low}(k).\ f_{j0+1}^{high}(k) \ldots f_j^{high}(k)$ được gọi là các hệ số của tín hiệu F(n), với $f_{j0}^{low}(k)$ là phần phân giải nhỏ nhất (xấp xỉ) của F(n) và $f_j^{high}(k)$

là phần chi tiết của F(n) tại các giải tần khác nhau. Tín hiệu ban đầu F(n) có thể được xây dựng lại từ các hệ số DWT bằng cách đệ quy như sau:

$$f_j^{low}(n) = \sum_k h_{n-2k} f_{j-1}^k + \sum_k g_{n-2k} f_{j-1}^{high}(k)$$

Để đảm bảo quan hệ giữa DWT và IDWT thì $H(\omega)$ và $G(\omega)$ phải thỏa mãn điều kiện trực giao sau: $|H(\omega)|^2 + |G(\omega)|^2 = 1$

Biến đổi DWT và IDWT cho mảng hai chiều MxN có thể được định nghĩa tương tự bằng cách thực hiện các biến đổi một chiều DWT và IDWT cho mỗi chiều tương ứng.

Biến đổi sóng có rất nhiều lợi thế so với các biến đổi khác, đó chính là:

-Biến đổi sóng lăn là một mô tả đa độ phân giải của ảnh. Quá trình giải mã có thể được xử lý tuần tự từ độ phân giải thấp cho đến độ phân giải cao.

-Biến đổi DWT gần gũi với hệ thống thị giác người hơn biến đổi DCT vì vậy, có thể nén với tỉ lệ cao bằng DWT mà sự biến đổi ảnh khó nhận thấy hơn nếu dùng DCT với tỉ lệ tương tự.

Biến đổi sóng tạo ra một cấu trúc được gọi là biểu diễn tỉ lệ không gian (scale-space representation). Trong biểu diễn này, các tín hiệu tần số cao được xác định chính xác trong miền điểm ảnh (pixel), còn các tín hiệu tần số thấp được xác định chính xác trong miền tần số.

5.3.2 Kỹ thuật sinh chuỗi giả ngẫu nhiên

Như đã trình bày ở phần trên, thủy vân có thể là ảnh, text hay một chuỗi bit được sinh ngẫu nhiên. Kỹ thuật sinh chuỗi giả ngẫu nhiên thường được sử dụng để tạo thủy vân dựa trên phương pháp thống kê. Tại sao lại là giả ngẫu nhiên (pseudo-random)? Vì không có cách nào để tạo ra các chuỗi

ngẫu nhiên thực sự từ một máy vi tính. Một khi chương trình đó chúng ta viết ra, thì chắc chắn số nó tạo ra có thể suy luận được. Phương pháp tốt nhất chúng ta hy vọng là viết các chương trình để tạo ra các chuỗi số có được nhiều thuộc tính giống như các số ngẫu nhiên.

5.3.3 Các kỹ thuật trải phổ trong truyền thông

Một điều hết sức kỳ diệu của tư duy đã được ứng dụng trong kỹ thuật giấu tin. Chúng ta đã biết giấu tin là kỹ thuật nhúng một lượng thông tin số nào đó vào trong một đối tượng thông tin số khác. Và những người nghiên cứu đã liên tưởng ngay đến một kỹ thuật trong truyền thông cũng có những thao tác tương tự và người ta đã áp dụng thành công ý tưởng đó. Kỹ thuật trải phổ trong truyền thông (spread-spectrum communication) có thể được mô tả một cách sơ lược như sau:

Từ một máy phát A muốn truyền một thông tin M trên một kênh truyền đến máy thu B, người ta chia thông tin M ra thành n gói thông tin nhỏ {s1,s2,…sn}, trước khi đưa lên kênh truyền dẫn mỗi gói tin nhỏ s_i được trải phổ bằng một mã trải phổ giả nhiễu. Mã trải phổ giả nhiễu này phải được xác định và cung cấp cho bên thu để bên thu nén phổ để lấy tin ra. Kết quả của việc trải phổ là phổ của tín hiệu được trải rộng ra gấp hàng trăm lần so với ban đầu và mật độ năng lượng phổ cũng thấp xuống làm cho giống nhiễu. Công việc này có một số ích lợi sau đây:

-Thứ nhất, thông tin thường có giải tần thấp dễ bị giao thoa với sóng khác.

-Thứ hai, đảm bảo độ an toàn truyền tin tránh bị các máy thu khác không chủ đích thu được tín hiệu.

-Thứ ba, trải phổ có tác dụng nhiều người dùng chung một giải băng tần.

Đến đầu thu, nhờ có mã giả nhiễu, máy thu sẽ thực hiện việc đồng bộ hóa. Việc đồng bộ hóa bao gồm hai giai đoạn đó là giai đoạn bắt chuỗi và bám chuỗi để tìm ra đúng pha của mã trải phổ giả ngẫu nhiên. Sau khi tìm được đúng mã trải phổ giả ngẫu nhiên thì thực hiện công việc nén phổ để thu được gói thông tin ban đầu. Các gói thông tin lại được kết hợp với nhau để thu được thông điệp M

Bây giờ ta hãy đặt bài toán giấu tin dưới góc nhìn của truyền thông. Các yêu cầu chung nhất đối với thủy vân số đó là thuộc tính ẩn và thuộc tính bến vững, nhưng hai thuộc tính này như có một cái gì đó mâu thuẫn nhau. Thuộc tính ẩn có nghĩa là nói đến những tín hiệu thủy vân phải có năng lượng nhỏ để tránh được những tri giác bình thường trong khi đó thuộc tính bền vững lại nói đên các tín hiệu phải đủ lớn để có thể phát hiện ra sự tồn tại của thủy vân và lấy ra được từ nguồn chứa. Dưới những điều kiện này thì ban đầu lý thuyết truyền thông trải phổ (spread spectrum) là một cách thích hợp nhất cho thủy vân số vì nó sẽ trải rộng tín hiệu thủy vân với một biên độ thấp nhưng băng thông đủ rộng để có thể nắm được năng lượng của các tín hiệu dành cho việc phát hiện thủy vân.

Ta có thể coi quá trình truyền đối tượng đã được nhúng thủy vân dưới sự tác động của các tấn công bên ngoài cũng giống như truyền dữ liệu trong môi trường không tin cậy. Tiến trình nhúng thủy vân cũng giống như tiến trình mã kênh (chanel coding). Giải mã để lấy thủy vân cũng giống như tiến trình xử lý ở bên nhận thông tin trong một phiên truyền thông.

Nhiều kỹ thuật và công cụ để nâng cao truyền thông cũng có thể được áp dụng để nâng cao độ bền vững của thủy vân. Sẽ rất thuận tiện nếu ta sử dụng

những lý thuyết truyền thông để áp dụng cho thủy vân số. Mặc dù những lý thuyết này chủ yếu chỉ nhằm vào tính bền vững của thủy vân và như vậy nghĩa là không đầy đủ. Tuy nhiên, nó rất hữu ích trong việc thiết kế và đánh giá thuật toán cho watermarking.

5.3.4 Các thuật toán kiểm định thủy vân

Đây là kỹ thuật con được sử dụng sau cùng trong kỹ thuật thủy vân. Thủy vân được nhúng sau khi giải mã sẽ được so sánh để kiểm định, chứng thực thủy vân. Có những thủy vân nhìn thấy được và mang ý nghĩa nhận biết thì công việc trở nên quá đơn giản chẳng hạn như thủy vân là một chuỗi mã ASCII mang thông tin nào đó như tên tác giả, ngày tháng…thì khi giải mã cũng dễ dàng nhận biết thông tin. Hay như thủy vân là một ảnh nào đó chẳng hạn thì giải mã ra cũng được một cái ảnh tương tự và ta có thể nhìn thấy sự khác biệt giữa hai ảnh.

Tuy nhiên, trong một số trường hợp thì thủy vân là một chuỗi bít nào đấy, thủy vân chuỗi bít mang ý nghĩa thống kê nên nó cũng thường được sử dụng. Vậy thì khi đó công việc nhận diện thủy vân sẽ không đơn giản. Hoặc ngay cả trong trường hợp thủy vân là những thông tin mang ý nghĩa nhận biết được thì cũng phải có kỹ thuật để kiểm định lượng thủy vân. Kỹ thuật đơn giản nhất là tính tỉ lệ đúng sai từng bít, chẳng hạn ta nhúng một thủy vân có độ dài là 1000 bit, khi giải mã thủy vân bị sai lệch mất 1 bit so với ban đầu vậy thì tỉ lệ sai là $1/1000=10^{-3}$

CHƯƠNG 6: GIỚI THIỆU MỘT SỐ KỸ THUẬT THỦY VÂN TRÊN ẢNH

Ta đã tìm hiểu một số khái niệm cơ bản của một hệ thống thủy vân. Miền ứng dụng của thủy vân rất lớn và mỗi ứng dụng có những yêu cầu về tính năng khác nhau. Nghệ thuật của thủy vân số chính là lựa chọn công nghệ tùy theo ứng dụng. Ta phải cân bằng giữa các yếu tố. Có nhiều các yếu tố khác nhau liên quan đến việc lựa chọn giải pháp cho thủy vân số như là: bảo mật, yếu tố tri giác, độ bền vững, độ phức tạp… Trong chương này, chúng ta sẽ tìm hiểu một số kỹ thuật thủy vân trên ảnh, những kỹ thuật này chủ yếu nhằm vào ứng dụng bảo vệ bản quyền ảnh.

6.1 Một số kỹ thuật thuỷ vân trên miền tần số

6.1.1 Kỹ thuật 1

Thuật toán dưới đây sẽ sử dụng phương pháp nhúng thủy vân trong miền tần số của ảnh, giải tần được sử dụng để chứa tín hiệu thủy vân là miền tần số ở giữa của môt khối DCT 8x8. Trong đó, các khối DCT 8x8 là những khối ảnh cùng kích thước đã được chọn ra ngẫu nhiên từ ảnh ban đầu và được áp dụng biến đổi Cosin rời rạc DCT để chuyển sang miền tần số. Mỗi tín hiệu thủy vân sẽ được chứa trong một khối.

6.1.1.1. Mô tả thuật toán

− Input:
- Một chuỗi các bít thể hiện bản quyền
- Một ảnh

− Output:
- Một ảnh sau khi thuỷ vân.
- Khoá để giải mã.

6.1.1.2. Quá trình Watermarking

− Chia ảnh có kích thước m × n thành (m×n)/64 khối 8×8, mỗi bít sẽ được giấu trong một khối.

− Chọn một khối bất kì B và biến đổi DCT khối đó thu được B'

− Chọn hai hệ số ở vị trí bất kì trong miền tần số ở giữa của khối DCT, giả sử đó là *b'(i,j)* và *b'(p,q)*.

Ta tính:

$$d = || b'(i,j)|- |b'(p,q)|| \ mod \ a$$

trong đó *a* là một tham số thoả mãn điều kiện: *a=2(2t+1)*, *t* là một số nguyên dương.

Bít s_i sẽ được nhúng sao cho thoả mãn điều kiện sau:

$d >= 2t+1$ nếu $s_i = 1$

$d < 2t+1$ nếu $s_i = 0$

− Nếu $d<2t+1$ và $s_i = 1$ thì một trong hai hệ số DCT *b'(i,j)* hoặc *b'(p,q)* có trị tuyệt đối lớn hơn sẽ bị thay đổi để $d>=2t+1$ theo công thức sau:

$$max(|b'(i,j)|, |b'(p,q)|) + (INT(0,75 *a) - d)$$

Với hàm *max(|b'(i,j)|, |b'(p,q)|)* là hàm chọn ra hệ số có trị tuyệt đối lớn hơn, hệ số được chọn sẽ được cộng thêm một lượng là *(INT(0,75 *a) - d)*.

Hoặc cũng có thể biến đổi một trong hai hệ số theo công thức:

$$min(|b'(i,j)|, |b'(p,q)|) - (INT(0,25 *a) + d)$$

Với hàm *min(|b'(i,j)|, |b'(p,q)|)* là hàm chọn ra hệ số có trị tuyệt đối nhỏ hơn, hệ số được chọn sẽ bị trừ đi một lượng là *(INT(0,25 *a) + d)*

INT() là hàm làm lấy phần nguyên của một số thực.

− Tương tự, nếu $d >= 2t+1$ và $s_i = 0$ thì một trong hai hệ số DCT *b'(i,j)* hoặc *b'(p,q)* có trị tuyệt đối lớn hơn sẽ được thay đổi để thoả mãn $d<2t+1$ như sau:

$$max(|b'(i,j)|,|b'(p,q)|) - (d - INT(0,25*a))$$

Hàm *max(|b'(i,j)|, |b'(p,q)|)* là hàm chọn ra hệ số có trị tuyệt đối lớn hơn, hệ số được chọn sẽ bị trừ đi một lượng là *(d - INT(0,25 *a))*

Hoặc

$$min(|b'(i,j))|,|b'(p,q)|) + INT(1,25*a) - d$$

6.1.1.3. Quá trình giải nhúng để lấy lại thông tin:

Đọc khối DCT từ ảnh chứa thuỷ vân và vị trí hai hệ số đã biến đổi, sau đó tính:

$$d = ||b'(i,j)|-|b'(p,q)|| \ mod \ a \text{ với } (a = 2(2t+1))$$

Nếu $d >= 2t+1$ thì gán $s_i = 1$

Nếu $d < 2t+1$ thì gán $s_i = 0$

6.1.1.4. Chứng minh tính đúng đắn của thuật toán.

Xét các trường hợp sau đây:

− Nếu $d < 2t+1$ với $s_i = 0$ và $d >= 2t+1$ với $s_i = 1$ thì sẽ không thay đổi gì hệ số của khối DCT, và vì DCT là phép biến đổi thuận nghịch nên khi giải mã thì ta cũng thu được kết quả chính xác.

− Trường hợp $d < 2t+1$ và $s_i = 1$.

Ta biến đổi một trong hai hệ số *b'(i,j)* và *b'(p,q)* như sau:

$$max(|b'(i,j)|,|b'(p,q)|) + (INT(0,75*a) - d)$$

Khi đó giá trị *d* mới là:

$$d' = (||b'(i,j)|- |b'(p,q)||)+ (INT(0,75*a) – d))mod \ a)$$

- $d' = (||b'(i,j)|- |b'(p,q)|| \ mod \ a)+ (INT(0,75*a)mod \ a) - (d \ mod \ a)$
- $d' = d + INT(0,75*a) - d = INT(0,75 *a)> 0,5 *a = 2t +1$ (dpcm)

Hoặc ta sử dụng cách biến đổi hai hệ số theo kiểu khác:

$$min(|b'(i,j)| - |b'(p,q)|) - (INT(0,25 *a) +d))$$

Tính lại d:

*d' = (||b'(i,j)|-|b'(p,q)|| - (INT(0,25*a) +d)) mod a*

- *d' = (||b'(i,j)|- |b'(p,q)|| mod a)- (INT(0,25*a) mod a) - (d mod a)*
- *d'= d-(INT(0,25*a) mod a)-d=-INT(0,25*a) mod a = INT(0,75*a) >2t +1*

− Trường hợp *d>=2t +1* và $s_i = 0$

Ta sẽ biến đổi một trong hai hệ số DCT *b'(i,j)* hoặc *b'(p,q)* như sau:

*max(||b'(i,j)|,|b'(p,q)||) - (d- INT(0,25*a))*

Giá trị mới của *d* sẽ là:

*d' = (||b'(i,j)|-|b'(p,q)|| - (d-(INT(0,25*a)) mod a*

- *d'= ((||b'(i,j)|-|b'(p,q)||) mod a) - (d mod a)+ (INT(0,25*a)mod a)*
- *d'= d-d + 0,25*a = 0,25*a < 0,5*a = 2t +1(dpcm)*

Hoặc ta sử dụng cách biến đổi khác đối với hai hệ số DCT:

*min(||b'(i,j)|- |b'(p,q)||) + INT(1,25*a) - d*

Khi đó tính lại *d* ta được:

*d' = (||b'(i,j)|-|b'(p,q)|| + INT(1,25*a) -d) mod a*

- *d'= (||b'(i,j))|-|b'(p,q)|| mod a)+ (INT(1,25*a) mod a) - (d mod a)*
- *d' = d +INT(0,25*a) –d = INT(0,25*a) < 0,5*a = 2t+1*

Vậy với các phép biến đổi trên, ta luôn thoả mãn được điều kiện giấu tin.

6.1.1.5. Kết luận

Bài viết đề xuất một thuật toán nhúng thuỷ vân vào ảnh tĩnh sử dụng kỹ thuật giấu tin trên miền biến đổi cosin rời rạc. Ảnh được chia thành các khối 8×8, các khối này được chọn một cách ngẫu nhiên để nhúng thuỷ vân. Mỗi khối sẽ được áp dụng phép biến đổi cosin rời rạc để chuyển dữ liệu về tần số của khối ảnh. Miền tần số thấp của khối ảnh đã được chứng minh là chứa dữ liệu nhìn thấy của ảnh, các thay đổi dữ liệu trên miền này sẽ dẫn đến thay đổi

đáng kể ảnh hiển thị. Ngược lại, miền tần số cao chứa dữ liệu ảnh không ảnh hưởng đáng kể đến tri giác ảnh. Đây là miền tần số cho phép thay đổi mà không gây nhiễu nhiều đến ảnh. Tuy nhiên, thuỷ vân trong miền này lại không bền vững với các phép biến đổi ảnh thông thường. Với thuật toán này, miền được chọn để giấu tin là miền có tần số ở giữa tần số cao và tần số thấp, kết quả thực nghiệm của thuật toán cũng cho thấy thuỷ vân đảm bảo được tính chất ẩn trên ảnh và bền vững trên một số phép biến đổi ảnh thông thường.

KẾT LUẬN

Cùng với giấu thông tin trong audio và giấu thông tin trong video, kỹ thuật giấu thông tin trong ảnh là hướng nghiên cứu chính của thuật toán giấu thông tin hiện nay và đã đạt được những kết quả khả quan. Luận văn đã trình bày một số khái niệm liên quan đến việc che giấu thông tin trong ảnh số cũng như trình bày một thuật toán giấu tin trong ảnh đen trắng, trên cơ sở đó phát triển thuật toán cho việc giấu tin trong ảnh màu và ảnh đa cấp xám.

Với thuật toán giấu tin trong ảnh màu thì tính vô hình của thông tin sau khi giấu được đảm bảo, thông qua việc chọn m,n đủ lớn những biến đổi không gây ra sự chú ý đáng kể nào. Về mặt lý thuyết thì sau khi đã có lượng thông tin được giấu vào trong ảng gốc, nó sẽ để lại dù nhiều, dù ít những dấu vết khác với ảnh gốc ban đầu. Tuy nhiên sau khi thực hiện một kỹ thuật giấu tin, quan sát bằng mắt thường và dùng những kỹ thuật thống kê đơn giản đã không thể phân biệt được đâu là ảnh gốc và đâu là ảnh có chứa thông tin ẩn. Như vậy kỹ thuật giấu tin mật vào trong ảnh tĩnh đã cho những kết quả rất triển vọng trong thông tin liên lạc có bảo mật. Còn đối với kỹ thuật Watermarking, em chưa thử nghiệm, song về nguyên tắc thì kỹ thuật giấu tin mật và kỹ thuật Watermarking không có gì khác nhau. Tuy nhiên, kỹ thuật Watermarking, để đảm bảo các yêu cầu của nó, thường người ta dùng ảnh JPEG làm ảnh môi trường do đó cần tìm hiểu thêm thuật toán biến đổi cosine rời rạc(DCT), do trình độ của em và cũng do thời gian hạn chế nên em đành lướt qua vấn đề này, mặc dù kỹ thuật Watermarking đã và đang được ứng dụng rộng rãi trong lĩnh vực kinh tế- xã hội. Em kính mong được các thầy, cô thông cảm, em xin chân thành cảm ơn.

TÀI LIỆU THAM KHẢO

1. Hồ Thị Hương Thơm- Luận án tiến sĩ toán học: Phát triển một số kỹ thuật giấu dữ liệu trong ảnh. Ứng dụng trong trao đổi thông tin
2. Luận văn tốt nghiệp cao học-Võ Văn Tùng-ĐHQG Hà Nội.
3. Giáo trình giấu tin và thủy vân ảnh-Nguyễn Xuân Huy, Trần Quốc Dũng.
4. Đồ án tốt nghiệp đại học: Thiết kế và cài đặt quy trình giấu tin trong ảnh màu-Phạm Văn Hòa.
5. Đồ án tốt nghiệp đại học: Xây dựng hệ thống bảo mật bằng kỹ thuật giấu tin- Trần Giang Nam
6. Đồ án tốt nghiệp đại học: Giấu tin trong ảnh và ứng dụng trong an toàn bảo mật thông tin-Nguyễn Thanh Cường.
7. Yu-Yuan-Chen, Hsing-Kuang Pan, and Yu-Chee Tseng. A secure Data Hiding Scheme for Two-Color Images. Taiwan