

LỜI CẢM ƠN

Trước hết em xin bày tỏ lòng biết ơn sâu sắc nhất tới cô giáo hướng dẫn Tiến sĩ Hồ Thị Hương Thom đã tận tình giúp đỡ, hướng dẫn em rất nhiều trong suốt quá trình tìm hiểu nghiên cứu và hoàn thành báo cáo tốt nghiệp.

Em xin chân thành cảm ơn các thầy cô trong bộ môn tin học – trường DHDL Hải Phòng cũng như các thầy cô trong trường đã trang bị cho em những kiến thức cơ bản cần thiết để em có thể hoàn thành báo cáo.

Xin gửi lời cảm ơn đến bạn bè những người luôn bên em đã động viên và tạo điều kiện thuận lợi cho em, tận tình giúp đỡ chỉ bảo em những gì em còn thiếu sót trong quá trình làm báo cáo tốt nghiệp.

Cuối cùng em xin bày tỏ lòng biết ơn sâu sắc tới những người thân trong gia đình đã giành cho em sự quan tâm đặc biệt và luôn động viên em.

Vì thời gian có hạn, trình độ hiểu biết của bản thân còn nhiều hạn chế. Cho nên trong đồ án không tránh khỏi những thiếu sót, em rất mong nhận được sự đóng góp ý kiến của tất cả các thầy cô giáo cũng như các bạn bè để đồ án của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

Hải phòng, ngày... tháng...năm 2013

Sinh viên thực hiện

Vũ Huy Tùng

MỤC LỤC

CHƯƠNG 1. TỔNG QUAN VỀ KỸ THUẬT GIẤU TIN.....	8
1. 1 TỔNG QUAN VỀ KỸ THUẬT GIẤU TIN.....	8
1. 1. 1 Sơ lược về dấu tin.....	8
1. 1. 2 Định nghĩa kỹ thuật giấu tin.....	8
1. 1. 3 Yêu cầu thiết yếu đối với một hệ thống giấu tin	8
1. 1. 4 Phân loại giấu tin	8
1. 1. 5. Mô hình giấu thông tin cơ bản.....	10
1. 1. 6 Môi trường giấu tin	12
1. 1. 7 Một số đặc điểm của việc giấu tin trên ảnh.....	14
1. 2. MỘT SỐ ẢNH ĐỊNH DẠNG BITMAP PHỔ BIẾN	15
1. 2. 1 Cấu trúc ảnh Bitmap	15
1. 2. 2 Cấu trúc ảnh PNG	18
1. 2. 2. 1 Lịch sử và phát triển.....	18
1. 3. PHƯƠNG PHÁP ĐÁNH GIÁ CHẤT LƯỢNG ẢNH SAU KHI GIẤU TIN	20
CHƯƠNG 2. KỸ THUẬT GIẤU TIN THUẬN NGHỊCH TRONG ẢNH	21
2. 1. KHÁI NIỆM GIẤU TIN THUẬN NGHỊCH	21
2. 1. 1. Khái niệm	21
2. 1. 2. Một số kỹ thuật giấu thuận nghịch điển hình	21
2. 2 kỹ thuật giấu tin dựa trên sự tương quan của miền không gian ảnh.....	25
2. 2. 1 giới thiệu và ý tưởng chính.	25
2. 2. 2 Phương pháp lấy mẫu	26
2. 2. 3 Thuật toán giấu tin	26
2. 2. 3 Thuật toán tách tin	29
2. 3 Ví dụ minh họa	32
CHƯƠNG 3: CÀI ĐẶT VÀ THỬ NGHIỆM.....	35
3. 1. Môi trường cài đặt	35
3. 2. Giao diện chương trình.....	36

3. 2. 1. Giao diện chương trình chính.....	36
3. 2. 2. Giao diện chức năng giấu tin	37
3. 2. 3. Giao diện chức năng tách tin	42
3. 2. 4. Giao diện chức năng đánh giá PSNR.....	49
3. 3. Kết quả thực nghiệm và nhận xét.....	54
3. 3. 1. Kết quả thực nghiệm	54
3. 3. 2. Nhận xét	64
TÀI LIỆU THAM KHẢO	66

DANH MỤC HÌNH

HÌNH 1. 1. HAI LĨNH VỰC CHÍNH CỦA KỸ THUẬT GIẤU THÔNG TIN ..	9
HÌNH 1. 2. LƯỢC ĐỒ CHUNG CHO QUÁ TRÌNH GIẤU TIN	11
HÌNH 1. 3. LƯỢC ĐỒ CHUNG CHO QUÁ TRÌNH TÁCH THÔNG TIN	12
HÌNH 2. 1. VÍ DỤ MINH HỌA TÁCH MẪU CON	26
HÌNH 2. 2. SƠ ĐỒ QUÁ TRÌNH GIẤU TIN	28
HÌNH 2. 3. SƠ ĐỒ QUÁ TRÌNH TÁCH TIN	31
HÌNH 3. 1. GIAO DIỆN CHƯƠNG TRÌNH CHÍNH	36
HÌNH 3. 2. GIAO DIỆN CHỨC NĂNG GIẤU TIN	37
HÌNH 3. 3. HỘP THOẠI CHỌN ẢNH CẤP XÁM 8 BIT CẦN GIẤU TIN	38
HÌNH 3. 4. HỘP THOẠI CHỌN THÔNG ĐIỆP CẦN GIẤU	39
HÌNH 3. 5. HỘP THOẠI CHO BIẾT TÊN ẢNH SAU KHI ĐÃ GIẤU TIN ...	40
HÌNH 3. 7. GIAO DIỆN SAU KHI GIẤU TIN	41
HÌNH 3. 8. GIAO DIỆN KẾT THÚC CHƯƠNG TRÌNH GIẤU TIN	42
HÌNH 3. 9. GIAO DIỆN CHỨC NĂNG TÁCH TIN	43
HÌNH 3. 10. HỘP THOẠI CHỌN ẢNH CẤP XÁM 8 BIT CẦN TÁCH TIN ..	44
HÌNH 3. 11. GIAO DIỆN CHỨC NĂNG NHẬP ĐỘ DÀI THÔNG ĐIỆP GIẤU TIN TỪ BÀN PHÍM	45
HÌNH 3. 12. HỘP THOẠI CHO BIẾT TÊN THÔNG ĐIỆP SAU KHI ĐÃ TÁCH TIN	46
HÌNH 3. 13. HỘP THOẠI CHO BIẾT TÊN ẢNH SAU KHI ĐÃ TÁCH TIN ..	47
HÌNH 3. 15. GIAO DIỆN SAU KHI TÁCH TIN	48
HÌNH 3. 16. GIAO DIỆN KẾT THÚC CHƯƠNG TRÌNH TÁCH TIN	49
HÌNH 3. 17. GIAO DIỆN CHỨC NĂNG GIẤU TIN	50
HÌNH 3. 18. HỘP THOẠI CHỌN ẢNH NHỊ PHÂN CẦN KIỂM TRA	51

HÌNH 3. 19. HỘP THOẠI CHỌN ẢNH NHỊ PHÂN CẦN KIỂM TRA.....	52
HÌNH 3. 20. GIAO DIỆN ĐÁNH GIÁ PSNR.....	53
HÌNH 3. 21. GIAO DIỆN THOÁT CHƯƠNG TRÌNH ĐÁNH GIÁ PSNR.....	54
HÌNH 3. 22. CHUỖI THÔNG ĐIỆP CẦN GIẤU ÍT KÍ TỰ'	55
HÌNH 3. 23 ẢNH TRƯỚC KHI GIẤU TIN	55
HÌNH 3. 24. ẢNH SAU KHI GIẤU TIN.....	56
HÌNH 3. 25. CHUỖI THÔNG ĐIỆP CẦN GIẤU NHIỀU KÍ TỰ'	58
HÌNH 3. 26. ẢNH TRƯỚC KHI GIẤU TIN	59
HÌNH 3. 27. ẢNH SAU KHI GIẤU TIN.....	60
HÌNH 3. 28. ẢNH GỐC	62
HÌNH 3. 29. ẢNH TÁCH TIN.....	62

DANH MỤC BẢNG

Bảng 1. 1. SO SÁNH GIẤU TIN MẬT VÀ GIẤU TIN THỦY VÂN SỐ.....	10
Bảng 1. 2. CẤU TRÚC ẢNH BITMAP	16
Bảng 1. 3. THÔNG TIN VỀ BITMAP HEADER	17
Bảng 1. 4. BẢNG MẪU CỦA ẢNH BITMAP	18
Bảng 3. 1. KẾT QUẢ ĐÁNH GIÁ PSNR.....	57
Bảng 3. 2. KẾT QUẢ ĐÁNH GIÁ PSNR.....	61
Bảng 3. 3. KẾT QUẢ ĐÁNH GIÁ PSNR.....	63

LỜI MỞ ĐẦU

Cuộc cách mạng thông tin kỹ thuật số đã đem lại những thay đổi sâu sắc trong xã hội và trong cuộc sống của chúng ta. Những thuận lợi mà thông tin kỹ thuật số mang lại cũng sinh ra những thách thức và cơ hội mới cho quá trình đổi mới. Với việc sử dụng mạng internet toàn cầu để thông tin, liên lạc ngày càng tăng trong mọi lĩnh vực chính trị, quân sự, quốc phòng, kinh tế, thương mại... Vấn đề được đặt ra đó là sự an toàn của dữ liệu. Một công nghệ phần nào giải quyết được vấn đề trên là giấu tin mật, nó cho phép giấu thông tin mật vào trong các nguồn thông tin khác, làm ẩn đi sự tồn tại của thông tin mật. Trong đồ án này em xin trình bày một kỹ thuật giấu tin đó là **“Kỹ thuật giấu tin dựa trên sự tương quan của miền không gian ảnh”**, gồm các chương sau:

Chương 1. Tổng quan về kỹ thuật giấu tin: Khái niệm giấu tin, mục đích của giấu tin, cấu trúc ảnh bitmap, đánh giá chất lượng ảnh bằng PSNR.

Chương 2. Kỹ thuật giấu tin dựa trên sự tương quan của miền không gian ảnh: Giới thiệu về kỹ thuật giấu tin thuận nghịch dựa trên sự tương quan của miền không gian ảnh, trình bày một số kỹ thuật giấu tin thuận nghịch, đưa ra thuật toán.

Chương 3. Cài đặt thử nghiệm: Trình bày một số giao diện của chương trình và thử nghiệm kỹ thuật giấu tin dựa trên sự tương quan của miền không gian ảnh, đưa ra nhận xét đánh giá.

Chương 1. TỔNG QUAN VỀ KỸ THUẬT GIẤU TIN

1. 1 TỔNG QUAN VỀ KỸ THUẬT GIẤU TIN

1. 1. 1 Sơ lược về giấu tin

Với hai mục đích là bảo mật cho những dữ liệu được giấu trong đối tượng mạng và bảo đảm an toàn cho chính các đối tượng chứa dữ liệu giấu trong đó (bảo vệ bản quyền) nên kỹ thuật giấu thông tin được phát triển thành hai lĩnh vực với những yêu cầu và tính chất khác nhau đó là giấu thông tin bí mật và thủy vân số.

1. 1. 2 Định nghĩa kỹ thuật giấu tin

Giấu thông tin là một kỹ thuật nhúng (giấu) một lượng thông tin số nào đó vào trong một đối tượng dữ liệu số khác (giấu thông tin chỉ mang tính quy ước không phải là một hành động cụ thể).

1. 1. 3 Yêu cầu thiết yếu đối với một hệ thống giấu tin

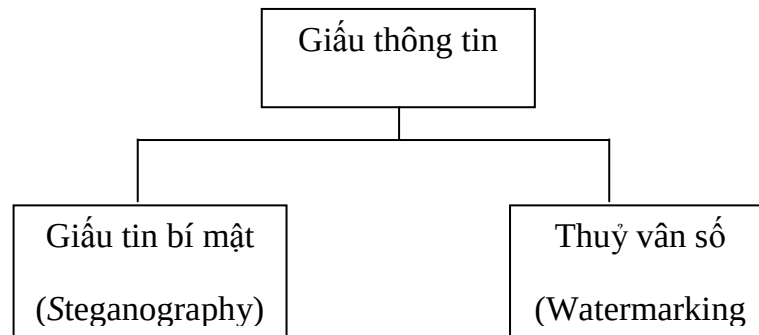
Có 3 yêu cầu thiết yếu đối với một hệ thống giấu tin:

- Tính vô hình: là một trong 3 yêu cầu của bất kì 1 hệ giấu tin nào.
- Tính bền vững: là yêu cầu thứ 2 của một hệ giấu tin. Tính bền vững là nói đến khả năng chịu được các thao tác biến đổi nào đó trên phương tiện nhúng và các cuộc tấn công có chủ đích.
- Khả năng nhúng: là yêu cầu thứ 3 của một hệ giấu tin. Khả năng nhúng chính là số lượng thông tin nhúng được nhúng trong phương tiện chứa.

1. 1. 4 Phân loại giấu tin

Có thể phân loại kỹ thuật giấu tin làm hai hướng:

- Giấu tin mật (Steganography).
- Thủy vân số (Watermarking).



Hình 1. 1. Hai lĩnh vực chính của kỹ thuật giấu thông tin

❖ **Giấu tin mật (Steganography)**

- Khả năng không thể nhận biết (imperceptibility).
- 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.

❖ **Thủy vân số (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 vầu 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.

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

Bảng 1. 1. So sánh giấu tin mật và giấu tin thủy vân số

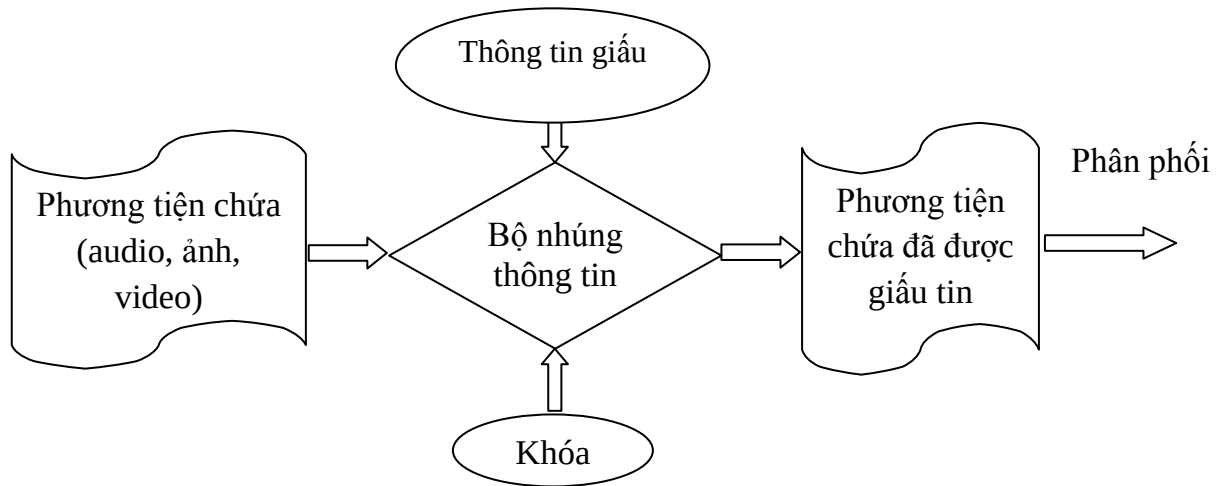
	Giấu thông tin mật	Thủy vân số
Mục tiêu	Tàng hình các phiên liên lạc để bảo mật thông tin Dung trong các liên lạc xác định	Chủ yếu phục vụ cho mục đích bảo vệ bản quyền Chủ yếu dùng trong các hoạt động xuất bản
Cách thực hiện	Không làm thay đổi phương tiện chứa	Có thể thay đổi nhỏ về cảm nhận tới phương 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. 1. 5. Mô hình giấu thông tin cơ bản

Thành phần cơ bản của kỹ thuật giấu thông tin bao gồm : thuật toán giấu tin và bộ giải mã thông tin(tính đến cả khóa mật).

Thuật toán giấu tin được dùng để giấu thông tin vào một phương tiện chứa bằng cách sử dụng một khoá bí mật được dùng chung bởi người mã và người giải mã, việc giải mã thông tin chỉ có thể thực hiện được khi có khoá. Bộ giải mã thực hiện quá trình giải mã trên phương tiện chứa đã chứa dữ liệu và trả lại thông điệp ẩn trong đó

❖ **Lược đồ chung cho quá trình giấu tin**



Hình 1. 2. Lược đồ chung cho quá trình giấu tin

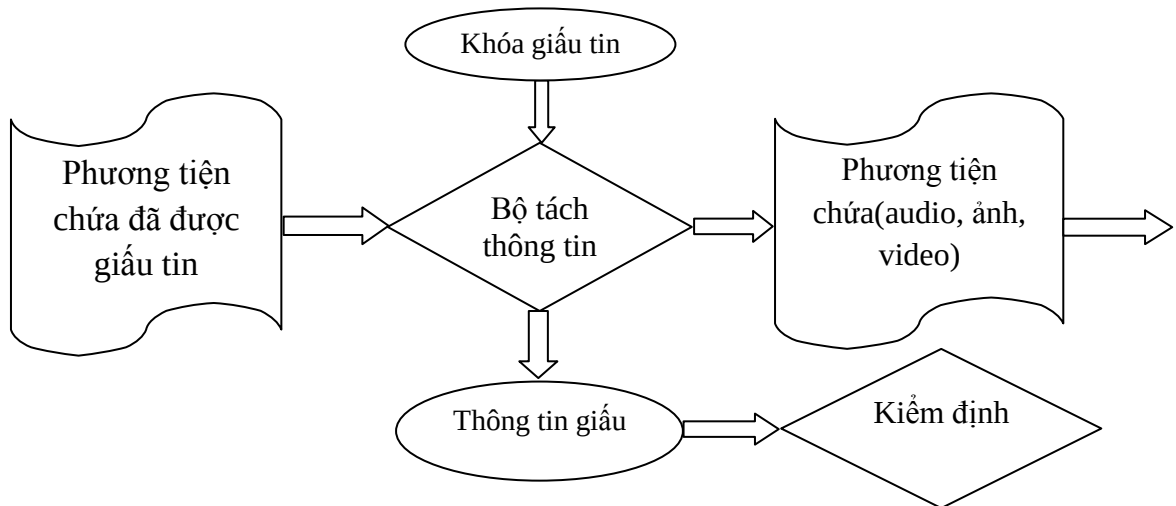
Thông tin cần giấu tùy theo mục đích của người sử dụng, nó có thể là thông tin mật (với các tin bí mật) hay các logo, hình ảnh bản quyền.

Phương tiện chứa: các *file* ảnh, *text*, *audio*... là môi trường để nhúng tin.

Bộ nhúng thông tin: là những chương trình thực hiện việc giấu tin

Đầu ra: là các phương tiện chứa đã có tin giấu trong đó

Tách thông tin từ các *phương tiện chứa* diễn ra theo quy trình ngược lại với đầu ra là các thông tin đã được giấu vào *phương tiện chứa*. *Phương tiện chứa* sau khi tách lấy thông tin có thể được sử dụng, quản lý theo những yêu cầu khác nhau.

❖ **Lược đồ chung cho quá trình tách thông tin****Hình 1. 3.** Lược đồ chung cho quá trình tách thông tin

Hình 1. 3 chỉ ra các công việc giải mã thông tin đã giấu. Sau khi nhận được đối tượng *phương tiện chứa* có giấu thông tin, quá trình giải mã được thực hiện thông qua một bộ giải mã tương ứng với bộ nhúng thông tin cùng với khoá của quá trình nhúng. Kết quả thu được gồm *phương tiện chứa* gốc và thông tin đã giấu. Bước tiếp theo thông tin đã giấu sẽ được xử lý kiểm định so sánh với thông tin ban đầu.

1. 1. 6 Môi trường giấu tin

Dữ liệu đa phương tiện là dạng dữ liệu đa dạng, phong phú hiện đang tồn tại trong mọi lĩnh vực ứng dụng của công nghệ thông tin. Chính vì vậy, dữ liệu đa phương tiện cũng được sử dụng làm phương tiện chứa trong quá trình giấu tin. Để lựa chọn dữ liệu làm phương tiện chứa cần phải tìm hiểu rõ về cấu trúc của loại dữ liệu đó. Một số dữ liệu đa phương tiện thường được sử dụng làm môi trường để giấu tin như: text, audio, video, ảnh,...

❖ **Giấu thông tin trong ảnh số (Data Hiding in Image)**

Giấu thông tin trong ảnh số là một phần của khái niệm giấu thông tin với việc sử dụng ảnh số làm phương tiện chứa. Hiện nay, giấu thông tin trong ảnh là một phương pháp chiếm tỉ lệ lớn nhất trong các chương trình ứng dụng. Các phần mềm hệ thống giấu tin trong đa phương tiện bởi lượng thông tin được trao đổi trong hình ảnh là rất lớn và hơn nữa giấu thông tin trong ảnh cũng đóng vai trò hết sức quan trọng trong hầu hết các ứng dụng bảo vệ an toàn thông tin như: xác thực thông tin,

xác định làm thay đổi thông tin, bảo vệ bản quyền tác giả, điều khiển truy cập, giấu thông tin mật.... Chính vì thế mà vấn đề này đã nhận được sự quan tâm rất lớn của các cá nhân, tổ chức, trường đại học và viện nghiên cứu trên thế giới.

Ngày nay, khi ảnh số đã được sử dụng khá phổ biến, thì giấu thông tin trong ảnh đã đem lại nhiều ứng dụng quan trọng trên các lĩnh vực của đời sống xã hội. Ví dụ như đối với các nước phát triển, chữ ký tay đã được số hoá và lưu trữ sử dụng như là hồ sơ cá nhân của các dịch vụ ngân hàng và tài chính. Nó được dùng để xác thực trong các thẻ tín dụng của người tiêu dùng.

Trong một số ứng dụng về nhận diện thẻ chứng minh, thẻ căn cước, hộ chiếu..., người ta có thể giấu thông tin trên các ảnh thẻ để xác định thông tin thực.

Một đặc điểm của giấu thông tin trong ảnh đó là thông tin được giấu một cách “vô hình”. Nó như là cách thức truyền thông tin mật cho nhau mà người khác không thể biết được, bởi sau khi giấu thông tin thì chất lượng ảnh gần như không thay đổi, đặc biệt đối với ảnh màu hay ảnh xám.

❖ *Giấu tin trong audio*

Giấu thông tin trong audio mang những đặc điểm riêng khác với giấu thông tin trong các đối tượng đa phương tiện khác. Một trong những yêu cầu cơ bản của giấu thông tin là đảm bảo tính chất “ẩn” của thông tin được giấu đồng thời không làm ảnh hưởng đến chất lượng của dữ liệu. Kỹ thuật giấu thông tin trong ảnh phụ thuộc vào hệ thống thị giác của con người - HVS (Human Vision System), còn kỹ thuật giấu thông tin trong audio lại phụ thuộc vào hệ thống thính giác - HAS (Human Auditory system). Một vấn đề phức tạp ở đây là hệ thống thính giác của con người nghe được các tín hiệu ở các dải tần rộng và công suất lớn nên sẽ gây khó khăn đối với các phương pháp giấu tin trong audio. Nhưng thật may là thính giác con người lại kém trong việc phát hiện sự khác biệt các dải tần và công suất, có nghĩa là các âm thanh to, cao tần có thể che giấu được các âm thanh nhỏ ở tần số thấp một cách dễ dàng. Các mô hình phân tích tâm lý đã chỉ ra điểm yếu trên và thông tin này sẽ giúp ích cho việc chọn các audio thích hợp cho việc giấu tin. Vấn đề khó khăn thứ hai đối với giấu thông tin trong audio là kênh truyền tin. Kênh truyền hay băng thông chậm sẽ ảnh hưởng đến chất lượng thông tin sau khi giấu. Ví dụ để nhúng một đoạn phần mềm java applet vào một đoạn audio (16 bit, 44. 100hz) có chiều dài bình thường thì các phương pháp thông thường cũng cần ít nhất tốc độ đường

truyền là 20bps. Giấu thông tin trong audio đòi hỏi yêu cầu rất cao về tính đồng bộ và tính an toàn của thông tin. Các phương pháp giấu thông tin cho audio đều lợi dụng điểm yếu trong hệ thống thính giác của con người.

❖ Giấu thông tin trong video

Cũng như giấu thông tin trong ảnh hay trong audio, giấu thông tin trong video đã được quan tâm và phát triển mạnh mẽ cho nhiều ứng dụng như điều khiển truy cập thông tin, xác thực thông tin và bảo vệ bản quyền tác giả, ví dụ như trong, các hệ thống chương trình tivi phải trả tiền. Các kỹ thuật giấu tin trong video cũng được phát triển mạnh mẽ và theo hai khuynh hướng là thủy văn số và giấu dữ liệu. Một phương pháp giấu tin trong video được đưa ra bởi COX là phương pháp phân bố đều. Ý tưởng cơ bản của phương pháp này là phân phối thông tin giấu dần trải theo tần số của dữ liệu gốc. Nhiều nhà nghiên cứu đã dùng những hàm số cosin riêng và các hệ số truyền sóng riêng để giấu thông tin. Trong các thuật toán khởi nguồn thì thường chỉ có các kỹ thuật cho phép giấu các ảnh vào trong video, nhưng thời gian gần đây các kỹ thuật mới đã cho phép giấu cả âm thanh và hình ảnh vào trong video.

❖ Giấu thông tin trong văn bản dạng text

Giấu tin trong văn bản dạng *text* khó thực hiện hơn do có ít các thông tin dư thừa, để làm được điều này người ta phải khéo léo khai thác các dư thừa tự nhiên của ngôn ngữ. Một cách khác là tận dụng các định dạng văn bản (mã hoá thông tin vào khoảng cách giữa các từ hay các dòng văn bản) => Kỹ thuật giấu tin đang được áp dụng cho nhiều loại đối tượng chứ không riêng dữ liệu đa phương tiện như ảnh, *audio*, *video*.

1. 1. 7 Một số đặc điểm của việc giấu tin trên ảnh

1. 1. 7. 1 Tính vô hình của thông tin

Khái niệm này dựa trên đặc điểm của hệ thống thị giác của con người. Thông tin nhúng là không tri giác được nếu một người với thị giác bình thường không phân biệt được ảnh môi trường và ảnh kết quả (tức là không phân biệt được ảnh trước và sau khi giấu thông tin). Trong khi giấu thông tin mật (*Steganography*) yêu cầu tính vô hình của thông tin ở mức độ cao thì giấu thông tin theo kiểu đánh dấu – thủy văn (*watermarking*) lại chỉ yêu cầu ở một cấp độ nhất định. Chẳng hạn như người ta áp

dùng giấu thông tin theo kiểu đánh dấu – thủy vân (*watermarking*) cho việc gắn một biểu tượng mờ vào một chương trình truyền hình để bảo vệ bản quyền.

1. 1. 7. 2 Tỷ lệ giấu tin

Lượng thông tin dấu so với kích thước ảnh môi trường cũng là một vấn đề cần quan tâm trong một thuật toán dấu tin. Rõ ràng là có thể chỉ giấu 1 bit thông tin vào mỗi ảnh mà không cần lo lắng về độ nhiễu của ảnh nhưng như vậy sẽ rất kém hiệu quả khi mà thông tin giấu có kích thước bằng kb. Các thuật toán đều cố gắng đạt được mục đích làm thế nào giấu được nhiều thông tin nhất mà không gây ra nhiễu đáng kể.

1. 1. 7. 3 Tính bảo mật

Thuật toán nhúng tin được coi là có tính bảo mật nếu thông tin được nhúng không bị tìm ra khi bị tấn công một cách có chủ đích trên cơ sở có hiểu biết đầy đủ về thuật toán nhúng tin và có bộ giải mã (trừ khóa bí mật), hơn nữa còn có được ảnh có mang thông tin (ảnh kết quả). Đây là một yêu cầu rất quan trọng đối với ảnh *image hiding*.

1. 1. 7. 4 Ảnh môi trường đối với quá trình giải mã

Yêu cầu cuối cùng là thuật toán phải cho phép lấy lại được những thông tin đã giấu trong ảnh mà không có ảnh gốc. Điều này là một thuận lợi khi ảnh môi trường là duy nhất nhưng lại làm giới hạn khả năng ứng dụng của kỹ thuật giấu tin.

1. 2. MỘT SỐ ẢNH ĐỊNH DẠNG BITMAP PHỔ BIẾN

1. 2. 1 Cấu trúc ảnh Bitmap

Ảnh BMP (*Bitmap*) được phát triển bởi *Microsoft Corporation*, được lưu trữ dưới dạng độc lập thiết bị cho phép *Windows* hiển thị dữ liệu không phụ thuộc vào khung chỉ định màu trên bất kì phần cứng nào. Tên *file* mở rộng mặc định của một *file* ảnh *Bitmap* là “. BMP”. Ảnh BMP được sử dụng trên *Microsoft Windows* và các ứng dụng chạy trên *Windows* từ *version 3. 0* trở lên.

Mỗi file ảnh *Bitmap* gồm 3 phần như bảng 1. 2:

Bảng 1. 2. Cấu trúc ảnh BitMap

Bitmap Header (54 byte)
Color Palette
Bitmap Data

1. 2. 1. 1. *Bitmap Header*

Thành phần *bitcount* (Bảng 1. 2) của cấu trúc *Bitmap Header* cho biết số bit dành cho mỗi 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 bit biểu diễn 1 điểm ảnh. Nếu bit mang giá trị “0” thì điểm ảnh là điểm đen, nếu bit mang giá trị “1” thì điểm ảnh là điểm trắng.
- 4: *Bitmap* là ảnh 16 màu, mỗi điểm ảnh được biểu diễn bằng 4 bit.
- 8: *Bitmap* là ảnh 256 màu, mỗi điểm ảnh được biểu diễn bằng 8 bit.
- 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 và xanh lơ (RGB) của điểm ảnh.
- 24: *Bitmap* là ảnh *True Color*, mỗi dãy 3 byte liên tiếp trong *Bitmap* biểu diễn cường độ tương đối của màu đỏ, xanh lá cây và xanh lơ (RGB) của điểm ảnh.

Thành phần *Color Used* của cấu trúc *Bitmap Header* xác định số lượng màu của *Palette* 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*.

Bảng 1. 3. Thông tin về Bitmap Header

Byte thứ	Ý nghĩa	Giá trị
1-2	Nhận dạng file	‘BM’ hay 19778
3-6	Kích thước file	Kiểu long trong Turbo C
7-10	Dự trữ	Thường mang giá trị 0
11-14	Byte bắt đầu vùng dữ liệu	Offset của byte bắt đầu vùng dữ liệu
15-18	Số byte cho vùng thông tin	4 byte
19-22	Chiều rộng ảnh BMP	Tính bằng pixel
23-26	Chiều cao ảnh BMP	Tính bằng pixel
27-28	Số Planes màu	Cố định là 1
29-30	Số bit cho 1 pixel (bitcount)	Có thể là: 1, 4, 8, 16, 24 tùy theo loại ảnh
31-34	Kiểu nén dữ liệu	0: Không nén 1: Nén runlength 8bits/pixel 2: Nén runlength 4bits/pixel
35-38	Kích thước ảnh	Tính bằng byte
39-42	Độ phân giải ngang	Tính bằng pixel / metter
43-46	Độ phân giải dọc	Tính bằng pixel / metter
47-50	Số màu sử dụng trong ảnh	
51-54	Số màu được sử dụng khi hiển thị ảnh (Color Used)	

1. 2. 1. 2. *Palette màu*

Bảng màu của ảnh. Chỉ những ảnh nhỏ hơn hoặc bằng 8 bit mới có bảng màu.

Bảng 1. 4. Bảng màu của ảnh BITMAP

Địa chỉ (Offset)	Tên	Ý nghĩa
0	RgbBlue	Giá trị cho màu xanh blue
1	RgbGreen	Giá trị cho màu xanh Green
2	RgbRed	Giá trị cho màu đỏ
3	RgbReserved	Dự trữ

1. 2. 1. 3 *Bitmap data*

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

1. 2. 2 *Cấu trúc ảnh PNG*

1. 2. 2. 1 *Lịch sử và phát triển*

Động cơ thúc đẩy cho việc tạo ra định dạng PNG bắt đầu vào khoảng đầu năm 1995, sau khi *Unisys* công bố họ sẽ áp dụng bằng sáng chế vào thuật toán nén dữ liệu LZW- được sử dụng trong định dạng GIF. Thuật toán được bảo vệ bởi bằng công nhận độc quyền sáng tạo ở Mỹ và tất cả các nước trên thế giới. Tuy nhiên, cũng đã có một số vấn đề với định dạng GIF khi cần có một số thay đổi nhất định trên hình ảnh, giới hạn của nó là 256 màu trong thời điểm máy tính có khả năng hiển thị nhiều hơn 256 màu đang trở nên phổ biến. Mặc dù định dạng GIF có thể thể hiện các hình ảnh động, song PNG vẫn được quyết định là định dạng hình ảnh đơn (chỉ có một hình duy nhất). Một người "anh em" của nó là MNG đã được tạo ra để giải quyết vấn đề ảnh động. PNG lại tăng thêm sự phổ biến của nó vào tháng 8 năm 1999, sau khi hãng *Unisys* huỷ bỏ giấy phép của họ đối với các lập trình viên phần mềm miễn phí, và phi thương mại.

- Phiên bản 1. 0 của đặc tả PNG được phát hành vào ngày 1 tháng 7 năm 1996, và sau đó xuất hiện với tư cách RFC 2083. Nó được tổ chức W3C khuyến nghị vào ngày 1 tháng 10 năm 1996.
- Phiên bản 1. 1, với một số thay đổi nhỏ và thêm vào 3 thành phần mới, được phát hành vào ngày 31 tháng 12 năm 1998.
- Phiên bản 1. 2, thêm vào một thành phần mở rộng, được phát hành vào ngày 11 tháng 8 năm 1999.
- PNG giờ đây là một chuẩn quốc tế (ISO/IEC 15948:2003), và cũng được công bố như một khuyến nghị của W3C vào ngày 10 tháng 11 năm 2003. Phiên bản hiện tại của PNG chỉ khác chút ít so với phiên bản 1. 2 và không có thêm thành phần mới nào.

1. 2. 2. 2 Thông tin kỹ thuật

a. Phần đầu của tập tin

Một tập tin PNG bao gồm 8-byte kí hiệu (89 50 4E 47 0D 0A 1A) được viết trong hệ thống có cơ số 16, chứa các chữ "PNG" và hai dấu xuống dòng, ở giữa là sắp xếp theo số lượng của các thành phần, mỗi thành phần đều chứa thông tin về hình ảnh. Cấu trúc dựa trên các thành phần được thiết kế cho phép định dạng PNG có thể tương thích với các phiên bản cũ khi sử dụng.

b. Các "thành phần" trong tập tin

PNG là cấu trúc như một chuỗi các thành phần, mỗi thành phần chứa kích thước, kiểu, dữ liệu, và mã sửa lỗi CRC ngay trong nó.

Chuỗi được gán tên bằng 4 chữ cái phân biệt chữ hoa chữ thường. Sự phân biệt này giúp bộ giải mã phát hiện bản chất của chuỗi khi nó không nhận dạng được.

Với chữ cái đầu, viết hoa thể hiện chuỗi này là thiết yếu, nếu không thì ít cần thiết hơn (*ancillary*). Chuỗi thiết yếu chứa thông tin cần thiết để đọc được tệp và nếu bộ giải mã không nhận dạng được chuỗi thiết yếu, việc đọc tệp phải được hủy.

c. Thành phần cơ bản

Một bộ giải mã (*decoder*) phải có thể thông dịch để đọc và hiển thị một tệp PNG.

- IHDR phải là thành phần đầu tiên, nó chứa đựng *header*

- PLTE chứa đựng bảng màu (danh sách các màu)
- IDAT chứa đựng ảnh. Ảnh này có thể được chia nhỏ chứa trong nhiều phần IDAT. Điều này làm tăng kích cỡ của tệp lên một ít nhưng nó làm cho việc phát sinh ảnh PNG mượt hơn (*streaming manner*).
- IEND đánh dấu điểm kết thúc của ảnh.

1. 3. PHƯƠNG PHÁP ĐÁNH GIÁ CHẤT LƯỢNG ẢNH SAU KHI GIẤU TIN

Để đánh giá chất lượng của bức ảnh (hay khung ảnh *video*) ở đầu ra của bộ mã hoá, người ta thường sử dụng hai tham số: Sai số bình phương trung bình – MSE (*Mean Square Error*) và phương pháp đề xuất với hệ số tỷ lệ tín hiệu / tín hiệu nhiễu PSNR (*Peak Signal to Noise Ratio*).

MSE giữa ảnh gốc và ảnh khôi phục được tính như sau:

$$MSE = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n (x_{ij} - y_{ij})^2$$

Ở đây:

x_{ij} biểu thị giá trị điểm ảnh gốc

y_{ij} biểu thị giá trị điểm ảnh đã được biến đổi

m và n lần lượt là chiều rộng và chiều cao của ảnh.

PSNR, đơn vị: *deciben* (dB), thường được sử dụng trong nghiên cứu xử lý hình ảnh:

Chương 2. KỸ THUẬT GIẤU TIN THUẬN NGHỊCH TRONG ẢNH

2. 1. KHÁI NIỆM GIẤU TIN THUẬN NGHỊCH

2. 1. 1. Khái niệm

Theo đặc tính tách tin cần lưu trữ hay không lưu trữ vật mang tin mà người ta phân kỹ thuật giấu tin ra làm 2 loại:

- Kỹ thuật giấu không thuận nghịch: Là kỹ thuật giấu sau khi tách thông điệp không thể khôi phục lại ảnh gốc. Những kỹ thuật này phục vụ trao đổi thông tin mật. Người ta có thể hủy vật mang tin khi cần thiết mà không cần lưu trữ.
- Kỹ thuật giấu thuận nghịch: Là kỹ thuật giấu sau khi tách thông điệp có thể khôi phục lại ảnh gốc. Những kỹ thuật này phục vụ trong một số lĩnh vực như: y học, quân sự, nghiên cứu năng lượng hoặc hệ thống thông tin vệ tinh, ...

2. 1. 2. Một số kỹ thuật giấu thuận nghịch điển hình

Năm 1999, Honsinger và các cộng sự đề xuất kỹ thuật giấu thuận nghịch đầu tiên [1], mở ra một hướng mới trong lĩnh vực giấu tin. Tiếp đó một loạt các kỹ thuật giấu tin thuận nghịch khác được công bố. Sau đây giới thiệu sơ lược một số kỹ thuật giấu tiêu biểu.

Kỹ thuật mở rộng sai phân DE (*Difference Expansion*) do Tian đưa ra [3], đây là kỹ giấu tin dựa trên mở rộng hệ số sai phân của điểm ảnh, dữ liệu ảnh được tính sai phân theo biểu thức $D_i = I_i - I_{i+1}$ (I_i là giá trị *pixel* của ảnh), thông tin được giấu trên LSB của các hệ số sai phân sau khi được mở rộng. Sau đó tác giả đề xuất tiếp phương pháp mở rộng trên các hệ số *wavelet* để giấu tin. Đến năm 2008, Shaowei Weng và các đồng nghiệp đưa ra kỹ thuật DE cải tiến bằng cách thêm vào hàm nén – giãn trong quá trình giấu tin sử dụng DE nhằm giảm nhiễu xảy ra (theo đánh giá bằng PSNR) của kỹ thuật giấu thuận nghịch DE.

Năm 2003, Ni và cộng sự đề xuất kỹ thuật giấu thuận nghịch dựa trên dịch chuyển biểu đồ tần suất gọi là NSAS [2]. Tiếp đó một loạt các kỹ thuật giấu thuận nghịch dựa phương pháp này ra đời.

2. 1. 2. 1 Phương pháp giấu NSAS.

Giới thiệu: Kỹ thuật histogram được Ni và cộng sự đề xuất năm 2006. So với nghiên cứu của Titan được đề xuất 2003 kỹ thuật Ni và cộng sự phù hợp với công việc đòi hỏi chất lượng hình ảnh [2].

Ý tưởng: Đề xuất của Ni với cộng sự nhúng dữ liệu bằng cách dịch chuyển các biểu đồ tần suất. Phương pháp này lần đầu tiên xây dựng một biểu đồ ảnh của ảnh gốc để có được một điểm cực đại và một điểm cực tiểu sau đó các dữ liệu nhúng bằng cách di chuyển các biểu đồ tần suất dựa vào điểm cực đại và điểm cực tiểu này.

Thuật toán nhúng.

Đầu vào: Một ảnh cấp xám C , một chuỗi thông điệp M .

Đầu ra: Ảnh giấu tin S .

Bước 1: Quét tất cả các ảnh và xây dựng biểu đồ tần số $H_1(x)$, $x \in [0, 255]$. Trong histogram, giá trị màu xám cao nhất được kí hiệu là a , giá trị màu xám thấp nhất được kí hiệu là b . $H_1(b) = 0$, thì b được gọi là một điểm cực tiểu. Để đơn giản, giả sử $a < b$.

Bước 2: Quét ảnh một lần nữa và ghi lại các giá trị điểm ảnh bằng b và đặt chúng vào bản đồ L . Sau đó, thay đổi biểu đồ tần số $H_1(x)$, $x \in [0, 255]$ sang bên phải một đơn vị để trống cột tần suất tại vị trí có giá trị $a+1$.

Bước 3: Trích một bit dữ liệu từ dữ liệu bí mật S . Quét tất cả các ảnh 1 lần nữa. Nếu quét giá trị các điểm ảnh và các bit dữ liệu nhúng là 1, thì đặt giá trị điểm ảnh là $a+1$. Nếu bit dữ liệu được nhúng là 0, thì không thay đổi các điểm ảnh được quét.

Bước 4: lặp lại bước 3 cho đến khi dữ liệu S được nhúng hoàn toàn.

Thuật toán tách tin.

Đầu vào: Ảnh Giấu tin S .

Đầu ra: Ảnh khôi phục và chuỗi thông điệp M .

Bước 1: Quét tất cả các ảnh theo thứ tự như trong giai đoạn nhúng. Nếu quét được giá trị a , thì tách bit 0 khỏi a . Nếu quét được giá trị $a+1$ thì tách bit 1 ra khỏi a .

Bước 2: Quét tất cả cá ảnh 1 lần nữa và dịch chuyển $H_1(x)$, $x \in [0, 255]$ sang trái 1 đơn vị.

Bước 3: thiết lập các giá trị các giá trị ghi được trong bản đồ L là b .

2. 1. 2. 2. Thuật toán cải tiến NSAS.

Giới thiệu: Thuật toán NSAS được J. H. Hwang, J. W. Kim, and J. U. Choi cải tiến và đề xuất năm 2010[3].

Ý tưởng: Thay vì dịch chuyển tất cả các điểm ảnh giữa điểm cực đại và điểm cực tiểu trước khi nhúng, ta kết hợp sự dịch chuyển và các quá trình nhúng với nhau để chỉ ra số lượng điểm ảnh dịch chuyển cho kích thước dữ liệu nhúng nhất định. Vì vậy không có thêm khoảng trống giữa các điểm ảnh để dịch chuyển so với phương pháp Ni và cộng sự (2006)

Thuật toán nhúng tin.

Đầu vào: Một ảnh cấp xám C , chuỗi thông điệp M .

Đầu ra: Ảnh giấu tin S .

Bước 1: Quét tất cả các ảnh và xây dựng biểu đồ tần suất $H_1(x)$, $x \in [0, 255]$. Trong histogram, có điểm cực đại a , điểm cực tiểu b . Không mất tính khái quát, giả sử $a < b$.

Bước 2: Thiết lập $k = 0$. Giá trị k được sử dụng để cho biết số bit dữ liệu nhúng.

Bước 3: Quét tất cả các ảnh một lần nữa. Nếu quét được giá trị điểm ảnh bằng 1, trích 1 bit dữ liệu từ S , thiết lập $k = k + 1$ và tiếp tục bước 4 để nhúng dữ liệu S , nếu không, thực hiện bước 5.

Bước 4: Nếu bit dữ liệu là 1, thì thiết lập giá trị điểm ảnh quét được là $a+1$, nếu không có thay đổi gì cho những điểm ảnh này, quay lại bước 3 tiếp tục quá trình nhúng.

Bước 5: Nếu tất cả các giá trị điểm ảnh quét được nằm trong khoảng (a, b) , thì cộng các giá trị điểm ảnh đó thêm 1. Ghi lại vị trí các điểm ảnh có giá trị điểm ảnh bằng b .

Thuật toán tách tin.

Đầu vào: Ảnh giấu tin S.

Đầu ra: Ảnh khôi phục và chuỗi thông điệp M.

Bước 1: Thiết lập $k = 0$.

Bước 2: Quét tất cả các ảnh theo thứ tự như trong quá trình nhúng. Nếu quét được giá trị là a, thì đặt $k = k+1$ và tách bit 0 khỏi a. Nếu quét được giá trị là a+1, thì $k = a+1$ và tách bit 1 ra khỏi a. Nếu giá trị quét nằm trong khoảng (a, b) thì các giá trị điểm ảnh quét được trừ đi 1. Nếu vị trí các điểm ảnh được ghi trong bản đồ L, thì thiết lập giá trị các điểm ảnh quét được là b.

Bước 3: Lặp lại bước 2 cho đến khi $k = |S|$

So sánh phương pháp NSAS và NSAS cải tiến để thấy được sự hiệu quả của phương pháp cải tiến NSAS. Một ảnh màu xám 8 bit: *Baboon* kích thước 512x512 được lựa chọn để thí nghiệm, dùng PSNR để đánh giá.

2. 1. 2. 3 Phương pháp giấu tin trên miền biến đổi Wavelet

Giới thiệu : Thuật toán Difference được Jun Tian đề xuất vào năm 2002[4].

Ý tưởng : Tách wavelet được các dải LL, LH. Dữ liệu giấu trên dải LH. Lấy 2 điểm ảnh gần nhau lấy giá trị trung bình của 2 điểm ảnh này, tính hiệu 2 điểm ảnh và lần lượt nhúng dữ liệu.

Thuật toán giấu tin.

Đầu vào: Ảnh sử dụng để giấu tin, thông tin cần giấu.

Đầu ra: Ảnh đã giấu tin.

Bước 1: Đặt $C_j = \{c_{jk}\}_{k=0}^{n^2-1}$ là khối thứ j kích thước $n \times n$ lấy từ LH (hoặc HL, LL) bằng con của miền biến đổi IWT (của ảnh đầu vào). Gọi $C_{jp} = \{c_p | \beta \leq c_p < 2\beta\}$ và $C_{jm} = \{c_m | -2\beta \leq c_m < -\beta\}$ là hai tập hợp con của C_j với β ở đây là một tham số điều khiển.

Bước 2: Chọn một khối C_j chưa được xử lý.

Bước 3: Nếu $|C_{jp}| \neq \emptyset$ thì sau đó trừ đi từ mỗi hệ số C_{jp} và đánh dấu một lá cờ để sửa đổi hệ số.

Bước 4: Nếu $|C_{jm}| \neq \emptyset$ thì sau đó cộng vào mỗi hệ số C_{jm} và đánh dấu một lá cờ để sửa đổi hệ số.

Bước 5: Giấu từng bit thông tin R bằng cách chọn các hệ số $c_i \in C_j$ với $0 \leq c_i < \beta$ (hoặc $-\beta \leq c_i < 0$) nhân c_i với 2 để có được $\hat{c}_i$, và thêm một thông tin cần giấu R vào $\hat{c}_i$ theo biểu thức $(\hat{c}_i + R)$.

Bước 6: Lặp lại từ bước 1 cho đến khi giấu hết các bit thông tin.

Thuật toán tách tin.

Đầu vào: Ảnh có giấu tin, thông số kiểm soát β và cờ.

Đầu ra: Thông tin được giấu trong ảnh, ảnh gốc.

Bước 1: Nhập một D_j là khối j thứ m bất kì của kích thước $n \times n$ được lấy từ các băng con LL, (HL), LH của miền biến đổi IWT.

Bước 2: Cho $\hat{D}_j = \{\hat{d}_j \mid -2\beta \leq \hat{d}_j < 2\beta\}$ với $\hat{D}_j \subseteq D_j$.

Bước 3: Thông tin đã giấu được lấy ra bằng phép mod $(\hat{d}_j, -2)$.

Bước 4: Các hệ số IWT $\hat{d}_j$ ban đầu (của ảnh gốc) một bit dữ liệu có thể được phục hồi bằng cách thực hiện hay $\tilde{d}_j = \lfloor \hat{d}_j / 2 \rfloor$ nếu $\hat{d}_j \geq 0$ hoặc $\tilde{d}_j = \lceil (\hat{d}_j / 2) - 0.5 \rceil$ nếu $\hat{d}_j < 0$.

Bước 5: Các hệ số IWT ban đầu có thể được phục hồi bằng cách cộng (hoặc trừ) β đến $\tilde{d}_j$ nếu $\tilde{d}_j \geq 0$ (hoặc $\tilde{d}_j < 0$), trong khi lá cờ của $\tilde{d}_j$ đã được đánh dấu.

Bước 6: Lặp lại từ bước 1 cho đến khi tất cả các bit dữ liệu được trích ra.

2. 2 Kỹ thuật giấu tin dựa trên sự tương quan của miền không gian ảnh

2. 2. 1 Giới thiệu và ý tưởng chính.

Giới thiệu: Thuật toán được K. S. Kim, M. J. Lee, H. Y. Lee, and H. K. Lee đề xuất vào năm 2009 [5].

Ý tưởng: Sử dụng miền không gian ảnh như một mẫu. Tách ra thành các mẫu con (nổi dung được tách ra giống như ảnh ban đầu được thu nhỏ, xem hình (2. 1). Sau đó chọn một mẫu con làm trọng tâm. Tính sai phân của các mẫu con với mẫu làm tâm được các hệ số sai phân. Chọn một khoảng hệ số $[-L, L]$ để giấu tin. Chi tiết giấu tin và tách tin xem hình dưới đây.

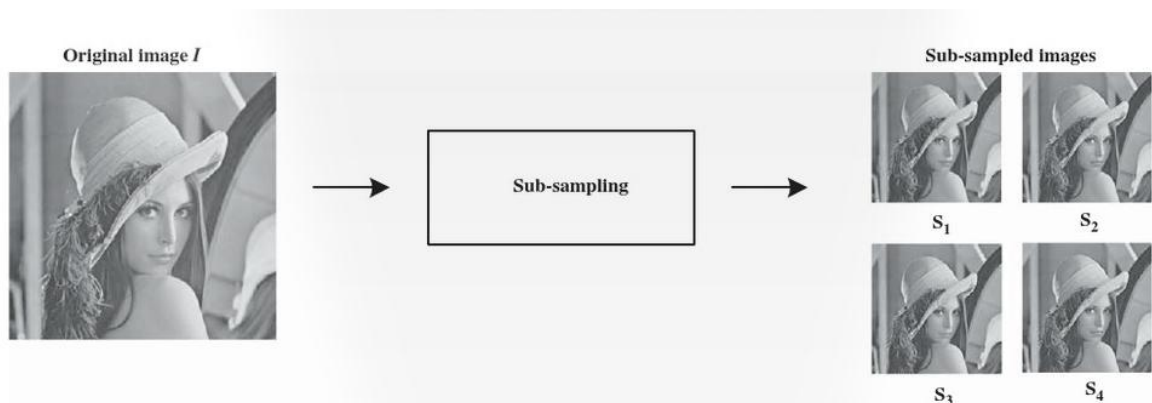
2. 2. 2 Phương pháp lấy mẫu

Lấy mẫu là quá trình lựa chọn đơn vị (ví dụ: điểm ảnh, hệ số) từ một hình ảnh. Giả sử rằng một ảnh có kích cỡ $N \times M$ điểm ảnh ký hiệu là $I(x, y)$, trong đó $x = 0, \dots, M-1$ và $y = 0, \dots, N-1$. Hai yếu tố lấy mẫu Δu và Δv , ta có phương pháp lấy mẫu theo hàng và cột theo phương trình (2. 1):

$$S_k(i, j) = I \left(\left\lfloor \frac{k-1}{\Delta u} \right\rfloor \cdot j \cdot \Delta u + ((k-1) \bmod \Delta u) \right) \quad (2.1)$$

Trong đó $i=0, \dots, M/\Delta v-1$, $j=0, \dots, N/\Delta u-1$, và $k=1, \dots, \Delta u \times \Delta v$, nếu $N/\Delta u$ hoặc $M/\Delta v$ không phải là một giá trị số nguyên, chúng ta thay đổi kích thước của tất cả các ảnh mẫu.

Ví dụ nếu $M=N=512$ và $\Delta u = \Delta v = 3$, kích thước của ảnh mẫu được đặt ở số nguyên gần nhất nhỏ hơn hoặc bằng số: $[512/3] = 170$ trong cả hai chiều rộng và chiều cao.



Hình 2. 1. Ví dụ minh họa tách mẫu con

2. 2. 3 Thuật toán giấu tin

Đầu vào: Một ảnh cấp xám S , một chuỗi nhị phân M .

Đầu ra: Ảnh giấu tin I .

Bước 1: Tạo ra các mẫu ảnh S_k theo phương trình (2. 1) từ ảnh ban đầu S sử dụng hai yếu tố lấy mẫu ($\Delta u, \Delta v$).

Bước 2: Xác định một tham chiếu phụ S_{Ref} để tối đa hóa sự tương quan không gian giữa các mẫu ảnh. Lựa chọn S_{Ref} từ các phương trình (2. 2). Ví dụ, khi $\Delta u = \Delta v = 3$, S_5 được xác định là tham chiếu. Nó được định nghĩa:

$$S_{\text{Ref}} = \left(\text{Round} \left(\frac{\Delta u}{2} - 1 \right) \right) \times \Delta v + \text{Round} \left(\frac{\Delta v}{2} \right) \quad (2.2)$$

Bước 3 : Tạo ra ảnh khác biệt sai phân giữa mẫu S_{Ref} và ảnh mẫu khác ký hiệu S_{Des} , ta được:

$$D_{\text{Ref-Des}}(k_1, k_2) = S_{\text{Ref}}(k_1, k_2) - S_{\text{Des}}(k_1, k_2) \quad (2.3)$$

Trong đó : $0 \leq k_1 \leq M/\Delta v - 1$, $0 \leq k_2 \leq N/\Delta u - 1$.

Bước 4 : trước khi làm rỗng một hệ số cột hệ số H của biểu đồ tần suất sai phân theo mức nhúng L , với $H = -255, \dots, 255$. Phụ thuộc vào mức độ giấu sẽ chọn L sao cho phù hợp vì L ảnh hưởng đến chất lượng ảnh và khả năng mang thông tin. Để đạt được điều này các hệ số sai phân âm và không âm bên ngoài miền lựa chọn giấu tin (tức là ngoài $[-L, L]$ sẽ bị dịch sang trái và sang phải). Khi đó chỉ những điểm ảnh trong các mẫu ảnh con bị thay đổi. các giá trị hệ số sai phân H_s được tính toán như sau:

$$H_s = \begin{cases} H + L + 1 & \text{nu } H \geq L + 1 \\ H - L - 1 & \text{nu } H \leq -L - 1 \end{cases} \quad (2.4)$$

Điều này có nghĩa:

$$D'_{\text{Ref-Des}}(k_1, k_2) = S_{\text{Ref}}(k_1, k_2) - S'_{\text{Des}}(k_1, k_2) \quad (2.5)$$

Với :

$$S'_{\text{Des}}(k_1, k_2) = \begin{cases} S_{\text{Des}}(k_1, k_2) - (L + 1) & \text{nu } H \geq L + 1 \\ S_{\text{Des}}(k_1, k_2) + (L + 1) & \text{nu } H \leq -L - 1 \end{cases} \quad (2.6)$$

Bước 5 : Nhúng tin $w(n)$ bằng cách thay đổi H_s , với $w(n) \in \{0, 1\}$. Các hệ số sai phân D' được quét. Khi một điểm ảnh với giá trị sai phân bằng $-L$ hoặc $+L$ thì $w(n)$ được nhúng. Quá trình này được lặp đi lặp lại cho đến khi có không có điểm ảnh với giá trị sai phân bằng $\pm L$. Các bước nhúng được thực hiện cho đến khi $L < 0$. Tương tự như vậy, chúng ta chỉ sửa các điểm ảnh trong mẫu con. Nhưng thông tin có thể tính toán như sau:

$$D''_{\text{Ref-Des}}(k_1, k_2) = S_{\text{Ref}}(k_1, k_2) - S''_{\text{Des}}(k_1, k_2) \quad (2.7)$$

Trong đó :

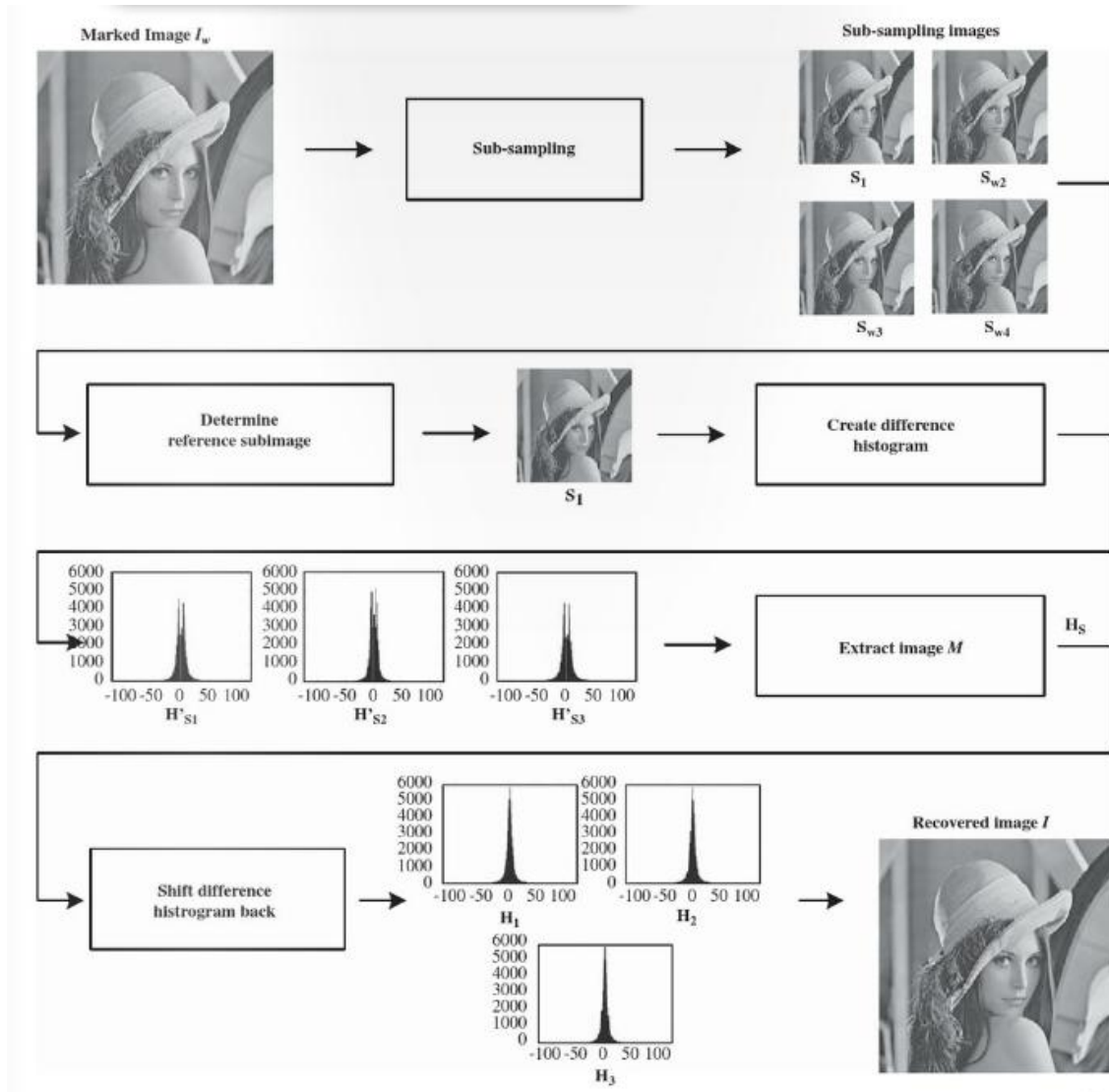
$$S''_{Des}(k_1, k_2) = \begin{cases} S'_{Des}(k_1, k_2) + (L + 1) & \text{nu } D' = -L, w(n) = 1 \\ S'_{Des}(k_1, k_2) - (L + 1) & \text{nu } D' = L, w(n) = 1 \\ S'_{Des}(k_1, k_2) + L & \text{nu } D' = -L, w(n) = 0 \\ S'_{Des}(k_1, k_2) - L & \text{nu } D' = L, w(n) = 0 \end{cases} \quad (2.8)$$

Đối với: $L > 0$

$$S''_{Des}(k_1, k_2) = \begin{cases} S'_{Des}(k_1, k_2) - 1 & \text{nu } D' = 0, w(n) = 1 \\ S'_{Des}(k_1, k_2) & \text{nu } D' = 0, w(n) = 0 \end{cases} \quad (2.9)$$

Đối với: $L = 0$

Bước 6 : Cuối cùng, có được I_w ảnh đã giấu tin thông qua phép nghịch đảo lấy mẫu từ mẫu con S_{Ref} và các mẫu con S''_{Des} bị chỉnh sửa.



Hình 2. 2. Sơ đồ quá trình giấu tin

2. 2. 3 Thuật toán tách tin

Đầu vào: Ảnh Giấu tin I.

Đầu ra: Ảnh khôi phục và chuỗi nhị phân M.

Bước 1 : Từ hai yếu tố lấy mẫu $((\Delta u, \Delta v))$ và mức nhúng L từ LSB của các điểm ảnh được lựa chọn bằng cách sử dụng khóa bí mật.

Bước 2 : Tạo ra phụ mẫu con theo phương trình (2. 1). Từ ảnh I_s được sử dụng các yếu tố lấy mẫu ở bước 1.

Bước 3 : Xác định tham chiếu phụ S_{ref} từ phương trình (2. 2)

Bước 4 : Tính sai phân của ảnh giữa S_{ref} và S_w . Theo (2. 10).

$$S_{ref-w}(k_1, k_2) = S_{ref}(k_1, k_2) - S_w(k_1, k_2) \quad (2.10)$$

Trong đó : $0 \leq k_1 \leq M/\Delta v - 1$, $0 \leq k_2 \leq N/\Delta u - 1$, and $w = 1, \dots, \Delta u \times \Delta v$, $w \neq Ref$.

Bước 5 : Tách các thông điệp giấu w (n) từ sai phân ảnh. Quá trình tách tin là nghịch đảo của quá trình giấu tin. Sử dụng một biến L' được đặt giá trị là 0, sai phân ảnh D được quét. Khi một điểm ảnh với giá trị sai phân bằng 1 thì bit 1 được lấy ra. Nếu các điểm ảnh với giá trị sai phân bằng 0 thì bit 0 được lấy ra. Quá trình này được lặp đi lặp lại cho đến khi không có điểm ảnh với giá trị sai phân bằng 0 và 1. Sau đó, L' tăng lên 1, các giá trị sai phân được quét một lần nữa, và thông điệp được tách ra bằng cách sử dụng quy tắc sau đây:

$$W(n) = \begin{cases} 0 & \text{nu } D = 2L' \text{ hoc } - 2L' \\ 1 & \text{nếu } D = 2L' + 1 \text{ hoặc } - 2L' - 1 \end{cases} \quad (2.11)$$

Quá trình quét và giải nén này được thực hiện cho đến khi $L' > L$.

Bước 6: Hủy bỏ các tin đã giấu w (n) từ những sai phân.

$$D'_{Ref-w}(k_1, k_2) = S_{Ref}(k_1, k_2) - S'_w(k_1, k_2) \quad (2.12)$$

Trong đó :

$$S'_w(k_1, k_2) = \begin{cases} S_w(k_1, k_2) + 1 & \text{nếu } D(k_1, k_2) = 1 \\ S_w(k_1, k_2) & \text{ngược lại} \end{cases} \quad (2. 13)$$

Với $L' = 0$,

$$S'_w(k_1, k_2) = \begin{cases} S_w(k_1, k_2) + L' & \text{nu } D(k_1, k_2) = 2L' \\ S_w(k_1, k_2) + (L' + 1) & \text{nu } D(k_1, k_2) = 2L' + 1 \\ S_w(k_1, k_2) - L' & \text{nu } D(k_1, k_2) = -2L' \\ S_w(k_1, k_2) - (L' + 1) & \text{nu } D(k_1, k_2) = -2L' - 1 \end{cases} \quad (2.14)$$

Với $1 \leq L' \leq L$.

Bước 7: Chuyển các biểu đồ H_s của ảnh sai khác để được sai phân biểu đồ H ban đầu như sau:

$$H \begin{cases} H_s - L - 1 \text{ nu } H_s \geq 2L + 2 \\ H_s + L + 1 \text{ nu } H_s \leq -2L - 2 \end{cases} \quad (2.15)$$

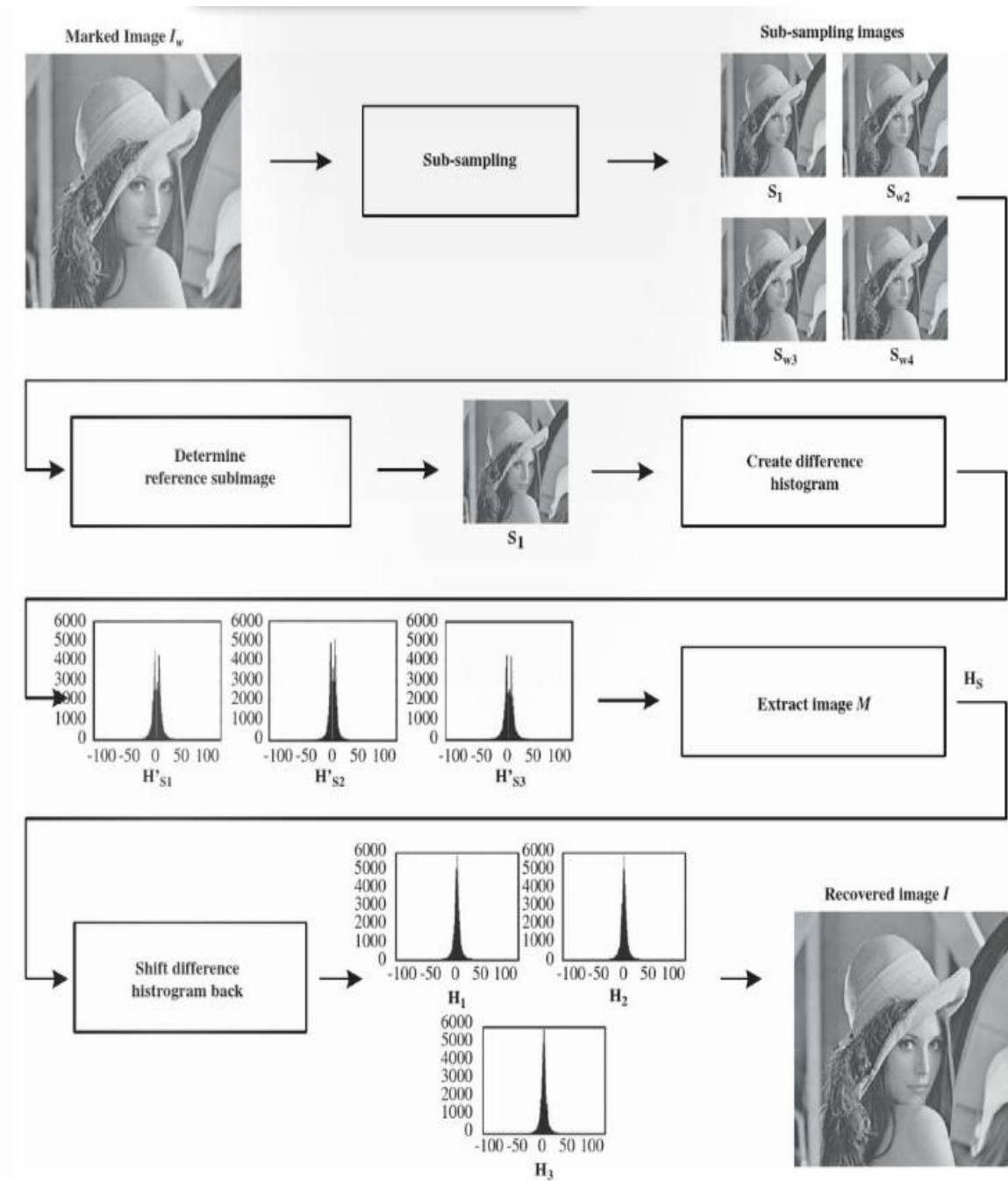
Hay :

$$S''_{\text{Ref-w}}(k_1, k_2) = S_{\text{Ref}}(k_1, k_2) - S''_w(k_1, k_2) \quad (2.16)$$

Trong đó :

$$S''_w(k_1, k_2) = \begin{cases} S'_w(k_1, k_2) + (L + 1) \text{ nếu } H_s \geq 2L + 2 \\ S'_w(k_1, k_2) - (L + 1) \text{ nếu } H_s \leq -2L - 2 \end{cases} \quad (2.17)$$

Bước 8 : Cuối cùng, ta được ảnh ban đầu thông qua biến đổi nghịch đảo tách mẫu từ S_{ref} và S''_w .



Hình 2. 3. Sơ đồ quá trình tách tin

2.3 Ví dụ minh họa

- Đầu vào:

➤ Ảnh có kích cỡ 6×6 :

167	167	168	153	154	158
165	165	167	153	152	154
166	163	168	154	155	156
167	163	168	166	166	160
165	165	166	165	160	162
165	163	168	161	163	159

➤ Thông tin cần giấu là chuỗi M = ‘TU’

- Đầu ra:

➤ Ảnh được giấu tin

- Các bước thực hiện:

Giấu tin dựa trên sự tương quan của miền không gian ảnh

Bước 1 : Chia ảnh thành các khối nhỏ với $\Delta u = \Delta v = 2$ ta được 4 khối mẫu con sau :

167	168	154	167	153	166	165	167	152	165	153	154
166	168	155	163	154	156	167	168	167	163	167	160
165	166	160	165	165	162	165	168	163	163	161	159

Bước 2: Chọn S_{ref} theo công thức (2. 2) ta được $S_{ref} = 1$

Bước 3: Ta được các mẫu sai phân sau:

0	15	-12	2	1	2	2	15	0
3	14	-1	-1	0	-12	3	1	-5
0	1	-2	0	-2	-3	2	5	1

Bước 4: Dịch chuyển các hệ số sai phân có giá trị không được phủ màu ở bước 3. Với $L=2$

0	18	-15
6	17	-1
0	1	-2

2	1	2
-1	0	-15
0	-2	-6

2	18	0
6	1	-8
2	8	1

Bước 5: Giấu tin: chuyển thông điệp “TU” sang chuỗi nhị phân ta được:

“TU” = 0111010001110101

Và các hệ số sau khi giấu tin:

0	18	-15
6	17	-2
1	1	-4

5	2	4
-3	0	-15
1	-5	-6

4	18	0
6	2	-8
5	8	3

Bước 6: Số bit thông điệp đã giấu là : 16 bit và kết quả ảnh sau khi giấu tin:

167	167	168	150	154	159
162	163	166	150	150	154
166	160	168	151	155	157
169	160	168	166	170	163
165	164	166	164	160	164
164	160	171	158	166	157

Tách tin :

Bước 1: Chia ảnh thành các khối nhỏ với $\Delta u = \Delta v = 2$ ta được 4 khối mẫu con sau :

167	168	154
166	168	155
165	166	160

167	150	169
160	151	157
164	164	164

162	166	149
169	168	170
164	171	166

163	150	154
160	165	163
160	158	157

Bước 2: Chọn S_{ref} theo công thức (2. 2) ta được $S_{ref} = 1$.

Bước 3: Ta được các mẫu sai phân sau:

0	18	-15
6	17	-2
1	2	-4

5	2	5
-3	0	-15
1	-5	-6

4	18	0
6	3	-8
5	8	3

Bước 4: Nhập số bit thông điệp đã giấu : 16 bit.

Bước 5: Trích xuất các thông điệp ẩn $w(n)$ từ mỗi hình ảnh khác biệt ta được chuỗi nhị phân :

MM= “0111010001110101”

Bước 6: Chuyển chuỗi nhị phân kí tự ta được thông điệp đã giấu :

“0111010001110101” = “TU”

Bước 7: Kết quả nhận được là thông điệp đã giấu “TU” và ảnh khôi phục :

167	167	168	153	154	158
165	165	167	153	152	154
166	163	168	154	155	156
167	163	168	166	166	160
165	165	166	165	160	162
165	163	168	161	163	159

CHƯƠNG 3: CÀI ĐẶT VÀ THỬ NGHIỆM

3. 1. Môi trường cài đặt

- Ngôn ngữ cài đặt: Ngôn ngữ lập trình Matlab phiên bản 7. 7
- Môi trường soạn thảo: Matlab phiên bản 7. 7
- Môi trường chạy chương trình: Môi trường giao diện Matlab phiên bản 7. 7
- Cấu hình tối thiểu để cài đặt Matlap:

+ Intel hoặc AMD x86 processor supporting SSE2

+ Windows XP SP2 x64, SP3, 7, 8 ...

+ Dung lượng ổ cứng từ 16 GB tới 32GB, bộ nhớ RAM tối thiểu 1GB

Ở chương 3 chúng ta sẽ cài đặt và thử nghiệm chương trình với 1 bài toán giấu tin trong ảnh cấp xám 8 bit như sau:

Đầu vào:

- Ảnh cấp xám 8 bit S có kích thước $M \times N$.
- Chuỗi thông điệp cần giấu.

Đầu ra:

- Ảnh cấp xám 8 bit I đã được giấu tin.

Chức năng giấu tin:

- Kỹ thuật giấu tin thuận nghịch bằng cách thay đổi nhỏ trên miền tần suất sai phân giữa các ảnh chia nhỏ của ảnh ban đầu. Kỹ thuật này khai thác sự tương quan giữa điểm ảnh với láng giềng của nó để tăng chất lượng ảnh và khả năng giấu tin. Từ kết quả thực nghiệm tác giả của các kỹ thuật chỉ ra hiệu quả của kỹ thuật bằng độ đo PSNR có giá trị cao.

- Giấu vào chuỗi kí tự: do người dùng nhập vào từ bàn phím.
- Giấu vào tệp văn bản: Chọn một tệp văn bản định dạng *. txt để giấu vào ảnh.

Chức năng tách tin:

- Tách tin trên bằng kỹ thuật giấu tin thuận nghịch trên miền tần suất sai phân giữa các mẫu con của ảnh đã được giấu từ trước.

- Tách chuỗi thông điệp đã giấu và lưu lại dưới dạng tập *. txt.

3. 2. Giao diện chương trình

3. 2. 1. *Giao diện chương trình chính*



Hình 3. 1. *Giao diện chương trình chính*

Đây là giao diện chính của chương trình, từ đây ta sẽ gọi đến các giao diện khác thông qua menu.

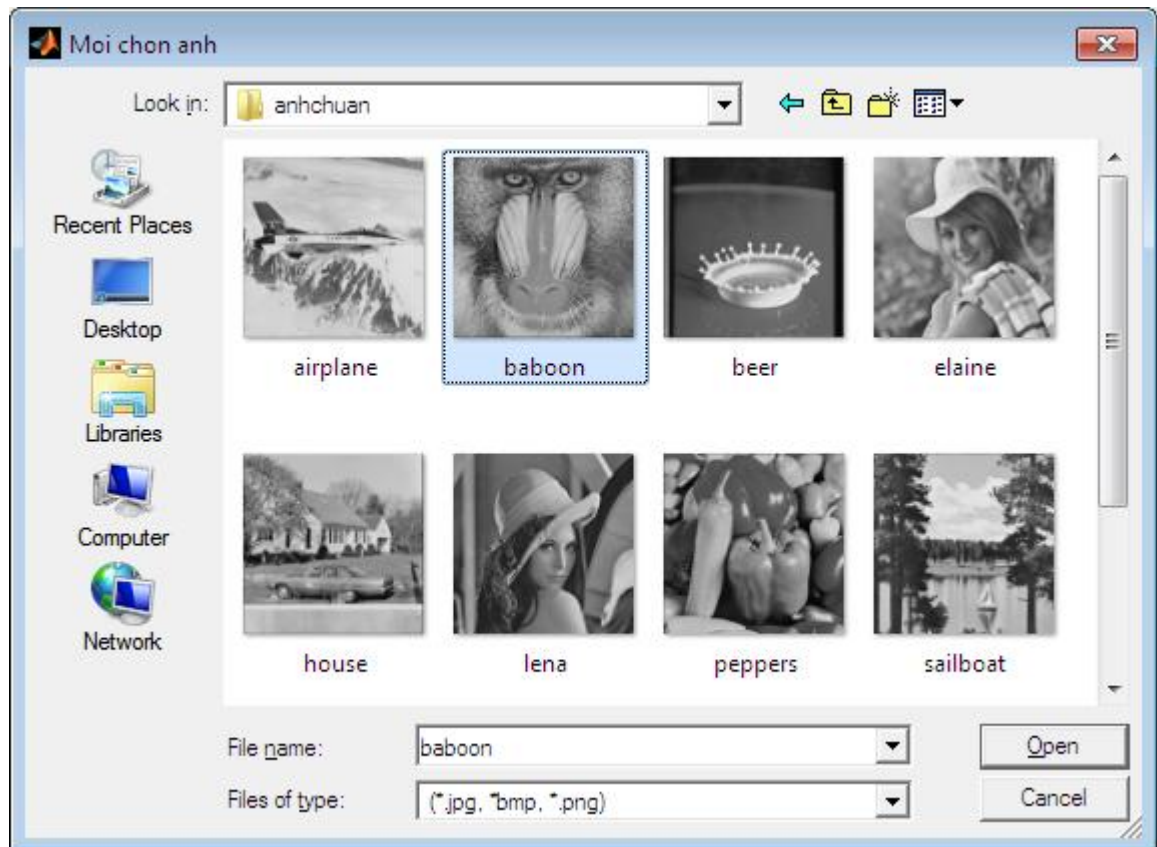
Từ menu chọn “GIẤU TIN” sẽ gọi đến giao diện:

3. 2. 2. Giao diện chức năng giấu tin



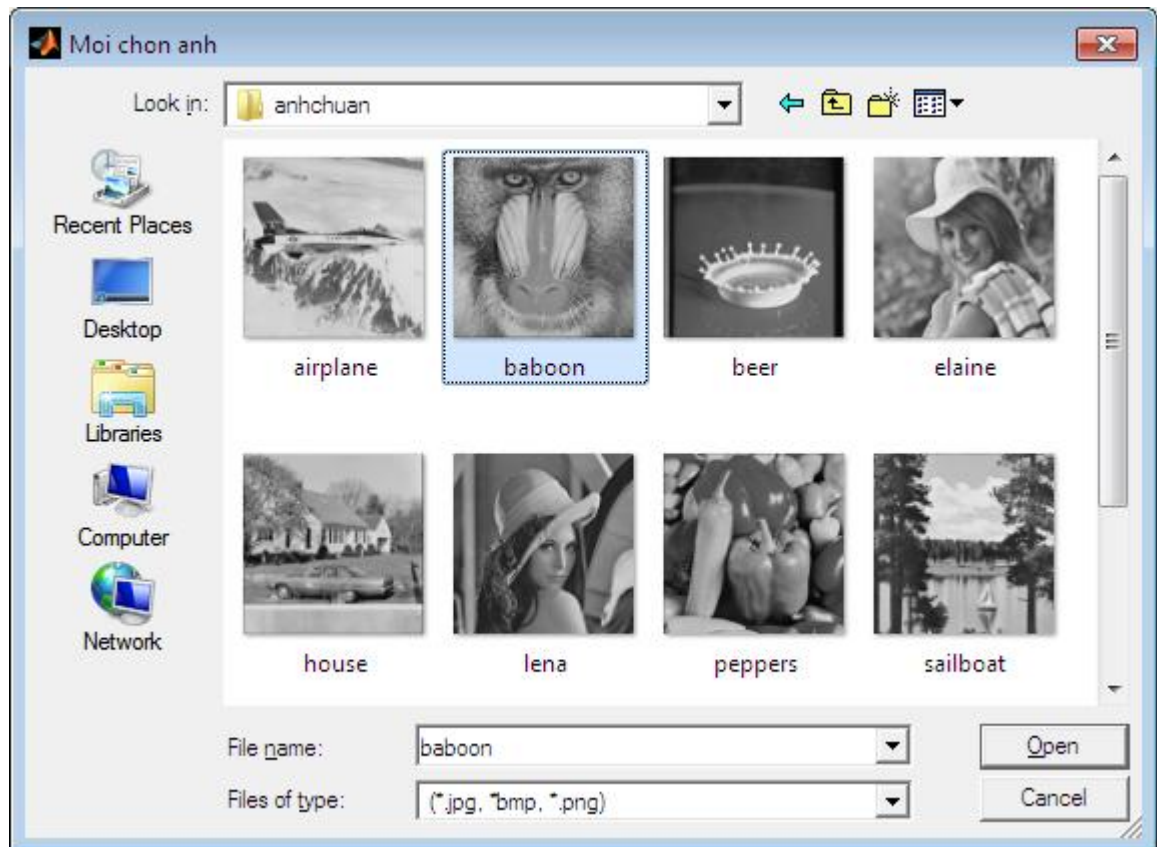
Hình 3. 2. *Giao diện chức năng giấu tin*

Từ giao diện chính của chương trình chúng ta chọn ảnh cần giấu tin bằng cách nhấn vào nút “NHẬP ẢNH”. Khi đó chương trình sẽ mở ra hộp thoại duyệt ảnh.



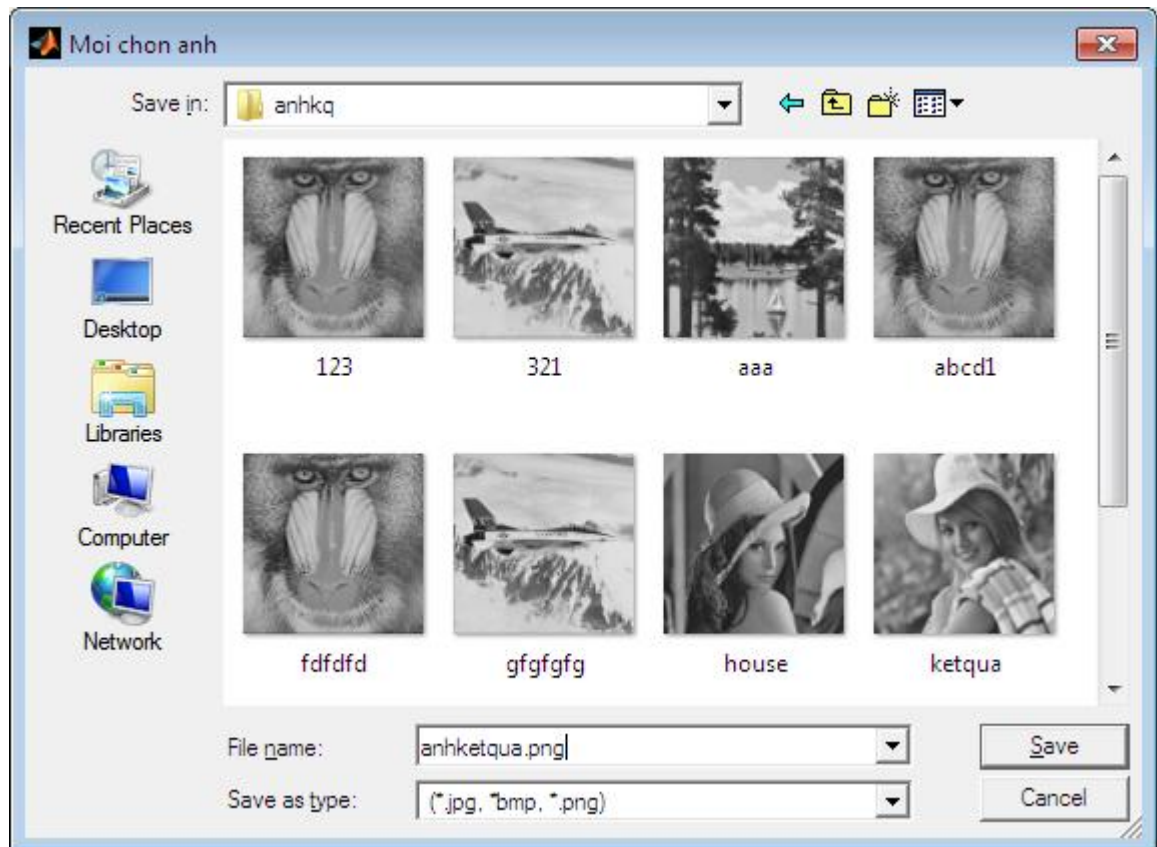
Hình 3. 3. Hộp thoại chọn ảnh cấp xám 8 bit cần giấu tin

Chúng ta sẽ chọn ảnh cấp xám 8 bit bất kì để thực hiện giấu tin vào ảnh đó. Sau khi chọn ảnh xong, ta nhập thông điệp vào từ bàn phím hoặc từ 1 tệp *.txt bất kì bằng cách kích vào nút “NHAP THONG DIEP” để giấu tin.



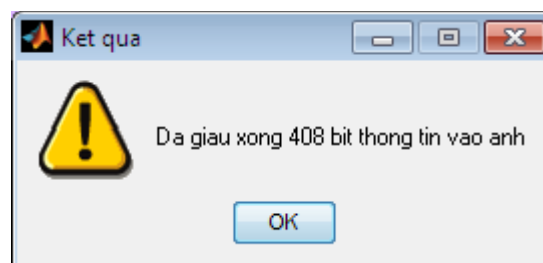
Hình 3. 4. Hộp thoại chọn thông điệp cần giấu

Chúng ta cần chọn nơi sẽ lưu ảnh kết quả sau khi đã giấu tin vào bằng cách chọn “LUU ANH GIAU TIN” từ giao diện.

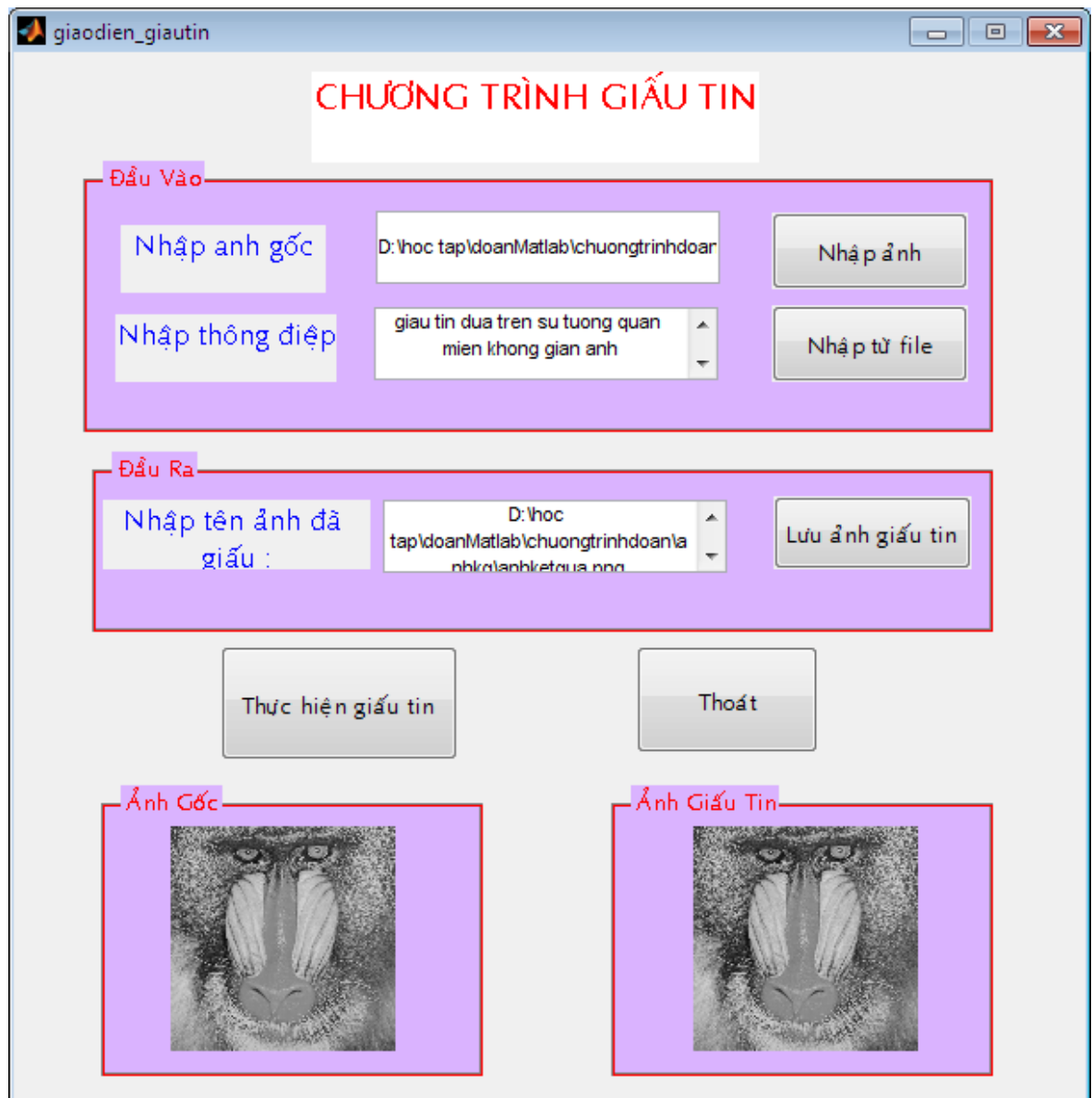


Hình 3. 5. Hộp thoại cho biết tên ảnh sau khi đã giấu tin

Sau khi đã lựa chọn xong đầu vào và đầu ra cho chương trình, chúng ta chọn nút “THUC HIEN GIAU TIN”. Chương trình sẽ thực hiện và đưa ra kết quả ảnh đã giấu tin và độ dài thông điệp qua 1 thông báo ngay trên giao diện của chương trình.

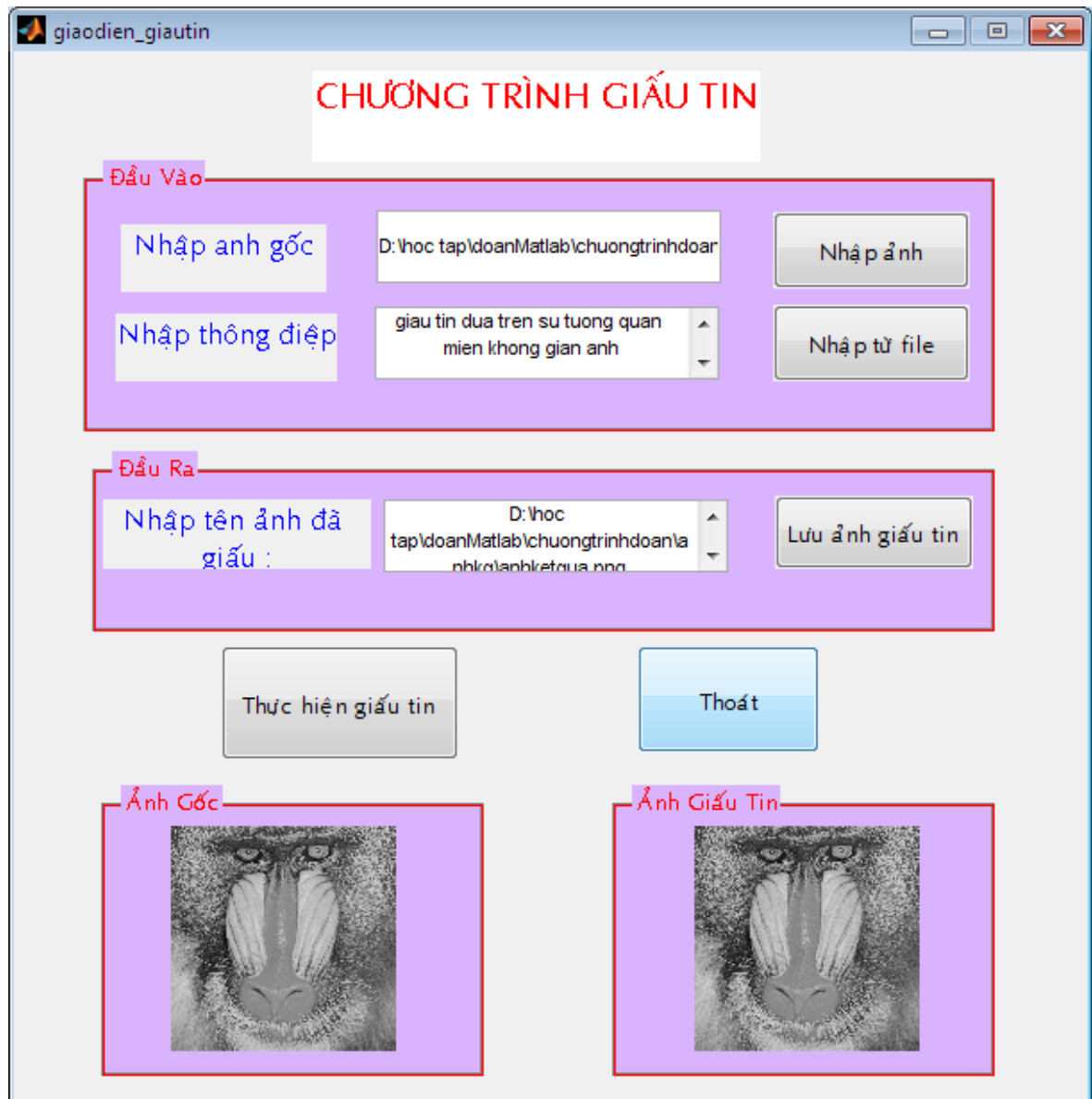


Hình 3. 6. Thông báo kết quả giấu tin



Hình 3. 7. Giao diện sau khi giấu tin

Sau khi hoàn thành mọi việc kích nút “THOAT” để thoát khỏi chương trình giấu tin.



Hình 3. 8. Giao diện kết thúc chương trình giấu tin

3. 2. 3. Giao diện chức năng tách tin

Từ menu của giao diện chính chọn “TÁCH TIN” sẽ gọi đến giao diện:



GIAO DIỆN TÁCH TIN

Đầu Vào

Tên ảnh giấu tin

Độ dài thông điệp

Đầu Ra

Thông điệp khôi phục

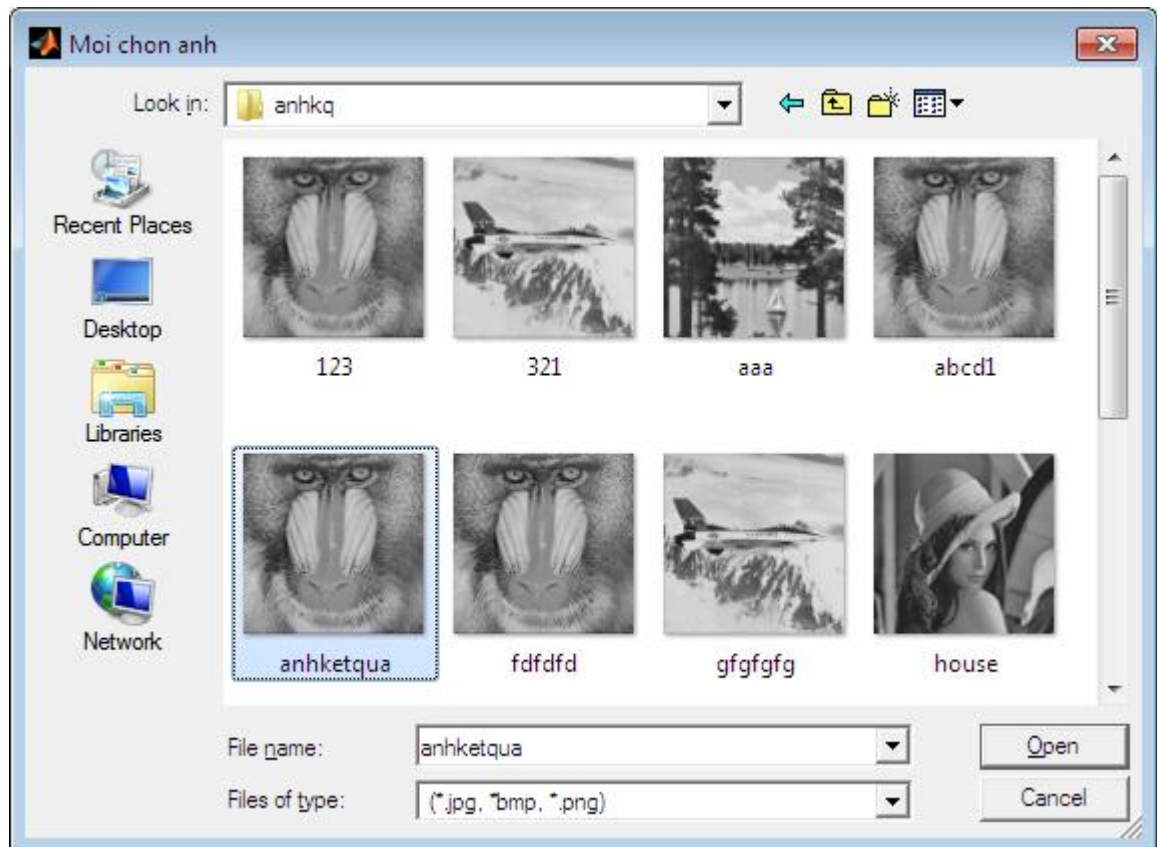
Tên ảnh khôi phục

Ảnh giấu tin

Ảnh tách tin

Hình 3. 9. Giao diện chức năng tách tin

Từ giao diện chính của chương trình chúng ta chọn ảnh cần giấu tin bằng cách nhấn vào nút “CHON ANH”. Khi đó chương trình sẽ mở ra hộp thoại duyệt ảnh.



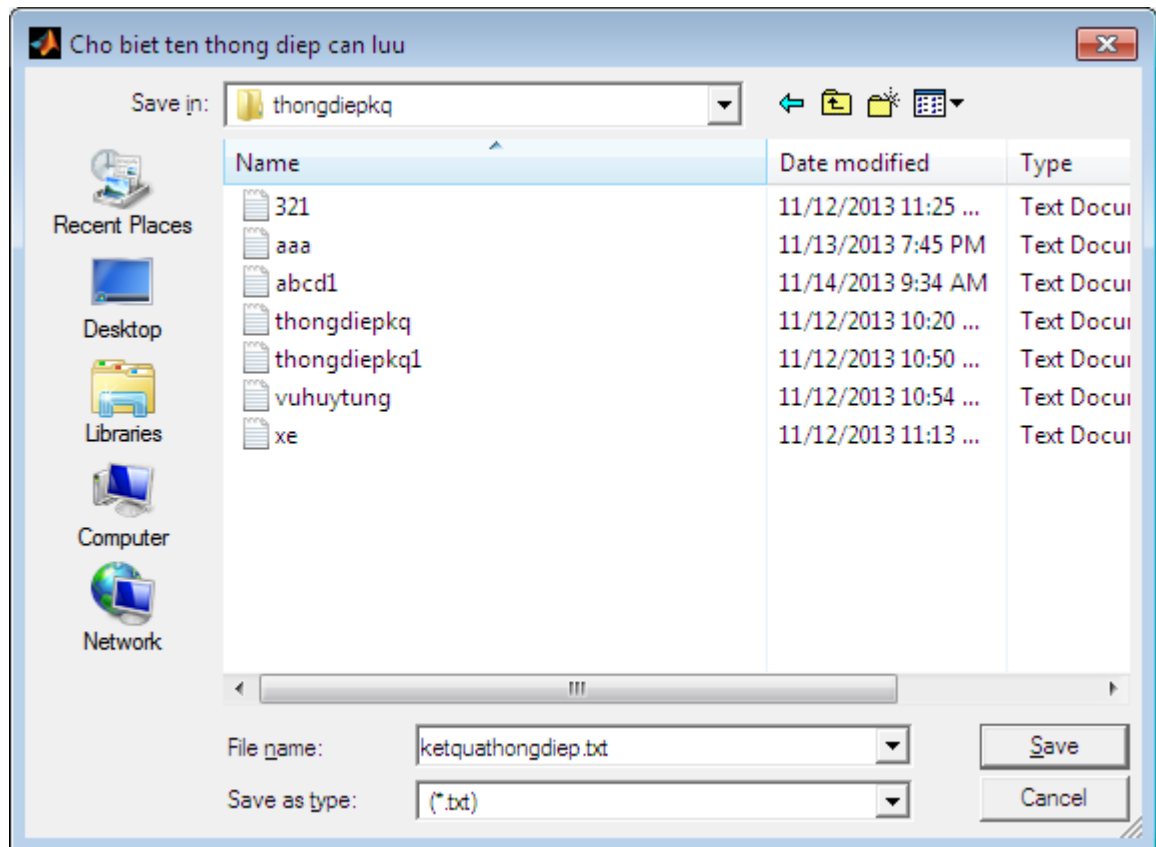
Hình 3. 10. Hộp thoại chọn ảnh cấp xám 8 bit cần tách tin

Chúng ta sẽ chọn ảnh cấp xám 8 bit đã giấu tin ở trên để thực hiện tách tin trong ảnh đó. Sau khi chọn ảnh cần tách xong ta sẽ nhập từ bàn phím độ dài thông điệp đã hiện ở bên giấu tin.



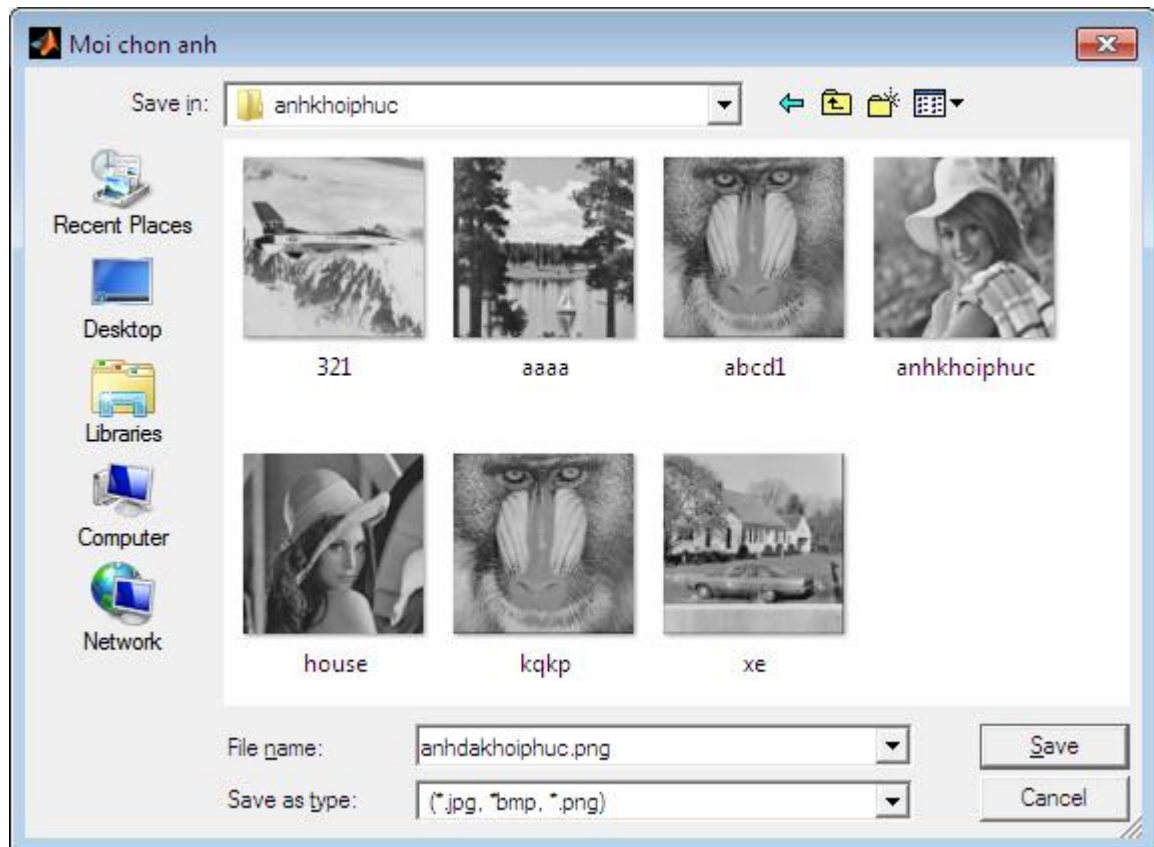
Hình 3. 11. Giao diện chức năng nhập độ dài thông điệp giấu tin từ bàn phím

Chúng ta cần chọn nơi sẽ lưu thông điệp kết quả sau khi đã tách tin vào bằng cách chọn “LƯU THÔNG ĐIỆP” từ giao diện.



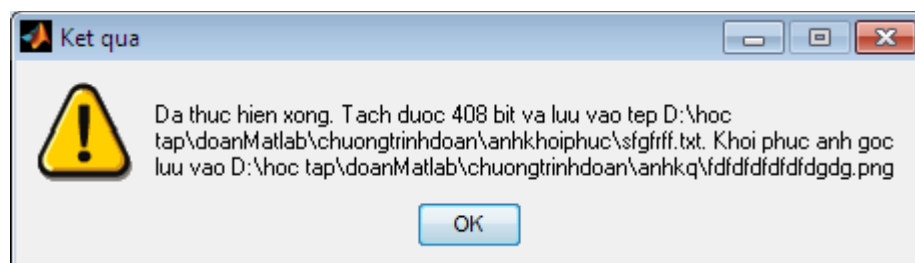
Hình 3. 12. Hộp thoại cho biết tên thông điệp sau khi đã tách tin

Chúng ta cần chọn nơi sẽ lưu ảnh kết quả sau khi đã tách tin vào bằng cách chọn “LUU ANH KHOI PHUC” từ giao diện.



Hình 3. 13. Hộp thoại cho biết tên ảnh sau khi đã tách tin

Sau khi đã lựa chọn xong đầu vào và đầu ra cho chương trình, chúng ta chọn nút “THUC HIEN TACH TIN”. Chương trình sẽ đưa ra cảnh báo và kết quả ảnh đã tách tin và thông điệp ngay trên giao diện của chương trình.



Hình 3. 14. Thông báo kết quả đã tách tin



Hình 3. 15. Giao diện sau khi tách tin

Sau khi hoàn thành mọi việc kích nút “THOAT” để thoát khỏi chương trình giấu tin.



Hình 3. 16. Giao diện kết thúc chương trình tách tin

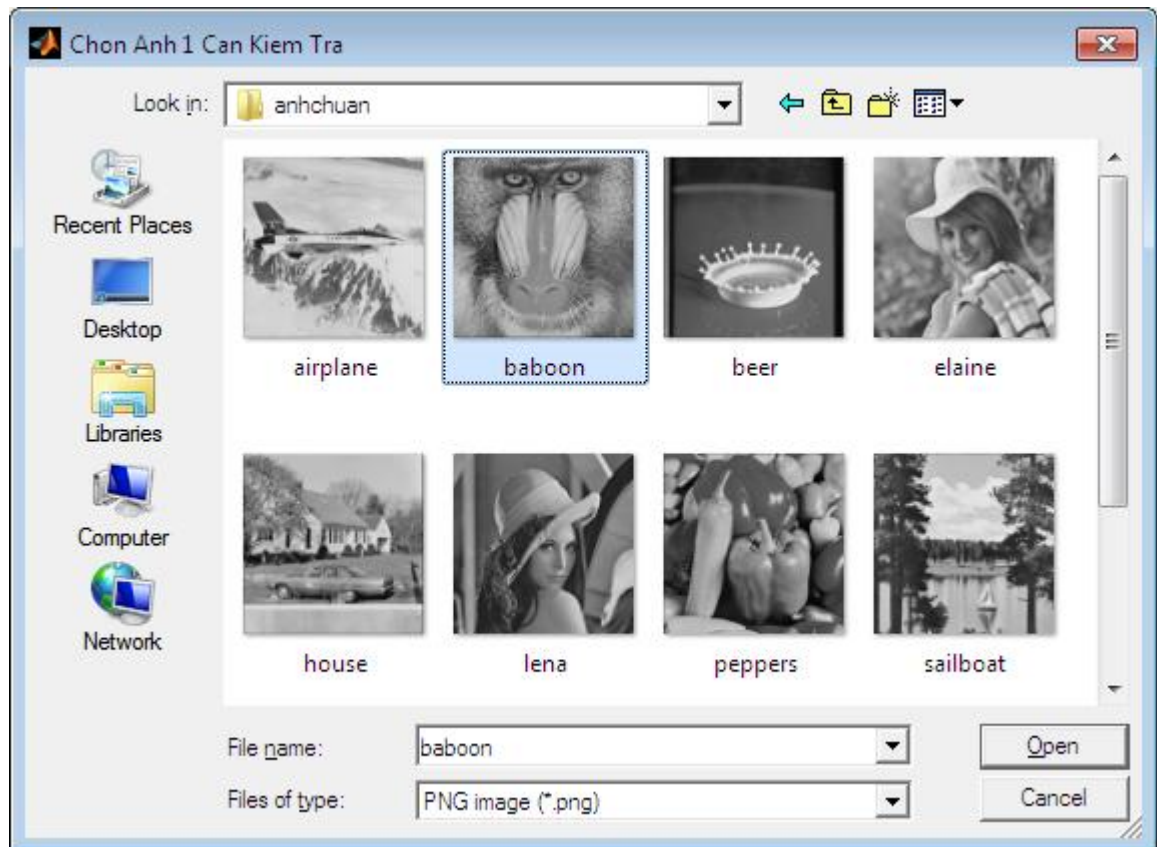
3. 2. 4. Giao diện chức năng đánh giá PSNR

Từ menu của giao diện chính chọn “PSNR” sẽ gọi đến giao diện:



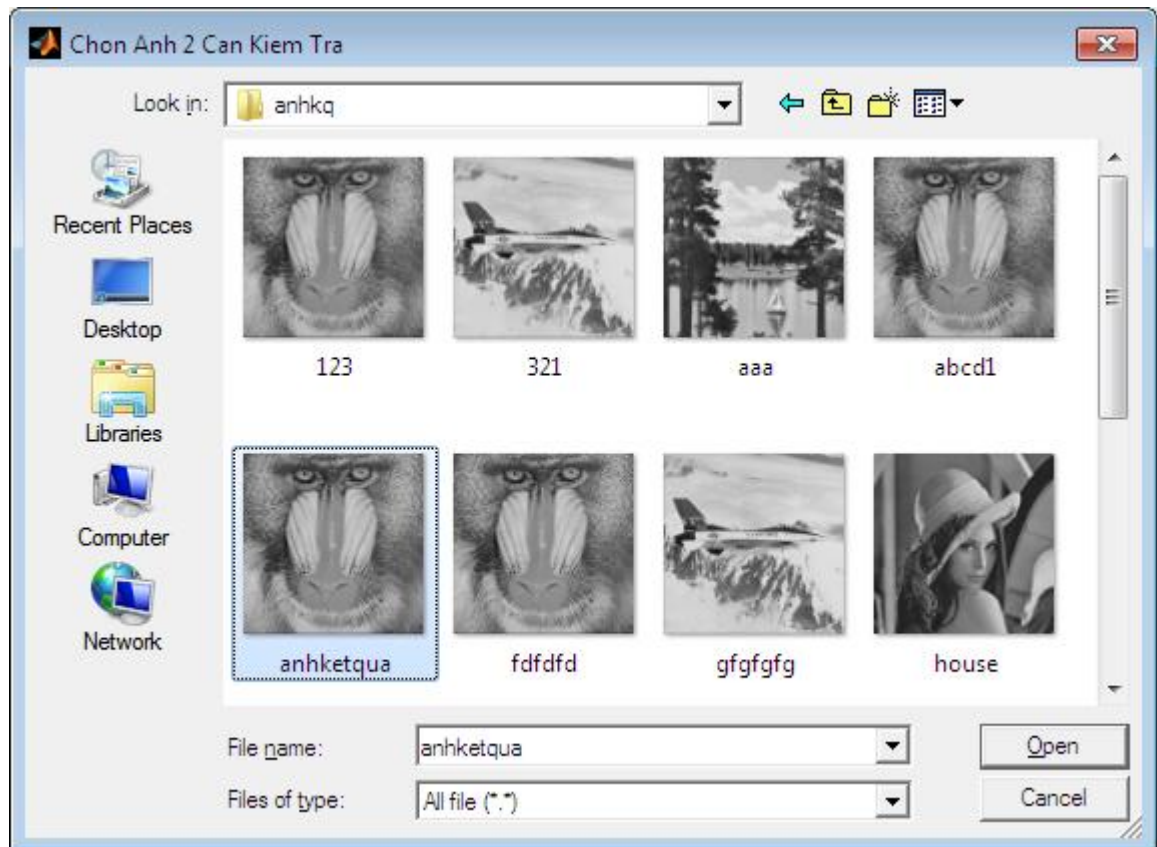
Hình 3. 17. Giao diện chức năng giấu tin

Từ giao diện chính của chương trình chúng ta chọn ảnh cần kiểm tra bằng cách nhấn vào nút “MO 1”. Khi đó chương trình sẽ mở ra hộp thoại duyệt ảnh.



Hình 3. 18. Hộp thoại chọn ảnh nhị phân cần kiểm tra

Từ giao diện chính của chương trình chúng ta chọn ảnh cần kiểm tra bằng cách nhấn vào nút “MO 2”. Khi đó chương trình sẽ mở ra hộp thoại duyệt ảnh.



Hình 3. 19. Hộp thoại chọn ảnh nhị phân cần kiểm tra

Từ giao diện chính của chương trình chúng ta nhấn vào nút “DANH GIA”. Khi đó chương trình sẽ cho ra kết quả đánh giá ngay trên giao diện trưng trình.



Hình 3. 20. Giao diện đánh giá PSNR

Thoát khỏi chương trình đánh giá PSNR bằng cách nhấn vào nút “THOAT” trên giao diện chương trình.



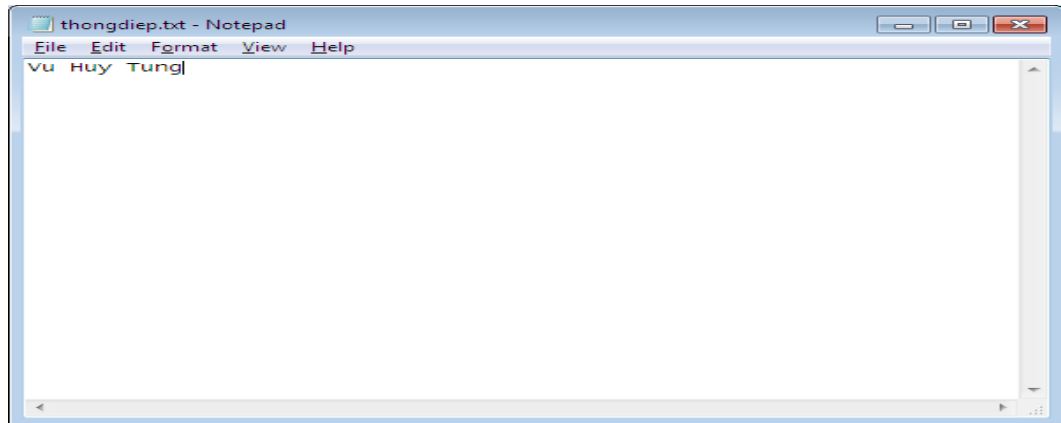
Hình 3. 21. Giao diện thoát chương trình đánh giá PSNR

3. 3. Kết quả thực nghiệm và nhận xét

3. 3. 1. Kết quả thực nghiệm

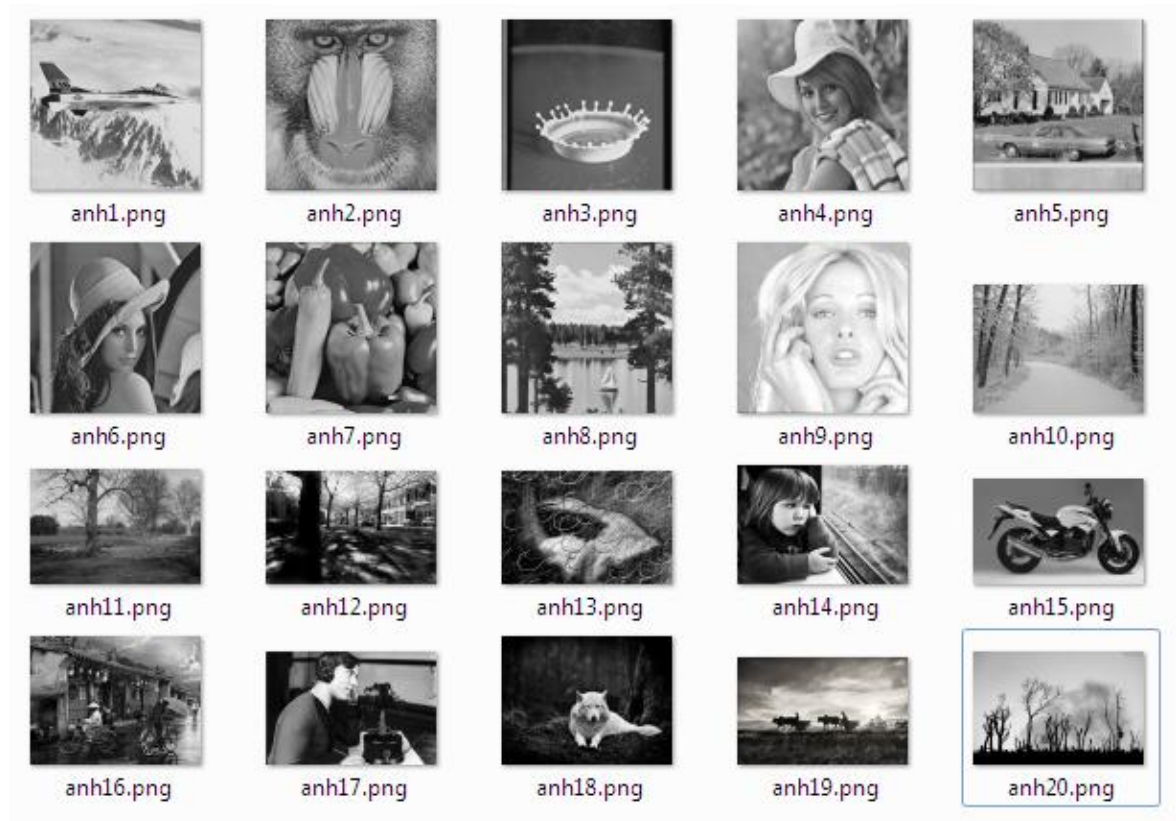
Thực nghiệm này sẽ đưa ra khả năng giấu tin khi sử dụng kỹ thuật giấu tin thuận nghịch bằng cách thay đổi nhỏ trên miền tần suất sai phân giữa các ảnh chia nhỏ của ảnh ban đầu.

Trường hợp 1. Giấu ít thông điệp: chuỗi gồm 12 kí tự.



Hình 3. 22. Chuỗi thông điệp cần giấu ít kí tự

Thực hiện bằng thuật toán giấu thông tin dựa vào sự tương quan của miền không gian ảnh với thông điệp có độ dài 12 bit (Hình 3. 21) ta được tập ảnh đã giấu tin (Hình 3. 23).



Hình 3. 23 Ảnh trước khi giấu tin



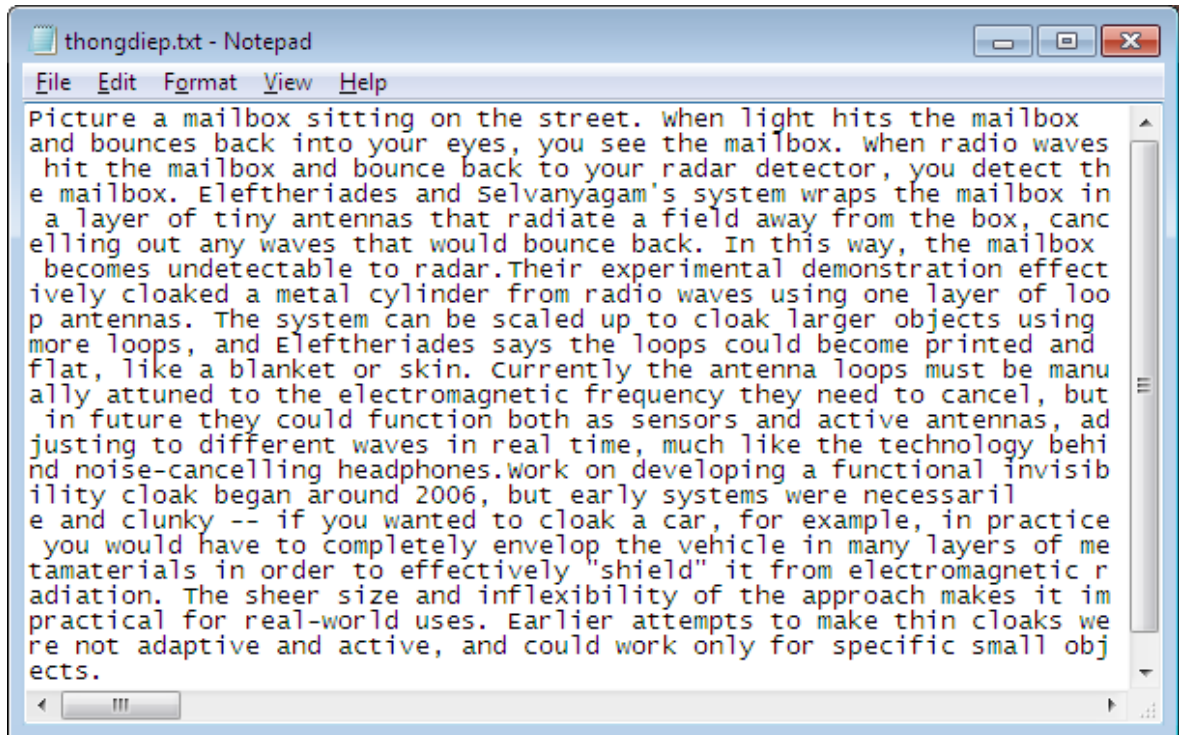
Hình 3. 24. Ảnh sau khi giấu tin

Thực hiện đánh giá PSNR giữa tập ảnh ban đầu với tập ảnh đã giấu tin bằng thuật toán giấu tin dựa trên sự tương quan của miền không gian ảnh ta được kết quả theo Bảng 3. 1

Bảng 3. 1. Kết quả đánh giá PSNR.

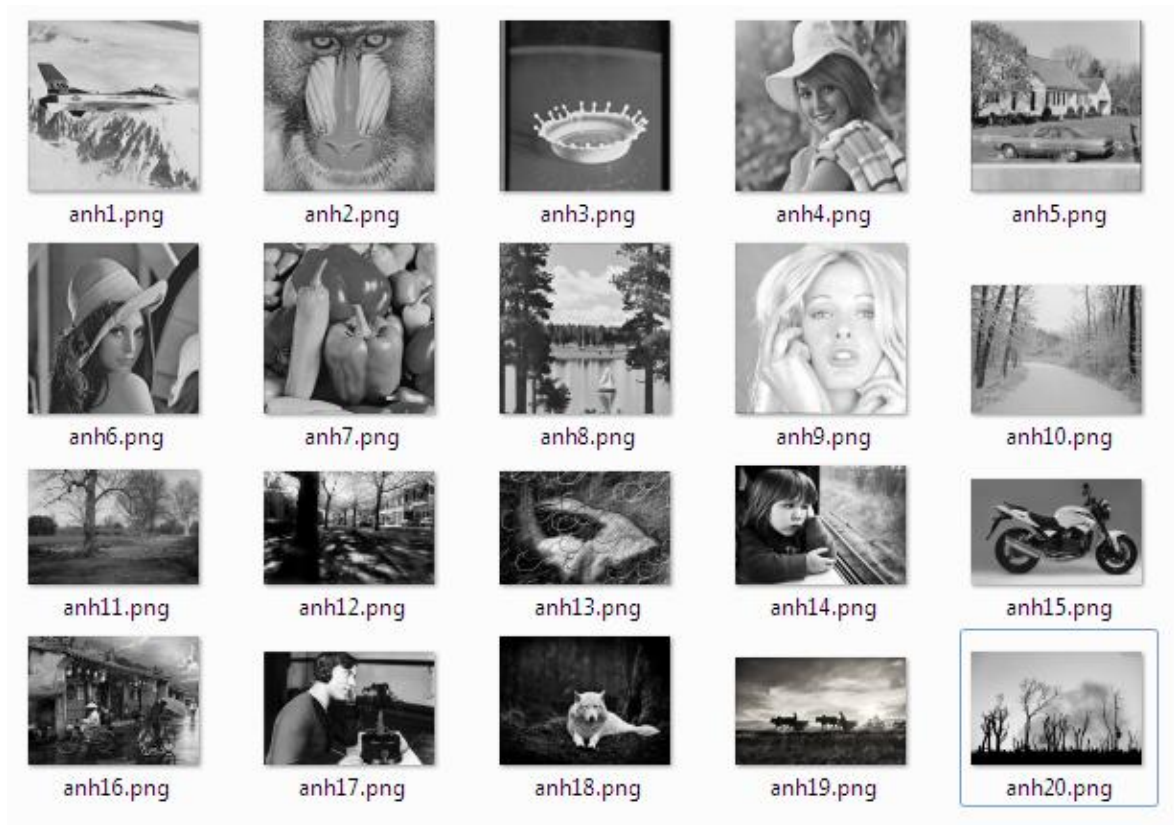
Tên ảnh(kích cỡ ảnh)	Đánh giá PSNR(dB)
anh1. png(768x512)	39. 8107 dB
anh2. png(768x512)	38. 7137 dB
anh3. png(768x512)	40. 4223 dB
anh4. png(768x512)	39. 5546 dB
anh5. png(768x512)	41. 1143 dB
anh6. png(768x512)	39. 3451 dB
anh7. png(768x512)	40. 5367 dB
anh8. png(512x512)	36. 3578 dB
anh9. png(512x512)	35. 9858 dB
anh10. png(512x512)	36. 5632 dB
anh11. png(512x512)	37. 9843 dB
anh12. png(512x512)	36. 7635 dB
anh13. png(512x512)	36. 4764 dB
anh14. png(512x512)	37. 8464 dB
anh15. png(512x512)	35. 9544 dB
anh16. png(512x512)	37. 4654 dB
anh17. png(512x512)	36. 4464 dB
anh18. png(512x512)	36. 5843 dB
anh19. png(512x512)	37. 4885 dB
anh20. png(512x512)	35. 4218 dB
Giá trị trung bình	37. 8418 dB

Trường hợp 2. Giấu nhiều thông điệp

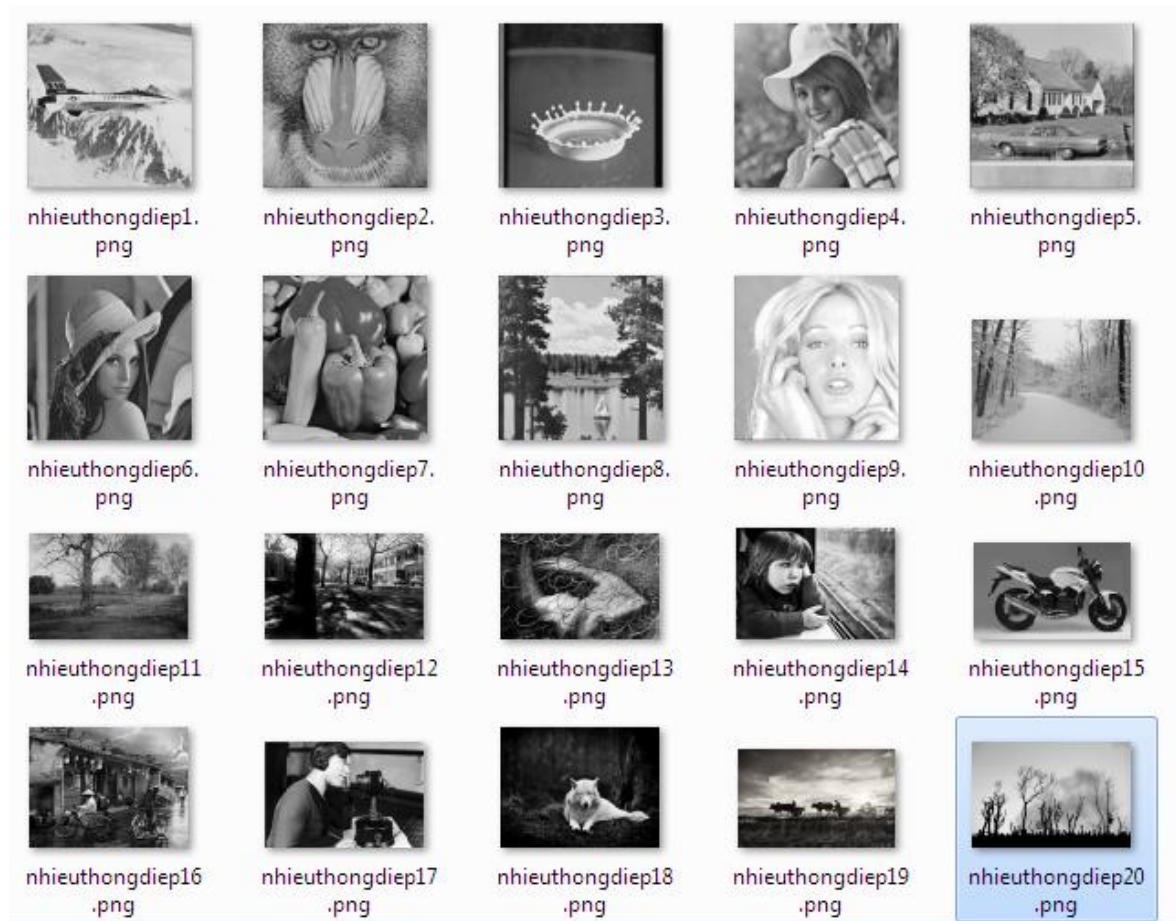


Hình 3. 25. Chuỗi thông điệp cần giấu nhiều kí tự

Thực hiện bằng thuật toán giấu thông tin dựa vào sự tương quan của miền không gian ảnh với thông điệp có độ dài 12 bit (Hình 3. 24) ta được tập ảnh đã giấu tin (Hình 3. 26).



Hình 3. 26. Ảnh trước khi giấu tin



Hình 3. 27. Ảnh sau khi giấu tin

Thực hiện đánh giá PSNR giữa tập ảnh ban đầu với tập ảnh đã giấu tin bằng thuật toán giấu tin dựa trên sự tương quan của miền không gian ảnh ta được kết quả theo Bảng 3. 2.

Bảng 3. 2. Kết quả đánh giá PSNR.

Tên ảnh(kích cỡ ảnh)	Đánh giá PSNR(dB)
anh1. png(768x512)	65. 4455 dB
anh2. png(768x512)	65. 5456 dB
anh3. png(768x512)	64. 4337 dB
anh4. png(768x512)	63. 4434 dB
anh5. png(768x512)	66. 1143 dB
anh6. png(768x512)	64. 3451 dB
anh7. png(768x512)	65. 5357 dB
anh8. png(512x512)	60. 3578 dB
anh9. png(512x512)	61. 9858 dB
anh10. png(512x512)	60. 6562 dB
anh11. png(512x512)	61. 9523 dB
anh12. png(512x512)	59. 3365 dB
anh13. png(512x512)	62. 4364 dB
anh14. png(512x512)	60. 4334 dB
anh15. png(512x512)	61. 4544 dB
anh16. png(512x512)	62. 3254 dB
anh17. png(512x512)	63. 4354 dB
anh18. png(512x512)	61. 4343 dB
anh19. png(512x512)	63. 4685 dB
anh20. png(512x512)	61. 4458 dB
Giá trị trung bình	62. 7793 dB

Trường hợp 3. Đánh giá PSNR giữa ảnh gốc và ảnh gốc khôi phục được từ ảnh đã giấu tin (trong phần tách tin).



Hình 3. 28. Ảnh gốc



Hình 3. 29. Ảnh tách tin

Thực hiện đánh giá PSNR giữa tập ảnh ban đầu với tập ảnh đã tách tin bằng thuật toán giấu tin dựa trên sự tương quan của miền không gian ảnh ta được kết quả theo Bảng 3. 2.

Bảng 3. 3. Kết quả đánh giá PSNR.

Tên ảnh(kích cỡ ảnh)	Đánh giá PSNR(dB)
anh1. png(768x512)	81. 3163 dB
anh2. png(768x512)	81. 5456 dB
anh3. png(768x512)	82. 4337 dB
anh4. png(768x512)	83. 4434 dB
anh5. png(768x512)	82. 1143 dB
anh6. png(768x512)	81. 3451 dB
anh7. png(768x512)	82. 5357 dB
anh8. png(512x512)	79. 3578 dB
anh9. png(512x512)	79. 9858 dB
anh10. png(512x512)	78. 6562 dB
anh11. png(512x512)	78. 3253 dB
anh12. png(512x512)	79. 3365 dB
anh13. png(512x512)	80. 4634 dB
anh14. png(512x512)	78. 4334 dB
anh15. png(512x512)	79. 4564 dB
anh16. png(512x512)	79. 3554 dB
anh17. png(512x512)	78. 4434 dB
anh18. png(512x512)	79. 4443 dB
anh19. png(512x512)	78. 4685 dB
anh20. png(512x512)	80. 4458 dB
Giá trị trung bình	80. 2453 dB

3. 3. 2. Nhận xét

Với kết quả thử nghiệm thu được, dù chuỗi thông điệp nhỏ hay lớn quan sát bằng mắt thường đều không thể phân biệt được ảnh đã giấu tin hay chưa giấu tin, ở trường hợp giấu chuỗi thông điệp nhỏ giá trị PSNR trung bình đạt được rất là cao xấp xỉ 37. 8418 dB. Còn ở trường hợp giấu chuỗi thông điệp lớn giá trị PSNR trung bình ở mức độ khá là cao xấp xỉ 62. 7793 dB. Có thể thấy phương pháp giấu tin trong ảnh này cho 1 kết quả và chất lượng hết sức cao. Nhưng sau khi đã có lượng thông tin được giấu vào trong ảnh gốc, nó sẽ để lại dù nhiều, dù ít. Những dấu vết khác với ảnh gốc ban đầu có thể thấy ở bảng 3. 3 khi đánh giá PSNR giữa ảnh gốc và ảnh khôi phục nhận được kết quả xấp xỉ 80. 2453 dB và không 1 ảnh nào đạt 100 dB để giống y hệt ảnh gốc ban đầu.

Kết quả thử nghiệm trong bảng 3. 1, 3. 2 cho thấy khả năng giấu tin của mỗi ảnh khác nhau là khác nhau. Những ảnh có cùng kích cỡ khả năng giấu của những ảnh đó nằm trong 1 khoảng giá trị và xấp xỉ bằng nhau. Điều đó chứng tỏ khả năng giấu tin phụ thuộc vào giá trị điểm ảnh của ảnh.

Thời gian xử lý giấu tin phụ thuộc lớn vào dữ liệu đầu vào như kích thước ảnh gốc, thông điệp giấu lớn hay nhỏ.

Qua thử nghiệm nhận thấy kỹ thuật giấu tin dựa trên sự tương quan của miền không gian ảnh có những ưu nhược điểm sau:

- Ưu điểm:

+ Khả năng bảo mật cao do độ dài thông điệp do người nhận và người gửi biết với nhau. Phải đúng độ dài đó mới có thể lấy được thông tin.

+ Độ nhiễu nhỏ, đạt chất lượng khá cao khi giấu lượng bit nhỏ và chấp nhận được khi giấu lượng bit lớn.

- Nhược điểm:

+ Sau khi đã có lượng thông tin được giấu vào trong ảnh gốc, nó sẽ để lại dù nhiều, dù ít những dấu vết khác với ảnh gốc ban đầu.

KẾT LUẬN

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. Đồ án đã trình bày một số khái niệm liên quan đến kỹ thuật giấu tin thuận nghịch, cũng như trình bày kỹ thuật giấu tin dựa trên sự tương quan của miền không gian ảnh.

Với kỹ thuật giấu tin dựa trên sự tương quan của miền không gian ảnh cho ảnh chất lượng cao trên ảnh cấp xám 8 bit thì tính vô hình của thông tin sau khi giấu được đảm bảo. Về mặt lý thuyết thì sau khi đã có lượng thông tin được giấu vào trong ảnh gốc, nó sẽ để lại dù nhiều, dù ít những dấu vết khác với ảnh gốc ban đầu. Dùng phương pháp đánh giá PSNR để đánh giá chất lượng ảnh trước và sau khi giấu tin kết quả đạt được là rất cao nếu giấu lượng bit nhỏ, và chấp nhận được nếu giấu lượng bit lớn. Như vậy kỹ thuật giấu tin dựa trên sự tương quan của miền không gian ảnh đã cho những kết quả hết sức tích cực.

Tuy nhiên, giấu tin mật là vấn đề phức tạp, cộng với khả năng và kinh nghiệm còn hạn chế nên em còn gặp một số khó khăn trong việc tìm hiểu nghiên cứu các kỹ thuật giấu tin thuận nghịch.

Vì vậy em rất mong nhận được sự đóng góp ý kiến quý báu của các thầy cô giáo cũng như bạn bè để báo cáo của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

TÀI LIỆU THAM KHẢO

- [1]. C. W. Honsinger, P. Jones, M. Rabbani, and J. C. Stoffel (1999), “Lossless recovery of an original image containing embedded data”, *US Patent application*, Docket no: 77102/E-D.
- [2]. Ni, Z., Shi, Y., Ansari, N., Su, W. (2003), “Reversible data hiding”, *Proc. ISCAS 2003*, pp. 912–915.
- [3]. J. H. Hwang, J. W. Kim, and J. U. Choi (2006), “A Reversible Watermarking Based on Histogram Shifting”, *IWDW 2006*, pp. 384-361.
- [4]. J. Tian (2002), “Wavelet Based Reversible Watermarking for Authentication”, *In Proc. Security and Watermarking of MultimediaContents IV*, Electronic Imaging 2002, Vol. 4675, pp. 679-690.
- [5]. K. S. Kim, M. J. Lee, H. Y. Lee, and H. K. Lee, Reversible data hiding exploiting spatial correlation between sub-sampled images, *Journal of Pattern Recognition*, vol. 42, pp. 3083-3096, 2009.
(<http://www.yumpu.com/en/document/view/7593070/reversible-data-hiding-exploiting-spatial-heung-kyu-lee>)