BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG
-------o0o-------

# ĐỒ ÁN TỐT NGHIỆP

NGÀNH CÔNG NGHỆ THÔNG TIN

HẢI PHÒNG 2013

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-------o0o-------

# PHƯƠNG PHÁP GIẤUTIN THUẬN NGHỊCH CHO ẢNH ĐÃ MÃ HÓA

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Công nghệ Thông tin

HẢI PHÒNG - 2013

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-------o0o-------

# PHƯƠNG PHÁP GIẤUTIN THUẬN NGHỊCH CHO ẢNH ĐÃ MÃ HÓA

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Công nghệ Thông tin

Sinhviênthựchiện:NGÔ VĂN HIỆP

Giáo viên hướng dẫn:TS. HỒ THỊ HƯƠNG THƠM

Mã số sinh viên:121315

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

CỘNG HÒA XA HỘI CHỦ NGHĨA VIỆT NAM
*Độclập - Tự do - Hạnhphúc*
-------o0o-------

# NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP

Sinhviên: NGÔ VĂN HIỆP Mã SV: 121315

Lớp: CT 1201 Ngành: CôngNghệThông Tin

Tênđềtài:Phươngpháp giấu tin thuậnnghịchchoảnhđãmãhóa

# NHIỆM VỤ ĐỀ TÀI

1. Nội dung vàcácyêucầucầngiảiquyéttrongnhiệmvụđềtàitốtnghiệp

a. Nội dung

- Tổngquanvềmãhóathông tin vàgiấu tin trongảnhsố.
- Tìmhiểuphươngpháp mãhóaảnh
- Kỹthuậtgiấu tin thuậnnghịchảnhđãmãhóa, táchthông tin, giảimãảnhmãhóa.
- Càiđặt, thửnghiệmchươngtrình

b. Cácyêucầucầngiảiquyết

a) Lýthuyết
   - Hiểuđượccấutrúccơbảncủaảnh Bitmap, phươngpháp mãhóaảnh
   - Nắmrõtổngquanvềkỹthuậtgiấu tin thuậnnghịchtrongảnh.
   - Hiểuvànắmrõkỹthuậtgiấu tin thuậnnghịchtrênảnhmãhóa, tách tin khôiphụcảnhmãhóa, giảimãảnhmãhóa.

b) Thựcnghiệm (chươngtrình)
   - CàiđặtđượckỹthuậtgiấubằngMatlab, thửnghiệmtrênmộttậpảnhđểcóthểđánhgiáđộtrựcquancủaảnhsaukhigiấu tin bằng PSNR, từđóđưaranhậnxétvềkỹthuậtgiấuápdụngchotậpảnhthửnghiệm.

2. Cácsốliệucầnthiếtđểthiếtkế, tínhtoán

- Tậpảnhđểthửnghiệm

# CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

**Ngườihướngdẫnthứnhất:**

Họvàtên:HồThịHươngThơm

Họchàm, họcvị: TiếnSĩ

Cơquancôngtác: TrườngĐạiHọcDânLậpHảiPhòng

Nội dung hướngdẫn:

**Ngườihướngdẫnthứhai:**

Họvàtên: …….

Họchàm, họcvị: …….

Cơquancôngtác: ……

Nội dung hướngdẫn:Phươngpháp giấu tin thuậnnghịchchoảnhđãmãhóa

Đềtàitốtnghiệpđượcgiaongàythángnăm 2013

Yêucầuphảihoànthànhtrướcngàythángnăm 2013

| | |
|---|---|
| Đãnhậnnhiệmvụ: Đ.T.T.N | Đãnhậnnhiệmvụ: Đ.T.T.N |
| Sinhviên | Cánbộhướngdẫn Đ.T.T.N |

TS. HồThịHươngThơm

*HảiPhòng, ngày .....tháng.....năm 2013*

HIỆU TRƯỞNG

***GS.TS.NGƯT TrầnHữuNghị***

# PHẦN NHẬN XÉT TÓM TẮT CỦA CÁN BỘ HƯỚNG DẪN

1. Tinhthầnthái độcủasinhviêntrongquátrìnhlàmđềtàitốtnghiệp:

.....

.....

.....

.....

.....

.....

2. Đánhgiáchấtlượngcủađềtàitốtnghiệp (so vớinội dung yêucầuđãđềratrongnhiệmvụđềtàitốtnghiệp)

.....

.....

.....

.....

.....

.....

.....

.....

3. Cho điểmcủacánbộhướngdẫn:

( Điểmghibằngsốvàchữ )

.....

.....

.....

Ngày.....tháng.....năm 2013

Cánbộhướngdẫnchính

*( Ký, ghirõhọtên )*

**PHẦN NHẬN XÉT ĐÁNH GIÁ CỦA CÁN BỘ CHẤM PHẢN BIỆN ĐỀ TÀI TỐT NGHIỆP**

**1. Đánhgiáchấtlượngđềtàitốtnghiệp (vềcácmặtnhưcơsởlýluận, thuyết minh chươngtrình, giátrịthựctế, ...)**

**2. Cho điểmcủacánbộphảnbiện**

( *Điểmghibằngsốvàchữ* )

.....

.....

.....

Ngày.....tháng.....năm 2013

Cánbộchấmphảnbiện

*( Ký, ghi rõ họ tên )*

## 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 Thơm đã tận tình giúp đỡ 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

# MỤC LỤC

## DANH MỤC HÌNH

# DANH MỤC BẢNG

# LỜI MỞ ĐẦU

Môi trường mạng Internet phát triển rộng rãi cùng với sự hỗ trợ của các phương tiện đa truyền thông đã đem lại nhiều thuận lợi và cơ hội cho con người trên mọi lĩnh vực đời sống, kinh doanh, hợp tác… Nhưng đồng thời cũng đặt ra nhiều thách thức trong việc đảm bảo tính an toàn cho thông tin được truyền giao qua các phương tiên truyền thông như: nguy cơ sử dụng trái phép, xuyên tạc bất hợp pháp thông tin lưu chuyển trên mạng.

Hơn nữa, sự phát triển mạnh mẽ của các phương tiện kỹ thuật số đã làm cho việc lưu trữ, sửa đổi và sao chép dữ liệu ngày càng đơn giản, từ đó việc bảo vệ bản quyền và chống xâm phạm trái phép các dữ liệu đa phương tiện cũng gặp nhiều khó khăn. Một công nghệ mới đã được ra đời đã giải quyết được phần nào khó khăn trên đó là giấu thông tin trong các nguồn đa phương tiện như âm thanh, hình ảnh.Mục tiêu của giấu thông tin là làm cho thông tin trở lên vô hình, từ đó khiến ta không thể thấy được đối tượng.

Trong những năm gần đây, giấu thông tin trong ảnh là một chương trình 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 trao đổi bằng ảnh là rất lớn. Nó đóng vai trò quan trọng trong hầu hết các ứng dụng bảo vệ an toàn thông tin như: xác nhận thông tin, xác định xuyên tạc thông tin, bảo vệ bản quyền của tác giả…Thông tin sẽ được giấu cùng với dữ liệu trong ảnh nhưng chất lượng ảnh ít thay đổi và không ai biết đằng sau nó mang những thông tin có nghĩa. Ngày nay khi ảnh số đã được sử dụng rất phổ biến thì giấu thông tin trong ảnh đã đem lại nhiều ứng dụng to lớn trên lĩnh vực đời sống. Trong đồán này tìm hiểu về tổng quan giấu tin trong ảnh, kỹ thuật giấu tin trong ảnh đã mã hóa. Nội dung của báo cáo được trình bày trong 3 chương:

- Chương 1: Trình bày tổng quan về giấu tin, cấu trúc ảnh bitmap và đánh giá chất lượng ảnh biến đổi bằng PSNR.
- Chương 2: Trình bày kỹ thuật giấu tin trong ảnh đã mã hóa.
- Chương 3: Cài đặt và thử nghiệm cho kỹ thuật giất tin trong ảnh đã mã hóa.

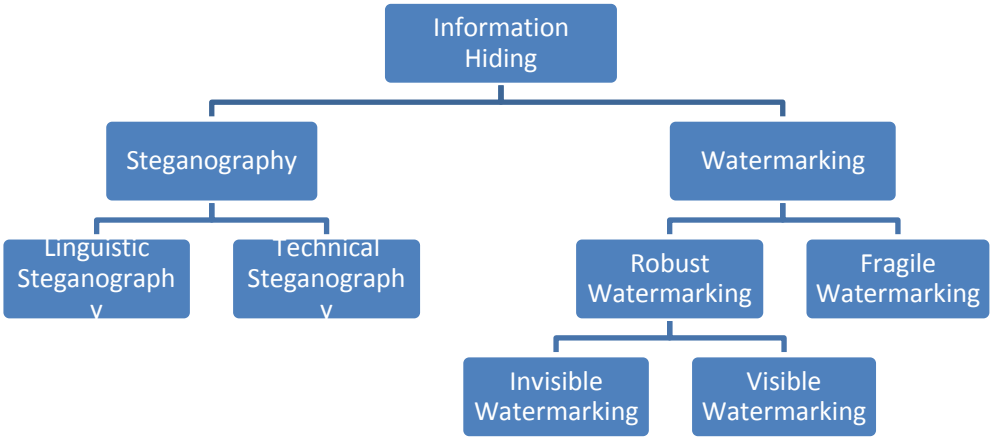
# CHƯƠNG 1. MỘT SỐ KHÁI NIỆM

## 1. 1 GIẤU THÔNG TIN

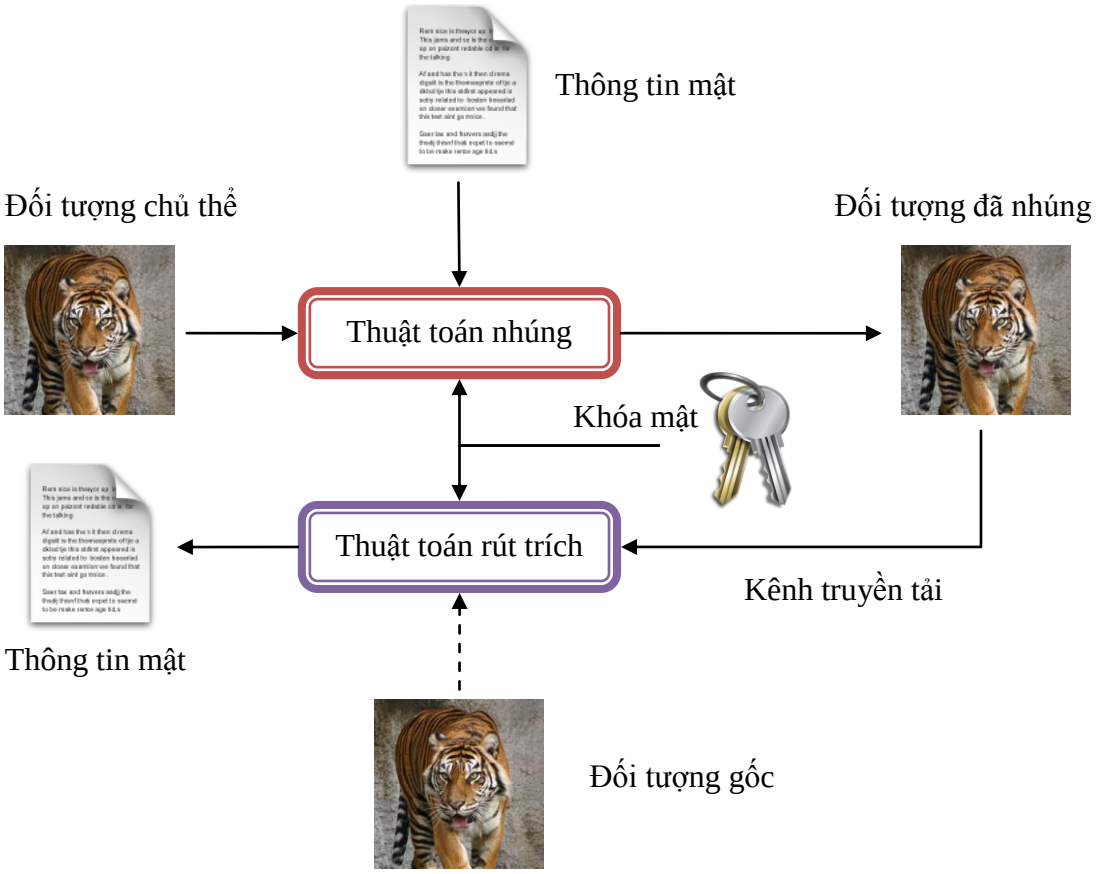
*1. 1. 1 Giới thiệu*

Cho tới nay, mã hóa thông tin và giấu thông tin là hai phương pháp được sử dụng phổ biến nhất trong vấn đề bảo mật thông tin.Trong phương pháp mã hoá thông tin, thông tin sẽ được chuyển từ dạng "có nghĩa" thành dạng "vô nghĩa", do đó chỉ những người hoặc máy tính có "chìa khóa" (key) mới giải mã được những thông tin này. Tuy nhiên, chính những thông tin "vô nghĩa" đó lại gây sự chú ý của tin tặc, dẫn đến việc thông tin có thể bị tấn công. Đối với phương pháp giấu thông tin, thông tin mật sẽ được giấu trong các dữ liệu khác (như ảnh số, tập tin phim ảnh, âm thanh, …), điều này làm cho tin tặc không nhận ra được sự tồn tại của thông tin mật. Thay vì truyền tải thông tin được mã hóa, ta sẽ truyền tải những dữ liệu "có nghĩa" có "chứa" thông tin mật bên trong, điều này tránh được sự nghi ngờ của tin tặc.

Giấu thông tin là thuật ngữ dùng chung để chỉ các phương pháp hay kĩ thuật che giấu và gắn thông tin vào các phương tiện chứa như hình ảnh, sách báo, tập tin phim ảnh hay các tập tin âm thanh…Thông tin được giấu rất đa dạng: nó có thể là một con số, một chuỗi các kí tự, một đoạn văn bản hay một ảnh số. Giấu thông tin có thể được chia thành hai hướng chính là **steganography** và **watermarking** (theo mô hình phân loại của B. Pflizmann). Mục đích của steganography là giấu thông tin quan trọng vào trong một phương tiện chứa nhằm bảo vệ thông tin mật đó, trong khi đó mục đích của watermarking là bảo vệ chính đối tượng được giấu thông tin.




**Hình 1.1** Sơ đồ phân loại giấu tin




**Hình 1.2** Mô hình giấu thông tin tổng quát

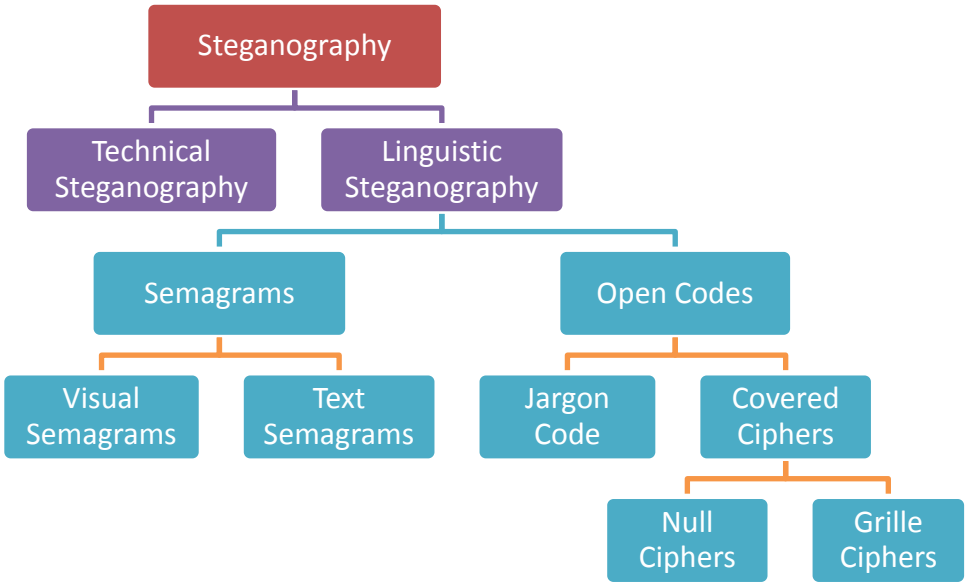
*1. 1. 2 Giấu tin mật (Steganography)*

Steganography là kĩ thuật che giấu, giấu thông tin nhằm bảo vệ thông tin đó không bị phát hiện.Thuật ngữ "steganography" có nguồn gốc từ hai chữ Hy Lạp là

"steganos" và "graphein", có nghĩa "che giấu" và "bản ghi chép". Mặc dù thuật ngữ "steaganography" mới xuất hiện vào cuối thế kỉ 15, nhưng cũng như mật mã học, những ứng dụng đầu tiên của steganography đã có từ hàng ngàn năm trước.Trong thời kì Hy Lạp cổ đại (năm 440 trước Công nguyên), thông điệp được giấu trong những bản ghi bằng sáp (bằng cách ghi trực tiếp thông tin lên gỗ rồi phủ sáp ong lên), hay được xăm lên da đầu của nô lệ. Về sau, trong thời kì chiến tranh thế giới I và II, cũng như trong các hoạt động gián điệp, khủng bố, …những kĩ thuật tiên tiến hơn được sử dụng như mực vô hình, vi ảnh, vi phim, đánh dấu kí tự…

*1. 1. 2. 1 Phân loại steaganography*

Theo B. Pflizmann, ta có các hướng phát triển của steaganography như sau:




**Hình 1.3** Phân loại Steaganography theo B. Pflizmann

Phân tích:

- Technical steganography
    - Technical steganography liên quan tới việc sử dụng các phương pháp vật lí hay hóa học để che dấu thông tin mật. Ví dụ phương pháp sử dụng mực vô hình, phương pháp vi ảnh. Các phương pháp xuất hiện từ rất lâu và hiện nay hầu như không còn được sử dụng.
- Linguistics steganography
    - Các phương pháp thuộc loại visual semagrams thường sử dụng các thực thể vật lí, nội dung thông điệp mật thường được thể hiện

quanh nghĩa của các đối tượng này. Ví dụ sử dụng vị trí của các quân cờ trên bàn cờ hoặc vẽ một con người trong các tư thế khác nhau và mỗi tư thế ứng với một ý nghĩa riêng.

- Trong khi đó, các phương pháp trong nhóm text semagrams nội dung thông tin mật được ẩn chứa thông qua cách hiển thị của văn bản. Một số kĩ thuật loại này có thể áp dụng cho cả văn bản viết tay và văn bản in. Ví dụ thay đổi khoảng cách giữa các kí tự.
- Ngoài ra các phương pháp thuộc nhóm jorgon code thường nhúng nội dung thông điệp vào trong những tín hiệu có công suất lớn, và như thế thông tin ẩn sẽ không bị phát hiện.

*1. 1. 2. 2 Ứng dụng của steganography*

Dùng trong truyền thông bí mật như trong các hoạt động phi pháp, gian lận tài chính, gián điệp công nghiệp, tình báo, …

Dùng để đính kèm thông tin bổ sung cho một đối tượng nào đó. Ví dụ như trong một album ảnh số, mỗi ảnh có thể kèm thêm thông tin về ngày, tháng, năm, nội dung ảnh, tên người chụp, …

*1. 1. 2. 3 Các yêu cầu của một thuật toángiấu thông tin*

**Tính bền vững:** Thể hiện ở khả năng ít thay đổi trước các tấn công bên ngoài như: thay đổi tính chất (thay đổi biên độ, thay đổi tần số lấy mẫu …) đối với tín hiệu âm thanh, các phép biến đổi affine (phép quay, tỉ lệ …), thay đổi chất lượng ảnh đối với tín hiệu ảnh, chuyển đổi định dạng dữ liệu (JPG – BMP, WAV – MP3, …). Hiện nay chưa có phương pháp nào có thể đảm bảo tính chất này một cách tuyệt đối.Với từng ứng dụng cụ thể, mức độ yêu cầu tính chất này sẽ khác nhau (yêu cầu cao hơn đối với watermarking).

**Khả năng không bị phát hiện:** Tính chất này thể hiện ở khả năng khó bị phát hiện, nghĩa là khó xác định một đối tượng có chứa thông tin mật hay không. Để nâng cao khả năng này, hầu hết các phương pháp ẩn dữ liệu dựa trên đặc điểm của hai hệ tri giác người là hệ thị giác (HVS – Human Visual System) và hệ thính giác (HAS – Human Auditory System). Đây là hai cơ quan chủ yếu được dùng để đánh giá chất lượng của một tín hiệu.

Khả năng không bị phát hiện phụ thuộc vào hai yếu tố sau:

Kĩ thuật giấu tin: dữ liệu được nhúng phải phù hợp với đối tượng chứa và thuật toán nhúng. Ví dụ như một thông tin mật sẽ khó bị phát hiện khi nhúng vào đối tượng A nhưng lại rất dễ thấy khi nhúng vào đối tượng B.

Kinh nghiệm của kẻ tấn công: nếu kẻ tấn công có nhiều kinh nghiệm thì khả năng phát hiện ra một đối tượng có chứa thông tin mật là không quá khó.

**Khả năng lưu trữ:** Khả năng này thể hiện ở lượng thông tin có thể nhúng trong đối tượng chủ thể. Do yêu cầu bảo mật nên khả năng lưu trữ luôn bị hạn chế. Do đó trong trường hợp muốn ẩn một thông tin có kích thước lớn, ta thường chia nhỏ thông tin ra và nhúng vào các đối tượng khác nhau.

**Tính chắc chắn:** Tính chất này khác quan trọng trong chứng nhận bản quyền, xác thực … Trong thực tế tiêu chí này được đặt nặng trong kĩ thuật gán nhãn thời gian.

**Tính bảo mật:** Có nhiều cấp độ bảo mật khác nhau, nhưng nhìn chung có 2 cấp độ chính:

- Người dùng hoàn toàn không biết có sự tồn tại của thông tin mật.
- Người dùng biết sự tồn tại của thông tin mật, nhưng phải có khóa khi truy cập.
- Bảng so sánh sau đây cho ta thấy những điểm khác biệt cơ bản giữa mật mã học và giấu thông tin.

**Bảng 1. 1** So sánh giữa mật mã học và giấu thông tin

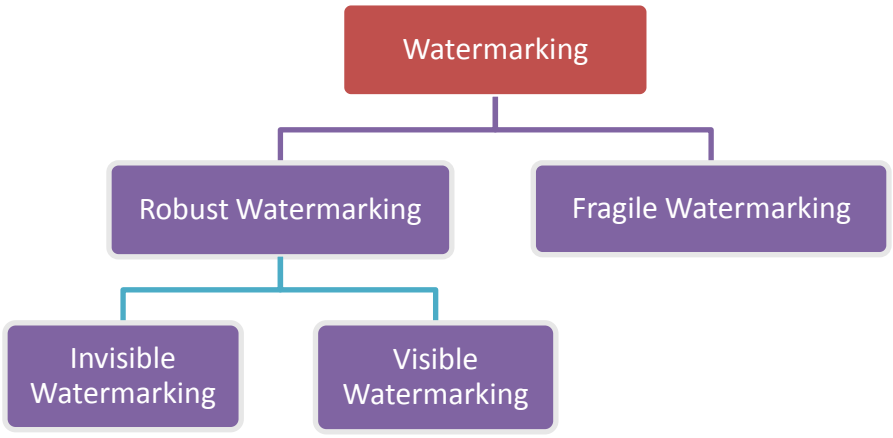
| **Mật mã học** | **Giấu thông tin** |
|---|---|
| Thông tin được mã hóa. | Thông tin được giấu đi. |
| Kĩ thuật phổ biến. | Kĩ thuật mới, vẫn còn tiếp tục phát triển. |
| Dựa vào độ phức tạp của thuật toán để bảo vệ thông tin mã hóa. | Dựa vào phương tiện chứa để che giấu sự tồn tại của thông tin mật. |

*1. 1. 3 Thủy vân số (Watermarking)*

Watermarking là kĩ thuật nhúng thông tin xác nhận hay bản quyền sở hữu vào các đối tượng như hình ảnh, tài liệu, tập tin phim ảnh và âm thanh.Với kĩ thuật watermarking số các tác giả sẽ có thể kí chữ kí trên tác phẩm của mình trước khi phân phối, chia sẻ trên Internet, mà không làm ảnh hưởng nhiều đến chất lượng tác

phẩm.Việc làm này thực sự hiệu quả, giúp bảo vệ và làm giảm thiểu sự tranh chấp về vấn đề bản quyền sở hữu trí tuệ số, vốn đang là vấn đề nóng hiện nay.

**Phân loại watermarking:** Dựa vào các tính chất khác nhau, ta có thể chia kĩ thuật watermarking thành các nhóm như sau:




***Hình 1.4****Phân loại Watermarking theo B. Pflizmann*

Thủy vân được nhúng theo phương pháp robust watermarking rất khó bị phá hủy. Thông thường robust watermarking được sử dụng trong trường hợp thông tin mật là thông tin rất quan trọng không được tiết lộ, chỉ có tác giả mới biết chính xác thông tin gì được giấu trong tác phẩm của họ. Dạng ứng dụng này thường thấy trong các trường hợp bảo vệ bản quyền sở hữu trí tuệ.

Đối với các ứng dụng được xếp vào loại thủy vân hiện rõ (Visible) thì người dùng có khả năng nhìn thấy thông tin mật.Thông thường các ứng dụng loại này sử dụng một logo làm thông tin mật nhằm mục đích chống giả mạo (Hình 1. 5).

***Hình 1.5*** *Nhúng logo vào tiền giấy*

Với các ứng dụng theo hướng Watermarking vô hình (Invisible), thì người dùng không thể biết được bất cứ thông tin nào được nhúng. Các ứng dụng loại này thường là bảo vệ quyền tác giả, bảo vệ quyền sở hữu trí tuệ.Thông tin được nhúng thường là logo hay đoạn văn bản.

Thủy vân dễ vỡ (Fragile Watermarking) thường được sử dụng trong các ứng dụng bảo vệ nội dung.Bất cứ sự thay đổi nhỏ nào cũng dẫn tới sự phá hủy hoàn toàn watermark.Ngoài mục đích xác thực nội dung thì một ứng dụng mới trong fragile watermarking là phát hiện lỗi trong quá trình truyền nhằm đánh giá hiệu quả truyền tải dữ liệu. Giống như thuật toán giấu thông tin tổng quát, một thuật toán watermarking cũng phải đảm bảo các yêu cầu.

- Tính bền vững.
- Khả năng không bị phát hiện.
- Khả năng lưu trữ.
- Khả năng vô hình.
- Tính chắc chắn.
- Tính bảo mật.

Ngoài ra còn có thêm một số yêu cầu khác do những tính chất riêng của watermarking.

**Ứng dụng của watermarking:** Bảo vệ bản quyền, bảo vệ sao chép, kiểm tra tính xác thực của dữ liệu truyền thông…

*1. 1. 4 Một số thuật ngữ cơ bản*

**Thông điệp (Message):** là thuật ngữ dùng để chỉ các thông tin được giấu trong các phương tiện chứa để chuyển đi. Thông điệp có thể có nhiều dạng như dạng văn bản hoặc hình ảnh. . .

**Phương tiện chứa gốc**: là phương tiện để chứa thông điệp mật. Đối tượng này được gọi là Cover – <datatype>.Tùy thuộc vào loại dữ liệu mà nó có các tên khác nhau.Ví dụ: cover image, cover audio, cover text, …

**Phương tiện chứa sau khi đã giấu tin**: là phương tiện sau khi nhúng thông tin mật, còn được gọi là Stego – <datatype>. Ví dụ nếu đối tượng bao tin là cover image thì đối tượng đã nhúng là stego image.

**Khóa mật:** là khóa tham gia vào quá trình nhúng, Tùy vào từng thuật toán mà ta có sử dụng khóa này hay không. Khóa này còn có tên gọi là stego key.

## 1. 2 KHÁI NIỆM VỀ ẢNH SỐ

### *1. 2. 1 Khái niệm*

Điểm ảnh (pixel) là thành phần cơ bản cấu tạo nên ảnh số (digital image).Ảnh nhị phân là ảnh đơn giản nhất trong đó mỗi điểm ảnh được cấu tạo bởi một bit. Chỉ có hai trạng thái (0, đen hoặc 1, trắng) cho mỗi điểm ảnh.Đối với ảnh xám (gray-scale image), mỗi điểm ảnh được biểu diễn nhiều hơn một bit.Một ảnh xám trong đó mỗi điểm ảnh gồm 8 bit sẽ có 256 mức xám. Mức độ xám của mỗi điểm ảnh thường có giá trị từ 0 đến 255 ( $2^8$ mức độ).Giá trị 0 ứng với màu đen và ngược lại 255 biểu diễn màu trắng (do áp dụng quy tắc ánh sáng trắng là sự kết hợp của các sóng màu tạo thành nên chọn 255 là màu trắng). Ảnh màu (được biểu diễn bằng 24 bit) là một ma trận kích thước $M \times N \times 3$, gồm ba kênh màu: đỏ, xanh lá, và xanh dương (RGB). Mỗi kênh màu của ảnh màu được xem như một ảnh xám kích thước $M \times N$ với các giá trị điểm ảnh trên kênh màu đó từ 0 đến 255.Giá trị thực của mỗi điểm trên ảnh màu được tổng hợp từ những điểm ảnh tương ứng trên từng kênh đỏ, xanh lá, và xanh dương.

### *1. 2. 2 Cấu trúc ảnh BMP*

Để thực hiện việc giấu tin trong ảnh, trước hết ta phải nghiên cứu cấu trúc của ảnh và có khả năng xử lý được ảnh tức là phải số hoá ảnh. Quá trình số hoá các dạng ảnh khác nhau và không như nhau.Có nhiều loại ảnh đã được chuẩn hoá như: JPEG, PCX, BMP… Sau đây là cấu trúc ảnh *.BMP

Mỗi file ảnh BMP gồm 3 phần:

- ✓ BitmapHeader (54 byte)
- ✓ Palette màu (bảng màu)
- ✓ BitmapData (thông tin ảnh)

Cấu trúc cụ thể của ảnh:

- Palette màu (bảng màu): bảng màu của ảnh, chỉ những ảnh lớn hơn hoặc bằng 8 bit màu mới có Palette màu.

- BitmapData (thông tin ảnh): phần này nằm ngay sau phần palette 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 từ trái sang phải. Giá trị của mỗi điểm ảnh là một chỉ số trỏ tới phần tử màu tương ứng của palette màu.

**Bảng 1.2** BitmapHeader (54 byte)

| **Byte** | **Đặt tên** | **Ý nghĩa** | **Giá trị** |
|---|---|---|---|
| 1 - 2 | ID | Nhận dạng file | 'BMP' hay 19778 |
| 3 – 6 | File_Size | Kích thước File | Kiểu Long trong turbo 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 của ảnh BMP | Tính bằng điểm ảnh |
| 23 – 26 | Height | Chiều cao của ảnh BMP | Tính bằng điểm ảnh |
| 27 – 28 | Planes | Số planes màu | Cố định là 1 |
| 29 – 30 | bitCount | Số bít cho một pixel | Có thể là 1, 4, 6, 16, 24 |
| 31-34 | Compression | Kiểu nén dữ liệu | 0: Không nén<br>1: Nén runlength8bits/ điểm ảnh<br>2: Nén runlength 4bits/điểm ảnh |

| | | | |
|---|---|---|---|
| 35 -38 | ImageSize | Kích thước ảnh | Tính bằng byte |
| 39 – 42 | XpelsPerMeter | Độ phân giải ngang | Tính bằng điểm ảnh/metr |
| 43 – 46 | YpelsPerMeter | Độ phân giải dọc | Tính bằng điểm ảnh/metr |
| 47 – 50 | ColorsUsed | Số màu sử dụng trong ảnh | |
| 51 – 54 | ColorsImportant | Số màu được sử dụng khi hiện ảnh | |

- Thành phần BitCount của cấu trúc BitmapHeader 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 đen, bit mang giá trị 1 điểm ảnh là điểm trắng.

4: Bitmap là ảnh 16 màu, mỗi điểm ảnh được biểu diễn bởi 4 bit

8: Bitmap là ảnh 256 màu, mỗi điểm ảnh biểu diễn bởi 1 byte

16: Bitmap là ảnh highcolor, 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 (RGB) 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.

## 1. 3 ĐÁNH GIÁ CHẤT LƯỢNG ẢNH

Để đánh giá chất lượng của bức ảnh ta thường sử dụng hai cách: Sai số bình phương trung bình – ***MSE (mean square error)*** và tỉ số tín hiệu trên nhiễu đỉnh – ***PSNR (pesak to signal to noise ratio)***. *MSE* giữa ảnh gốc và ảnh khôi phục được tính như sau:

$$\mathrm{MSE} = \sigma_q^2 = \frac{1}{N}\sum_{j,k}(f[j,k] - g[j,k])^2 \qquad (1.1)$$

Trong đó tổng lấy theo*j, k* tính cho tổng tất cả các điểm ảnh trong ảnh và*N* là số điểm ảnh trong ảnh.Còn *PSNR* giữa hai ảnh (*b* bít cho mỗi điểm ảnh, *RMSE* là căn bậc 2 của *MSE*) đước tính theo công thức *dB* như sau:

$$\mathrm{PSNR} = -20\log_{10}\frac{\mathrm{RMSE}}{2^{b}-1} \qquad (1.2)$$

Thông thường, nếu PSNR ≥ 40dB thì hệ thống mắt người gần như không phân biệt được giữa ảnh gốc và ảnh khôi phục.

# Chương 2. KĨ THUẬT GIẤU TIN THUẬN NGHỊCH TRONG ẢNH

## 2. 1Khái niệm giấu tin thuận nghịch

### *2. 1. 1 Khái niệm*

Dựa vào mục đích giấu tin và sau khi khôi phục thông điệp người ta có thể phân loại kĩ thuật giấu thành hai nhóm kĩ thuật giấu cơ bản sau:

- Kĩ thuật giấu thuận nghịch: là kĩ thuật giấu sau khi khôi phục thông điệp ta có thể khôi phục xấp xỉ ảnh gốc. Những kĩ thuật này được phục vụ trong các lĩnh vực như: y học, khoa học kĩ thuật, quân sự, vệ tinh…Đây là những lĩnh vực cần lưu lại ảnh gốc sau khi tách tin để đảm bảo an toàn cho thông tin đã giấu người ta có thể hủy vật mang tin.
- Kĩ thuật giấu không thuận nghịch: là kĩ thuật giấu sau khi tách tin không thể khôi phục ảnh gốc. Những kĩ thuật này phục vụ trong lĩnh vực trao đổi thông tin mật trong trường hợp cần thiết người ta có thể hủy vật mang tin mà không cần lưu trữ

### *2. 1. 2Một số kĩ thuật giấu thuận nghịch điển hình*

*2. 1. 2. 1 Phương pháp giấu NSAS*

**Giới thiệu:** Kỹ thuật histogram được Ni và các 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 [1].

**Ý tưởng**: Đề xuất của Ni với cộng sự (2004) 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 cực tiểu này.

*2. 1. 2. 2Thuật 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 [2].

**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 1 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 các cộng sự (2006).

*2. 1. 2. 3 Phươngpháp giấu tin trên miền biến đổi wavelet*

**Giới thiệu**: Thuật toán Difference được Jun Tian đề xuất vào năm 2002[6].

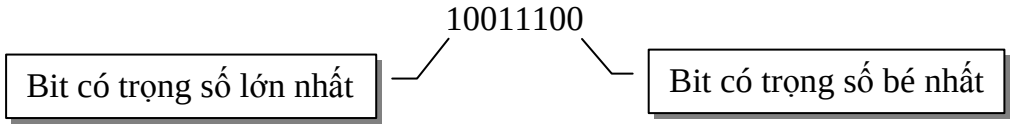
**Ý tưởng:**Tách waveletđược các dải LL, LH. Dữ liệu giấu trên dải LH. Lấy 2 điểm ảnh gần nhau tính giá trị trung bình của 2 điểm ảnh này, tính hiệu 2 điểm ảnh và lần lượt nhúng dữ liệu

## 2. 2 Một số khái niệm

*2. 2. 1 Kĩ Thuật giấu tin trên LSB*

**Bit có trọng số bé nhất (LSB- Least significant bit):** Trong một chuỗi bit B, bit có trọng số bé nhất (LSB) của B được định nghĩa là bit đầu tiên từ phải sang trong biểu diễn nhị phân của B, đồng thời cũng là bit đơn vị của B. Bit này cho ta biết tính chẵn lẻ của B và khi nó bị thay đổi sẽ không làm ảnh hưởng nhiều đến giá trị của B. Ngược lại bit có trọng số lớn nhất là bit đầu tiên tử trái sang trong biểu diễn nhị phân của B. Khi ta thay đổi bit này sẽ ảnh hưởng rất lớn đến giá trị của B.

Ví dụ: Với B = “10011100” (156 trong hệ thập phân), các bit có trọng số bé nhất và bit có trọng số lớn nhất như sau:




***Hình 2. 1*** *Minh họa phương pháp giấu dùng LSB*

Khi thay đổi bit có trọng số bé nhất từ 0 thành 1, giá trị của B là "10011101" (157 trong hệ thập phân). Khi thay đổi bit có trọng số lớn nhất từ 1 thành 0, giá trị của B là "00011100" (28 trong hệ thập phân).

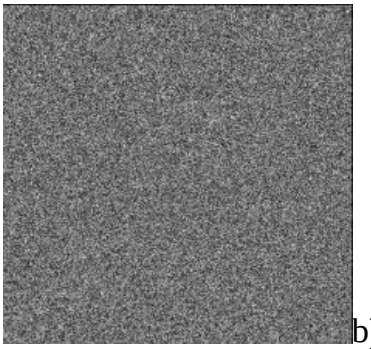
**Giấu tin dùng LSB**: là kĩ thuật giấu đơn giản nhất, ta nhúng trực tiếp các bit của thông điệp cần gửi vào các bit có trọng số bé nhất của ảnh chủ thể. Sự thay đổi các bit có trọng số bé nhất sẽ không tạo ra sự khác biệt mà mắt người có thể nhận ra, nguyên nhân là do biên độ của sự thay đổi nhỏ.

Ưu điểm của phương pháp LSB là dễ dàng thực hiện và được sử dụng trong nhiều kĩ thuật khác. Tuy nhiên, hiệu quả của phương pháp này tỉ lệ nghịch với số bit thay thế trong mỗi điểm ảnh và chỉ có thể dừng lại trong khoảng ba bit, nên lượng thông tin nhúng sẽ không được nhiều (chỉ khoảng 20% dung lượng so với ảnh chủ thể).

### *2. 2. 2 Mã hóa ảnh*

Mã hóa ảnh là phương pháp mã hóa các pixel ảnh thành dạng khó hiểu.Ví dụ minh họa hình 2.2. Trong đó hình 2. 2 a) là ảnh ban đầu trước khi mã hóa, hình 2.2 b) là ảnh sau khi mã hóa.


a)


b)

**Hình 2.2** Ảnh trước và sau khi mã hóa, a) ảnh ban đầu, b) ảnh sau khi mã hóa

**Phương pháp mã hóa ảnh phổ biến**

Trong [3] sử dụng phương pháp mã hóa bằng phép XOR (mã hóa vòng) chophép mã hóa từng điểm ảnh với một giá trị trên từng bit cụ thể như sau:

Giả sử ảnh ban đầu với kích thước $N_1$ x $N_2$ không nén định dạng và mỗi điểm ảnh có giá trị màu xám thuộc [0, 255]. Ký hiệu các thông tin của một điểm ảnh là $b_{i, j, 0}$, $b_{i, j, 1}$, …, $b_{i, j, 7}$ với $1 \le i \le N_1$, $1 \le j \le N_2$, giá trịmàu xám là $p_{i, j}$, và số điểm ảnh là N ($N=N_1$ x $N_2$). Điều đó có nghĩa:

$$b_{i,j,u} = \left[\frac{p_{i,j}}{2^u}\right] \bmod 2,\ u = 0, 1, 2 \ldots, 7 \quad (2.1)$$

Và

$$p_{i,j} = \sum_{u=0}^{7} b_{i,j,u}\, 2^u \quad (2.2)$$

Trong giai đoạn mã hóa, kết quả phép toán xor (exclusive_or) giữa bit thông tin ban đầu với bit giả định ngẫu nhiên được tính theo công thức:

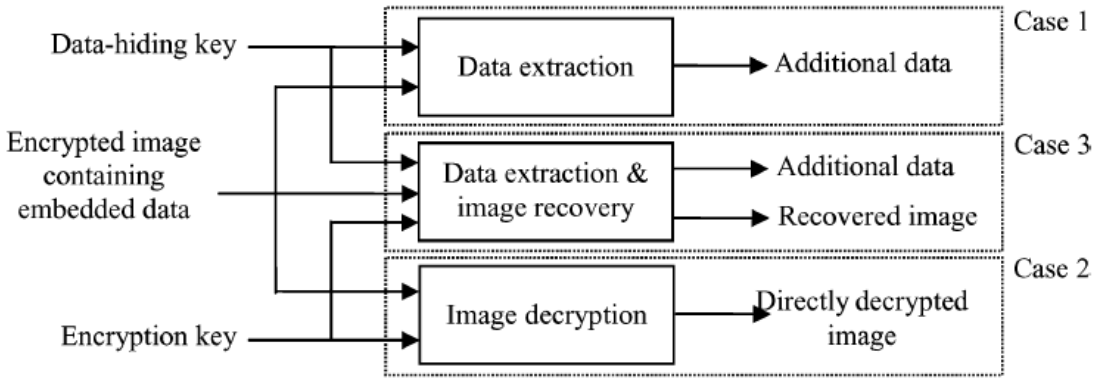
$$B_{i,j,u} = b_{i,j,u} \otimes r_{i,} \quad (2.3)$$

Với $r_{i,j,u}$ được xác định là một khoá mã hóa (sử dụng một thuật toán mật mã tiêu chuẩn dòng). Sau đó, $B_{i.j.u}$ được nối có trật tự như dữ liệu được mã hóa.

## 2. 3 Kĩ thuật giấu tin trên ảnh đã mã hóa

### *2. 3. 1 Giới thiệu*

Đây là phương pháp giấu tin trên LSB của ảnh đã mã hóa do XiNpeng Zhang đề xuất năm 2002 [3]. Phương pháp này là kĩ thuật giấu thuận nghịch.Đầu tiên, ảnh gốc (ảnh chưa nén)được mã hóa bằng một khóa nào đó.Sau đó người giấu thông tin có thể nén các bit LSB của ảnh đã mã hóa bằng một khóa mã hóa tạo ra không gian phù hợp để chèn dữ liệu. Với ảnh đã mã hóa bao gồm thông tin giấu nếu người nhận có 1 khóa giấu tin họ có thể tách ra thông tin đã giấu mà không biết nội dung thật của ảnh.Nếu người nhận có khóa mã hóa ảnh họ có thể giải mã để nhận biết nội dung của ảnh nhưng không thể tách được thông tin đã giấu. Nếu người nhận có cả khóa mã hóa và khóa giấu tin họ có thể vừa tách được thông tin đã giấu và khôi phục ảnh mà không có bất kì lỗi nào liên quan đến chất lượng ảnh khi lượng thông tin không quá lớn. Minh họa theo hình 2. 3.




**Hình 2.3** Minh họa ba trường hợp của người nhận khi có khóa tách thông tin [3]

### 2. 3. 2 Thuật toán giấu tin và tách tin

*2. 3. 2. 1Thuật toán giấu tin*

Trong giai đoạn nhúng thông tin, một số thông số được gắn vào một lượng nhỏ của các điểm ảnh đã mã hóa và LSB của điểmảnh được nén để tạo ra một không gian chứa thông tin và các dữ liệu ban đầu bởi các tham số. Các thủ tục chi tiết như sau.

Theo khóa giấu tin, người giấu chọn một cách giả ngẫu nhiên $N_p$ điểm ảnh đã mã hóa để mang các thông số cho giấu thông tin.Ở đây, $N_p$ là một số nguyên dương nhỏ, ví dụ: $N_p$= 20. Các điểm ảnh mã hóa (N – $N_p$) khác hoán vị một cách giả ngẫu nhiên và chia thành từng nhóm, mỗi nhóm trong số đó chứa L điểm ảnh.Cách hoán vị này cũng được xác định bằng khóa giấu tin.Mỗi nhóm điểm ảnh, tập hợp ít nhất M bit thông tin ít quan trọng nhất (LSB) của L điểm ảnh, và biểu thị chúng là B(k, 1), B(k, 2), …, B(k, M*L) trong đó k là một chỉ số nằm trong nhóm [1, ( N – $N_p$)/L] và M là số nguyên dương nhỏ hơn 5. Người ẩn dữ liệu cũng tạo ra một ma trận G kích thước (M*L –S) x M*L, bao gồm hai phần:

$$G = [I_{M,L-s}Q] \tag{2. 4}$$

Trong khi phần bên trái là một (M*L – S) x (M*L – S) ma trân nhận dạng, phần bên phải Q có kích thước (M*L – S) x S là một ma trận được nhị phân giả định ngẫu nhiên có nguồn gốc từ khóa giấu tin.Ở đây, S là một số nguyên dương nhỏ.Sau đó, gắn các giá trị của các tham số$N_P$ vào các điểm ảnh đã mã hóa LSB đã chọn.

Ví dụ: $N_p$ = 20, người giấu tin có lấy các giá trị của M, L là 2, 14 và 4 bit thông tin, tương ứng, và thay thế các điểm ảnh mã hóa LSB đã chọn với 20 bit thông tin. Tiếp theo, tổng (N – $N_p$) * S/L bit thông tin tạo thành từ$N_p$ các LSB ban đầu đã chọn và (N – $N_p$) * S/L – $N_p$ bit thông tin thêm vào sẽ được gắn vào các nhóm điểm ảnh. Với mỗi nhóm, tính toán

$$\begin{bmatrix} B'(k,1) \\ B'(k,2) \\ \vdots \\ B'(k,ML-S) \end{bmatrix} = G.\begin{bmatrix} B(k,1) \\ B(k,2) \\ \vdots \\ B(k,ML) \end{bmatrix} \tag{2. 5}$$

Trong đó phép toán số học là phép chia dư cho 2.Như (2. 5), B(k, 1), B (k, 2), …, B (k, M*L) được nén như (M*L – S) bit, và một không gian trống có sẵn cho việc lưu giữ thông tin. Đặt [B′ (k, M*L – S + 1), B′ (k, M*L – S + 2) ,…, B′ (k,

M*L) của mỗi nhóm là các LSB ban đâu điểm ảnh đã mã hóa đã chọn và dữ liệu cần giấu. Sau đó, thay thế [B(k, 1), B(k, 2), …, B(k, M*L)] với [B′(k, 1), B′(k, 2), …, B′(k, M*L)]mới, và đặt chúng vào vị trí ban đầu của chúng bằng một phép hoán vị nghịch đảo. Tại cùng thời điểm, (8-M) bit quan trọng nhất (MSB) của điểm ảnh đã mã hóa được giữ không thay đổi.Kể từ khi S bit được gắn vào từng nhóm điểm ảnh, tất cả$(N - N_p) \cdot S/L$ bit có thể được bố trí trong tất cả các nhóm. Rõ ràng, tỉ lệ nhúng, tỷ lệ giữa số lượng dữ liệu giấu và tổng số điểm ảnh gốc, là

$$R = \frac{((N - N_P) . \frac{S}{L} - N_P)}{N} \approx \frac{S}{L} \quad (2.6)$$

Từ đó ta có thể tóm lược lại thuật toán giấu tin như sau:

*Thuật toán giấu tin*

**Đầu vào**: cho 1 ảnh cấp xám 8 bit, chuỗi thông điệp

**Đầu ra:** ảnh đã giấu tin

Các bước thực hiện:

**Bước 1**: (Mã hóa ảnh) Giả sử ảnh ban đầu có kích thước $N_1xN_2$. Mỗi điểm ảnh với giá trị màu xám thuộc [0, 255]. Chuyển các điểm ảnh sang nhị phân và thực hiện mã hóa các điểm ảnh này với $r_i$( r là khóa mã hóa có kích cỡ bằng $N_1$ x $N_2$, giá trị sinh ngẫu nhiên) theo công thức (2. 1), (2. 2) và (2. 3) được ma trận ảnh đã mã hóa B.

**Bước 2**: Sử dụng khóa giấu tin là các tham số M (số bit LSB sẽ tách ra từ các điểm ảnh), L (kích cỡ của nhóm chia), S (số bit được giấu trong từng nhóm). Chọn giả ngẫu nhiên $N_P$ bit LSB để giấu các tham số vào ảnh. Chọn $(N-N_P)$ điểm ảnh còn lại để giấu tin.

**Bước 3**: Chia $(N-N_P)$ này thành các nhóm $G_i$ mỗi nhóm kích cỡ L pixel.

**Bước 4**: Mỗi nhóm $G_i$thực hiện giấu tin như sau: mỗi pixel của nhóm được tách ra M bit LSB được chuỗi bit ký hiệu là B(k, 1), B(k, 2), …, B(k, M*L) trong đó k là một chỉ số nằm trong nhóm [1, $(N - N_P)/L$] và M là số nguyên dương nhỏ hơn 5, sau đó trong chuỗi bit này giấu S bit thông điệp bằng cách thay thế bit thông điệp vào chuỗi bit một cách giả ngẫu nhiên.

**Bước 5**: Quá trình lặp lại cho đến khi giấu hết chuỗi bit thông điệp

**Bước 6**: thay thế lại các bit của các chuỗi vào các pixel tương ứng của từng nhóm (theo cách ta tách ra) ta được ảnh mã hóa đã giấu tin.

*2. 3. 2. 2 Thuật toán tách tin và khôi phục ảnh gốc*

Trong quá trình này, chúng ta sẽ xem xét ba trường hợp: người nhận có chỉ khóa giấu dữ liệu, chỉ có khoá mã hóa, và cả khóa giấu dữ liệu và khóa mã hóa.

Với một ảnh đã mã hóa chứa dữ liệu gắn vào, nếu người nhận chỉ có khóa giấu dữ liệu, đầu tiên ta có thể nhận các giá trị của các tham số M, L và S từ LSB các điểm ảnh mã hóa được lựa chọn.Sau đó, đổi trật tự và phân chia $(N - N_P)$ điểm ảnh khác vào các nhóm $(N - N_P)/L$ và tách ra S bit được gắn vào từ M bit LSB của từng nhóm. Khi có tất cả $(N - N_P) \cdot S/L$ bit được tách, người nhận có thể chia chúng vào thành $N_P$các điểm ảnh mã hóa LSB ban đầu đã chọnvà $(N - N_P)$ $S/L - N_P$bit thông tin đã giấu. Lưu ý rằng vì các điểm ảnh được lựa chọn giả ngẫu nhiên và hoán vị, bất kỳ kẻ tấn công mà không có khóa giấu tin không thể có được giá trị của tham số và các nhóm điểm ảnh, do đó không thể tách các dữ liệu được gắn vào. Hơn nữa, mặc dù người nhận có khóa giấu tin có thể tách thành công dữ liệu được gắn vào nhưng không thể có được bất kì thông tin nào của nội dung ảnh ban đầu.

Xem xét trường hợp người nhận chỉ có khóa mã hóa, nhưng không biết khóa giấu tin.Rõ ràng, không thể có được giá trị của tham số M, L, S và không thể tách được dữ liệu đã giấu. Tuy nhiên, nội dung hình ảnh ban đầu có thể được phục hồi.Biểu thị các đơn vị thông tin của điểm ảnh trong ảnh được để mã hóa chứa dữ liệu giấu vào là $B'_{i,j,0}$,$B'_{i,j,1}$, …, $B'_{i,j,7}$($1 \le I \le N_1$ và $1 \le j \le N_2$) người nhận có thể giải mã dữ liệu nhận được

$$b'_{i,j,u} = B'_{i.j.u} \otimes r_{i.j.u} \qquad (2.7)$$

trong đó $r_{i,j,u}$ có được từ khoá mã hóa. Các giá trị màu xám của các điểm ảnh được giải mã là:

$$p'_{i,j} = \sum_{u=0}^{7} b'_{i.j.u} \cdot 2^u \qquad (2.8)$$

Hoạt động giấu tin không làm thay đổi bất kỳ MSB của hình ảnh được mã hóa, MSB phải được giải mã giống như MSB ban đầu. Vì vậy, nội dung của hình ảnh được giải mã tương tự như hình ảnh ban đầu.Theo (2. 5), nếuB(k, M *L – S + 1) = B(k, M *L – S +2) = …=B(k, M *L) = 0, thì:

$$B'(k.v) = B(k.v), \quad v = 1, 2, \ldots, ML - S \qquad (2.9)$$

Xác suất của trường hợp này là 1/2S, và, trong trường hợp này, (M*L – S) bit gốc trong M LSB có thể được tách tin một cách chính xác.Từ S nhỏ hơn M*L, chúng ta bỏ qua các biến dạng khác của S bit được giải mã. Nếu có các bit khác 0trong B(k, M * L – S + 1 ), B(k, M *L – S +2), …, B(k, M *L), các dữ liệu được

mã hóa trong M bit LSB đã được thay đổi bởi việc giấu tin, do đó dữ liệu tách ra trong các LSB khác so với các dữ liệu ban đầu. Giả sử rằng sự phân bố ban đầu của các dữ liệu trong M bit LSB – là đồng nhất, biến dạng năng lượng cho mỗi điểm ảnh tách tin là:

$$D_E = 2^{-2M} \sum_{\propto=0}^{2^M-1} \quad \sum_{\beta=0}^{2^M-1} (\alpha - \beta)^2 \tag{2. 10}$$

Bởi vì xác suất của trường hợp này là( 2S – 1)/2S, năng lượng trung bình của sự biến dạng là:

$$A_E = \frac{2^S-1}{2^S} . 2^{-2M} . \sum_{\alpha=0}^{2^M-1} \sum_{\beta=0}^{2^M-1} (\alpha - \beta)^2 \tag{2. 11}$$

Ở đây, sự biến dạng ở các điểm ảnh $N_P$ đã chọn cũng bị bỏ qua từ khi số của nó nhỏ hơn so với kích thước N của hình ảnh.Vì vậy, giá trị của PSNR trong ảnh giảimã trực tiếp là:

$$PSNR = 10. \log_{10}(A_E) \tag{2. 12}$$

Nếu người nhận cócả khóa giấu tin và khoá giải mã, anh ta có thể là giải mã các dữ liệu được gắn vào và phục hồi hình ảnh ban đầu. Theo khóa giấu tin, các giá trị của M, L và S, LSB ban đầu của các điểm ảnh $N_P$ được mã hóa đã chọn, và (N – $N_P$)·S/L – $N_P$bit bổ sungcó thể được tách ra từ ảnh được mã hóa chứa dữ liệu gắn vào. Bằng cách đặt $N_P$ LSB vào vị trí ban đầu của nó, các dữ liệu đã mã hóa của $N_P$ điểm ảnh đã chọn được khôi phục, các giá trị màu xám ban đầu của nó có thể được giải chính xác bằng cách sử dụng khóa mã hóa. Trong phần sau đây, chúng ta sẽ khôi phục lại các giá trị màu xám ban đầu của (N – $N_P$) điểm ảnh khác. Một nhóm điểm ảnh, bởi vì B′(k, 1) , B′(k, 2), …, B′(k, M*L - S) trong (2.5) được đưa ra [B′(k, 1), B′(k, 2), …, B′(k, M *L)] phải là một trong số các vectơ có mặt

$$v = [B'(k,1)B'(k.2) \dots B'(k, ML - S)00 \dots 0]^T + a . H \tag{2. 13}$$

Trong đó a là một vector nhị phân có kích cỡ 1 x S, và H là một ma trận kích cỡ S x ML được tạo ra từ việc đảo chỗ của Q và một ma trận nhận dạng S x S

$$H = [Q^T I_S] \tag{2. 14}$$

Nói cách khác, với những hạn chế của (2.5), có thể có $2^S$ giải pháp là [B(k, 1), B(k, 2) , …, B(k, M *L)]T. Với mỗi vecto, chúng ta cố gắng để đặt các yếu tố trong đó vị trí ban đầu để có được một nhóm điểm ảnh được mã hóa và sau đó giải mã nhóm điểm ảnh bằng cách sử dụng khoá mật mã. Biểu thị nhóm điểm ảnh được giải mã là $G_k$ và các giá trị màu xám trong nó như là $t_{i, j}$, tính toán tất cả sự khác biệt giữa giải mã và ước lượng giá trị màu xám trong nhóm

$$D = \sum_{(i,j) \epsilon G_k} t_{i,j} - p_{i,j} \quad (2.15)$$

Trong đó các giá trị màu xám ước lượng được tạo ra từ những điểm lân cận của ảnh được giải mã trực tiếp, theo (2. 16), được hiển thị ở dưới cùng của trang. Rõ ràng, các giá trị màu xám ước tính trong (2. 16) chỉ phụ thuộc vào MSB của các điểm ảnh lân cận. Vì vậy, chúng ta có $2^S$ sai phân D tương ứng với $2^S$của nhóm điểm ảnh $G_k$ được giải mã.Trong số $2^S$ nhóm điểm ảnh được giải mã, một trong số đó phải là những giá trị màu xám ban đầu và sở hữu một D thấp theo mối tương quan không gian trong ảnh tự nhiên. Vì vậy, chúng ta tìm thấy D nhỏ nhất và coi vecto v tương ứng như [B(k, 1), B(k, 2), …, B(k, M *L)]T và giải mã $t_{i, j}$ như nội dung được phục hồi. Miễn là số lượng các điểm ảnh trong một nhóm là đủ lớn, và không có quá nhiều số bit gắn vào mỗi nhóm, nội dung ban đầu có thể được khôi phục hoàn toàn bằng các tiêu chí tương quan không gian. Từ $2^S$ sai phân D phải được tính trong mỗi nhóm, độ phức tạp trong tính toán của khôi phục nội dung phục hồi là O (N. $2^S$).Mặt khác, nếu nhiều điểm ảnh lân cân và một phương pháp dự đoán thông minh hơn được sử dụng để ước tính các giá trị màu xám, hiệu suất của khôi phục nội dung sẽ tốt hơn, nhưng phức tạp tính toán là cao hơn. Để giữ cho độ phức tạp tính toán thấp, chúng ta cho S nhỏ hơn 10 và chỉ sử dụng bốn điểm ảnh lân cận để tính toán giá trị ước lượng như (2. 16).

$$p_{i,j} = \frac{\left[\frac{p'_{i-1,j}}{2^M}\right] + \left[\frac{p'_{i+1,j}}{2^M}\right] + \left[\frac{p'_{i,j-1}}{2^M}\right] + \left[\frac{p'_{i,j+1}}{2^M}\right]}{4} . 2^M + 2^{M-1} \quad (2.16)$$

Từ mô tả quá trình tách tin trên chúng ta đưa ra được thuật toán tách tin cho trường hợp có cả khóa tách tin và khóa giải mã sau:

*Thuật toán tách tin*

**Đầu vào:** Ảnh đã được giấu tin

**Đầu ra:** Chuỗi thông điệp đã giấu vàảnh gốc khôi phục

**Bước 1**: Sử dụng khóa giấu tin là các tham số M (số bit LSB sẽ tách ra từ các điểm ảnh), L (kích cỡ của nhóm chia), S (số bit được giấu trong từng nhóm). Tách ra$N_P$ bit LSB đã giấu các tham số trong ảnh. Chọn (N–$N_P$) điểm ảnh còn lại để tách tin.

**Bước 2**: Chia (N-$N_P$) này thành các nhóm $G_i$ mỗi nhóm kích cỡ L pixel.

**Bước 3**: Mỗi nhóm $G_i$ thực hiện tách tin như sau: mỗi pixel của nhóm được tách ra M bit LSB được chuỗi bit ký hiệu là B(k, 1), B(k, 2), …, B(k, M*L) trong đó k là một chỉ số nằm trong nhóm [1, ( N – $N_P$)/L] và M là số nguyên dương nhỏ

hơn 5, sau đó trong chuỗi bit này tách ra S bit thông điệp đã giấu dựa theo thuật toán giấu.

**Bước 4**: Quá trình lặp lại cho đến khi tách hết số bit thông điệp đã giấu.

**Bước 5**: (Giải mã ảnh) thực hiện giải mã các điểm ảnh với $r_i$(r là khóa đã mã hóa có kích cỡ bằng $N_1$ x $N_2$, ) theo công thức (2. 1), (2. 2) và (2. 3) được ma trận ảnh đã giải mã B. Khôi phục lại ảnh ban đầu theo công thức (2. 16).

### *2.3.3 Ví dụ minh họa*

Đầu vào gồm một ảnh 6x6 và chuỗi thông điệp "TH".Ký hiệu ma trận điểm ảnh là C.

| | | | | | |
|---|---|---|---|---|---|
| 155 | 165 | 166 | 153 | 154 | 166 |
| 165 | 155 | 162 | 153 | 152 | 154 |
| 163 | 163 | 163 | 154 | 155 | 156 |
| 155 | 163 | 145 | 167 | 155 | 159 |
| 155 | 165 | 136 | 165 | 165 | 162 |
| 165 | 163 | 137 | 161 | 161 | 160 |

**Bước 1**: Mã hóa ảnh với khóa mã hóa r (r là ma trận sinh ngẫu nhiên có kích cỡ bằng 6x6) như sau:

| | | | | | |
|---|---|---|---|---|---|
| 41 | 67 | 58 | 20 | 209 | 110 |
| 203 | 167 | 233 | 113 | 222 | 233 |
| 79 | 176 | 39 | 27 | 21 | 46 |
| 135 | 191 | 211 | 246 | 102 | 67 |
| 42 | 115 | 137 | 1 | 66 | 37 |
| 154 | 21 | 255 | 198 | 204 | 34 |

Khi đó ảnh sau khi mã hóa ta được:

B = C Xor r =

| | | | | | |
|---|---|---|---|---|---|
| 178 | 230 | 156 | 141 | 75 | 200 |
| 110 | 60 | 75 | 232 | 70 | 115 |
| 236 | 19 | 132 | 129 | 142 | 178 |
| 28 | 28 | 66 | 81 | 253 | 220 |
| 177 | 214 | 1 | 164 | 231 | 135 |
| 63 | 182 | 118 | 103 | 109 | 130 |

**Bước 2**:

Chuyển chuỗi thông tin "TH" sang nhị phân ta được:0101010001001000

Chuyển ma trận ảnh đã mã hóa sang ma trận nhị phân

| | | | | | |
|---|---|---|---|---|---|
| 10110010 | 11100110 | 10011100 | 10001101 | 01001011 | 11001000 |
| 01100101 | 00111100 | 01001011 | 11101000 | 01000110 | 01110011 |
| 11101100 | 00010011 | 10000100 | 10000001 | 10001110 | 10110010 |
| 00011100 | 00011100 | 01000010 | 01010001 | 11111101 | 11011100 |
| 10110001 | 11010110 | 00000001 | 10100100 | 11100111 | 10000111 |
| 01111111 | 10110110 | 01110110 | 01100111 | 01101101 | 10000010 |

Thực hiện giấu tin với các tham số$N_P$ = 12, M=2, L=6, S= 4.

Chuyển các giá trị sang nhị phân

M =2 =$(0010)_2$

L = 6 = $(0110)_2$

S = 4 = $(0100)_2$

Ghép các chuỗi này ta được chuỗi P= 0010 0110 0100

Theo $N_P$ =12ta thực hiện chọn 12 bit LSB của 12 pixel trong ảnh đã mã hóa để giấu chuỗi nhị phân P của tham số M, L, S. Để đơn giản ta chọn 12 bit LSB của 2 dòng pixel đầu tiên của ảnh đã mã hóa.

| | | | | | |
|---|---|---|---|---|---|
| 1011001**0** | 1110011**0** | 1001110**1** | 1000110**0** | 0100101**0** | 1100100**1** |
| 0110010**1** | 0011110**0** | 0100101**0** | 1110100**1** | 0100011**0** | 0111001**0** |
| 11101100 | 00010011 | 10000100 | 10000001 | 10001110 | 10110010 |
| 00011100 | 00011100 | 01000010 | 01010001 | 11111101 | 11011100 |
| 10110001 | 11010110 | 00000001 | 10100100 | 11100111 | 10000111 |
| 01111111 | 10110110 | 01110110 | 01100111 | 01101101 | 10000010 |

Thực hiện giấu tin vào 36 - $N_P$ =24 pixel còn lại. Chia 24 pixel thành 24/L = 4 nhóm (mỗi nhóm L=6pixel) ta được 4 nhóm sau:

Nhóm 1:236 19 132 129 142 178

Nhóm 2:28 28 66 81 253 220

Nhóm 3:177 214 1 164 231 135

Nhóm 4:63 182 118 103 109 130

**Bước 3**:

Tách ra mỗi pixel của nhóm M bit LSB ta được 4 chuỗi nhị phân

Chuỗi 1: 001100011010

Chuỗi 2: 000010010100

Chuỗi 3: 011001001111

Chuỗi 4: 111010110110

**Bước 4**:

Với chuỗi 1 ta thực hiện giấu tin bằng cách thay thế4bit thông điệp0101(vì S= 4) vào 4 bit của chuỗi. Để đơn giản ta thay thế vào các vị trí lẻ của chuỗi từ bên trái sang. Ta được chuỗi mới:

00110011010

**Bước 5**:

Lặp lại bước 4 cho các chuỗi 2, 3, 4 để giấu các chuỗi nhị phân 0100, 0100, 1000.Ta được các chuỗi mới.

Chuỗi 2: 00100010100

Chuỗi 3: 01100101001111

Chuỗi 4: 11000010110

**Bước 6:**

Thay thế lại các bit của các chuỗi vào các pixel tương ứng của từng nhóm (theo cách ta tách ra) ta được:

Nhóm 1:236 19 132 129 142 178

Nhóm 2:28 2964 81 253 220

Nhóm 3:177 214 1 164 231 135

Nhóm 4:63 180 116 101 190 130

Cuối cùng ta được ảnh mã hóa đã giấu tin có nội dung như sau:

| | | | | | |
|---|---|---|---|---|---|
| 178 | 230 | 157 | 140 | 74 | 201 |
| 110 | 60 | 74 | 233 | 70 | 114 |
| 236 | 19 | 132 | 129 | 142 | 178 |
| 28 | 29 | 64 | 81 | 253 | 220 |
| 177 | 214 | 1 | 164 | 231 | 135 |
| 63 | 180 | 116 | 101 | 109 | 130 |

*Thực hiện tách tin theo thư tự ngược lại như sau:*

**Bước 1**: Ảnh mã hóa đã giấu thông tin

| | | | | | |
|---|---|---|---|---|---|
| 178 | 230 | 157 | 140 | 74 | 201 |
| 110 | 60 | 74 | 233 | 70 | 114 |
| 236 | 19 | 132 | 129 | 142 | 178 |
| 28 | 29 | 64 | 81 | 253 | 220 |
| 177 | 214 | 1 | 164 | 231 | 135 |
| 63 | 180 | 116 | 101 | 109 | 130 |

Chuyển ảnh đã giấu thông tin sang dạng nhị phân

| | | | | | |
|---|---|---|---|---|---|
| 10110010 | 11100110 | 10011101 | 10001100 | 01001010 | 11001001 |
| 01100101 | 00111100 | 01001010 | 11101001 | 01000110 | 01110010 |
| 11101100 | 00010011 | 10000100 | 10000001 | 10001110 | 10110010 |
| 00011100 | 00011110 | 01000000 | 01010000 | 11111101 | 11011100 |
| 10110001 | 11010110 | 00000001 | 10100100 | 11100111 | 10000111 |
| 01111111 | 10110100 | 01110100 | 01100101 | 01101101 | 10000010 |

Với $N_P$ = 12 có thể lấy được giá trị của L, M, S

**Bước 2**:

Thực hiện tách tin vào 36-$N_p$ =24 pixel còn lại. Chia 24 pixel thành 24/L = 4 nhóm (mỗi nhóm L=6 pixel) ta được 4 nhóm sau:

Nhóm 1: 236 19 132 129 142 178

Nhóm 2: 28 29 64 81 253 220

Nhóm 3: 177 214 1 164 231 135

Nhóm 4: 63 180 116 101 190 130

**Bước 3**:

Tách ra 2 bit LSB của mỗi pixel trong nhóm ta được 4 chuỗi bit sau:

Chuỗi 1:001100111010

Chuỗi 2: 001000010100

Chuỗi 3: 011001001111

Chuỗi 4: 110000010110

Thực hiện tách tin tại vị trí lẻ của 4 chuỗi, mỗi chuỗi S ký tự ta được thông tin cần tách M= 4 ta được các chuỗi nhị phân 0101 0100 0100 1000

**Bước 4**:

Khôi phục lại ảnh (giải mã) địa vào khóa r ta được ma trận ảnh:

| | | | | | |
|---|---|---|---|---|---|
| 155 | 165 | 166 | 153 | 154 | 166 |
| 165 | 155 | 162 | 153 | 152 | 154 |
| 163 | 163 | 163 | 154 | 155 | 156 |
| 155 | 163 | 145 | 167 | 155 | 159 |
| 155 | 165 | 136 | 165 | 165 | 162 |
| 165 | 163 | 137 | 161 | 161 | 160 |

# CHƯƠNG 3. CÀI ĐẶT VÀ THỬ NGHIỆM

## 3. 1 Môi trường thử nghiệm

-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

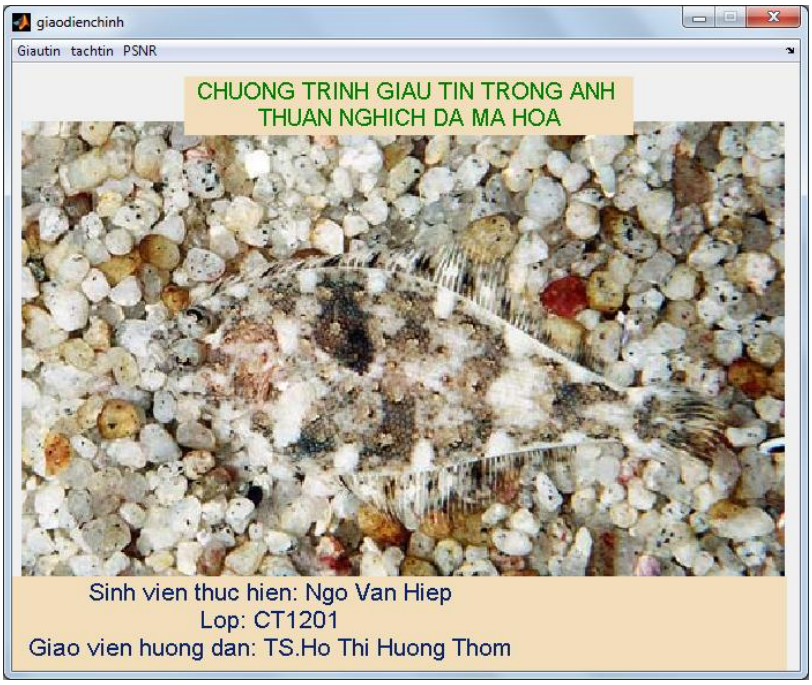
+Windows XP SP2 x64, SP3, …

+Dung lượng ổ cứng từ 1GB tới 5GB

+ Bộ nhớ RAM tối thiểu 1GB

## 3. 2 Giao diện chương trình

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




**Hình 3.1** Giao diện chính của chương trình

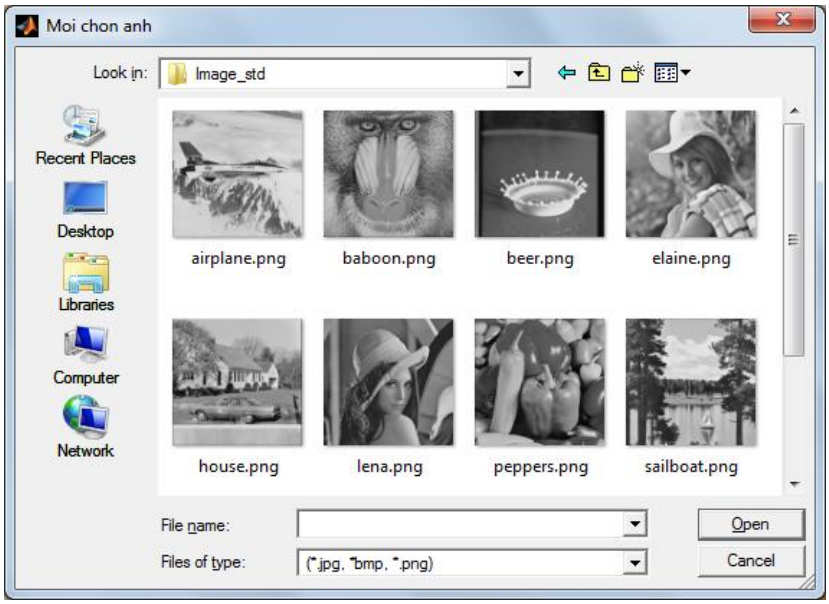
Đây là giao diện khi khởi động, từ đây ta sẽ gọi đến các giao diện khác thông qua menu.

### *3. 2. 2 Giao diện chương trình giấu tin*

Từ giao diện chính của chương trình kích vào giao diện giấu tin, cửa sổ chương trình giấu tin sẽ được hiện ra

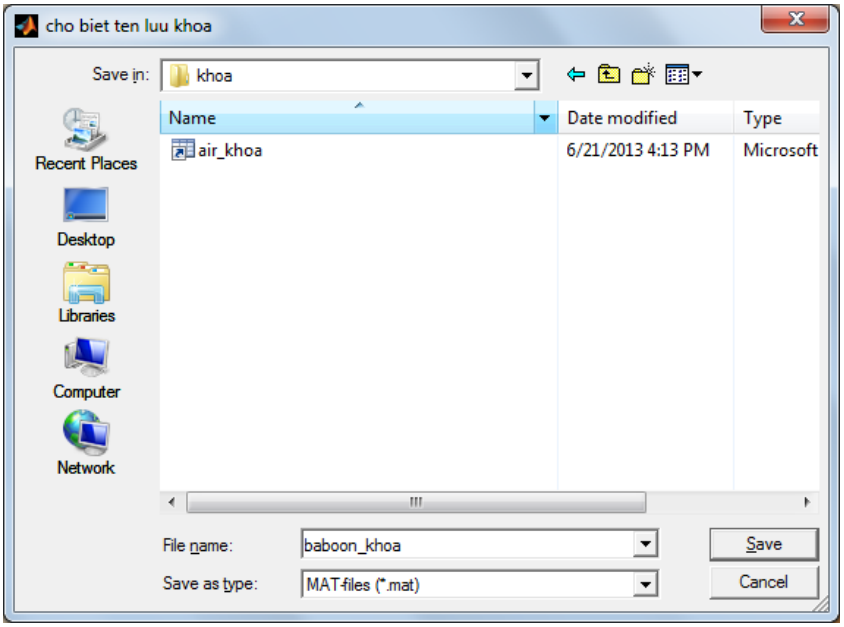

**Hình 3.2** Giao diện giấu tin

Để nhập ảnh gốc cần giấu thông tin ta kích vào nút nhập ảnh cửa sổ thư mục chứa ảnh gốc sẽ xuất hiện ta chọn ảnh bất kì để giấu thông tin.




**Hình 3.3** Thư mục chứa ảnh gốc

Sau khi nhập ảnh gốc ta kích vào nút lưu khóa, nhập tên khóa mã hóa ảnh.Khóa này sinh ngẫu nhiên bằng với kích thước của ảnh.

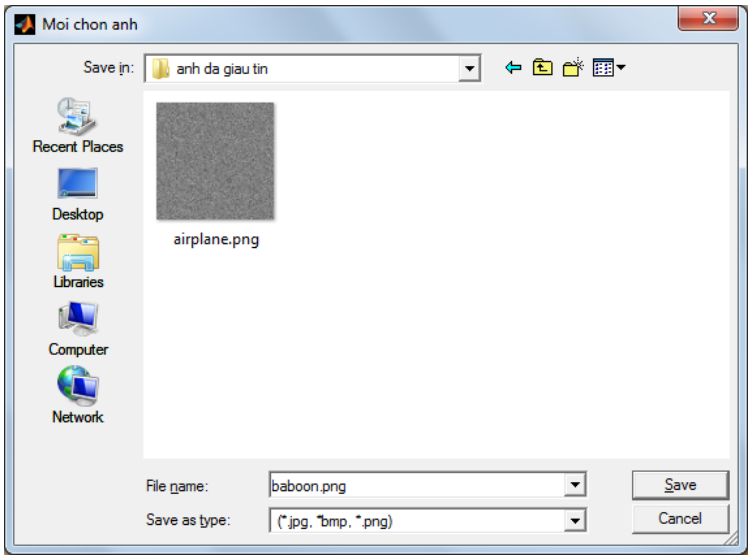



**Hình 3.4** Chọn khóa để mã hóa ảnh

Nhập khóa giấu tin M, L, S vào các ô, trong đó khóa M nằm trong các giá trị từ 1 nhỏ hơn5.


**Hình 3.5** Nhập khóa giấu tin M, L, S

Nhập tên ảnh sau khi đã mã hóa và giấu thông tin ta chọn nút lưu ảnh đã mã hóa.




**Hình 3.6** Nhập tên ảnh đã mã hóa chứa thông tin

Sau khi nhập xong khóa cần giấu tin ta bắt đầu thực hiện giấu tin kích vào nút "thực hiên giấu tin", quá trình mã hóa và giấu thông tin sẽ được bắt đầu.


**Hình 3.7** Chương trình mã hóa và giấu chuỗi thông tin vào ảnh

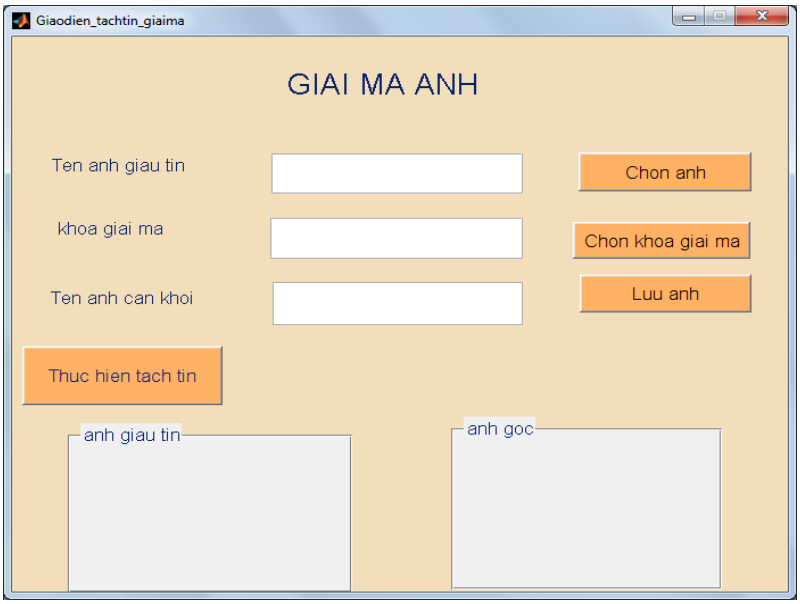
Khi quá trình mã hóa và giấu thông tin thực hiện xong sẽ xuất hiên thông báo "đã giấu tin xong" ta được ảnh mã hóa và giấu thông tin trong hộp "đã mã hóa ảnh và giấu tin".


**Hình 3.8** Chương trình sau khi đã thực hiện giấu tin

### 3. 2. 3 Giao diện tách tin

#### *3. 2. 3. 1 Giao diện giấu tin chỉ có khóa giải mã*

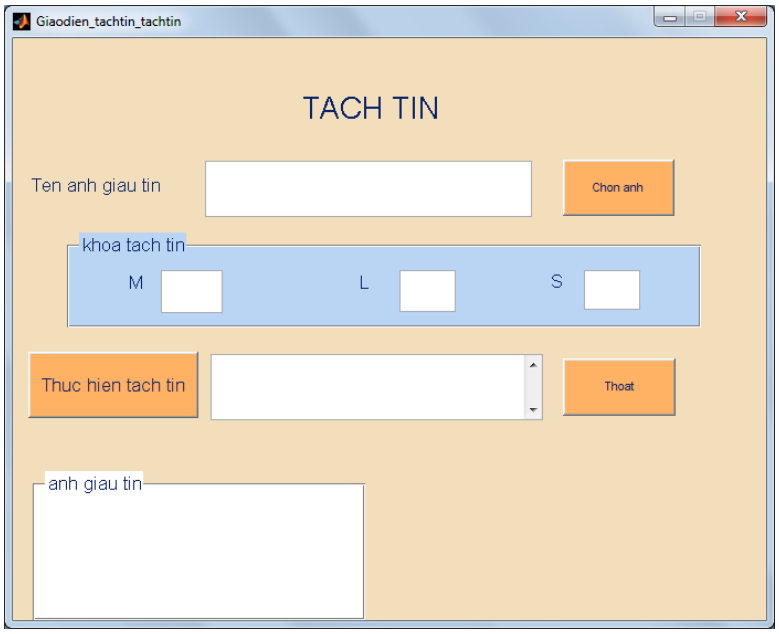
Giao diện chức năng này chỉ có thể khôi phục ảnh gốc, không tách được thông tin**.**




**Hình 3.9** Giao diện chỉ có khóa giải mã

#### *3. 2. 3. 2 Giao diện chỉ có khóa tách tin*

Giao diện này chỉ tách chuỗi thông điệp giấu trong anh mà không khôi phục được ảnh gốc.



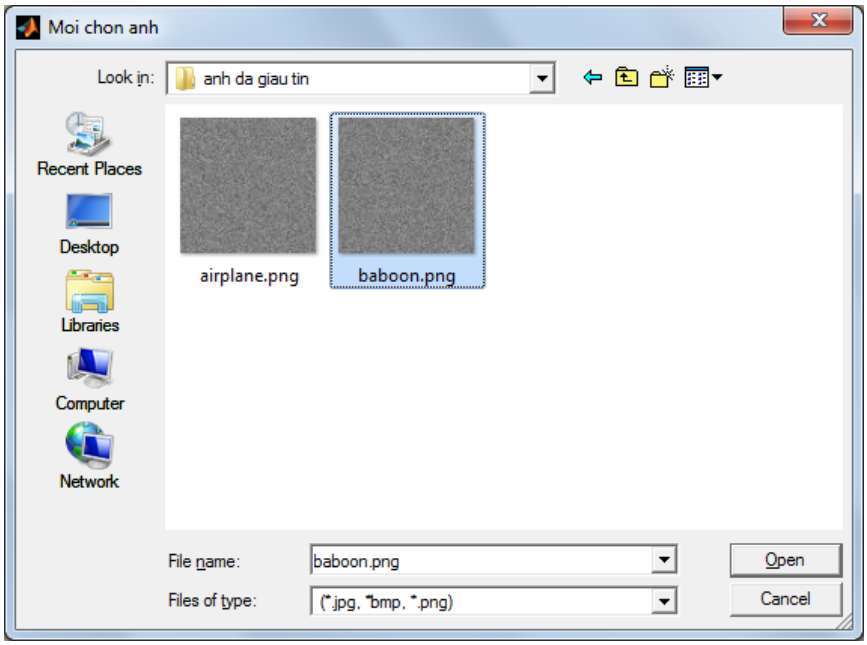

**Hình 3.10** Giao diện tách tin chỉ có khóa tách tin

### *3. 2. 3. 3 Giao diện tách tin có cả khóa mã hóa và khóa tách tin*

Sau khi thực hiện giấu tin xong, ta quay lại giao diện chính chọn vào nút tách tin. Lựa chọn menu thư mục (có đủ khóa mã hóa và khóa tách tin), giao diện tách xuất hiện.

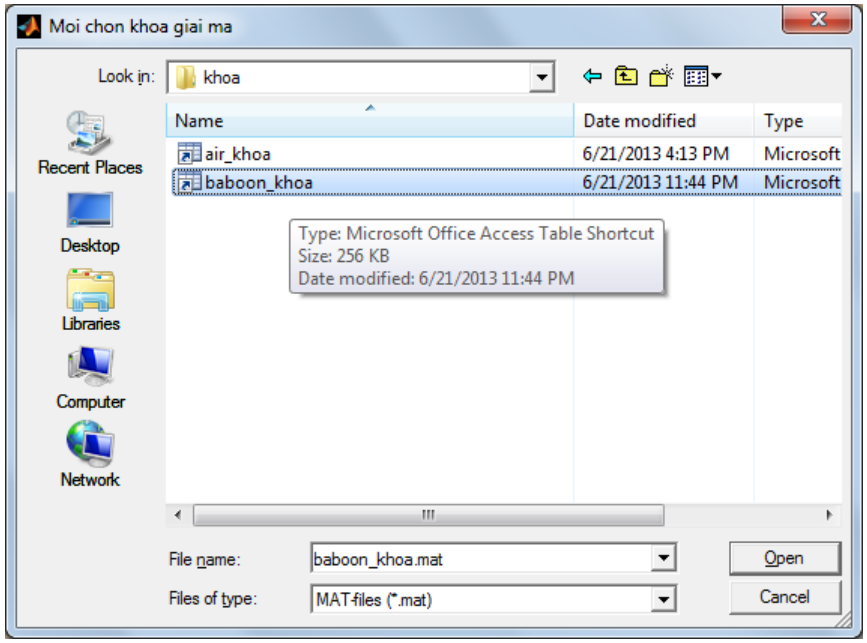

**Hình 3.11** Giao diện tách tin có khóa giải mã và khóa tách tin

Với giao diện tách tin có cả khóa giải mã và khóa tách tin ta có thể tách được chuỗi thông điệp và khôi phục lại ảnh gốc. Kích vào nút "chọn ảnh" sau đó lựa chọn ảnh cần tách tin.



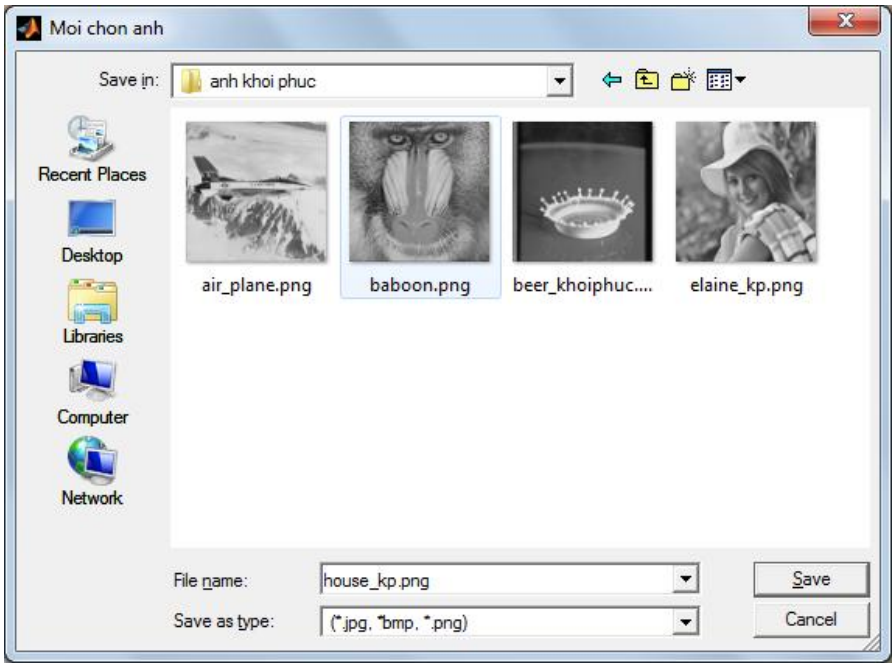

**Hình 3.12** Thư mục chưa ảnh đã giấu tin

Kích vào nút chọn khóa giải mã sẽ xuất hiện thư mục mời chọn khóa giải mã, ta chọn khóa giải mã của đúng ảnh cần tách tin.



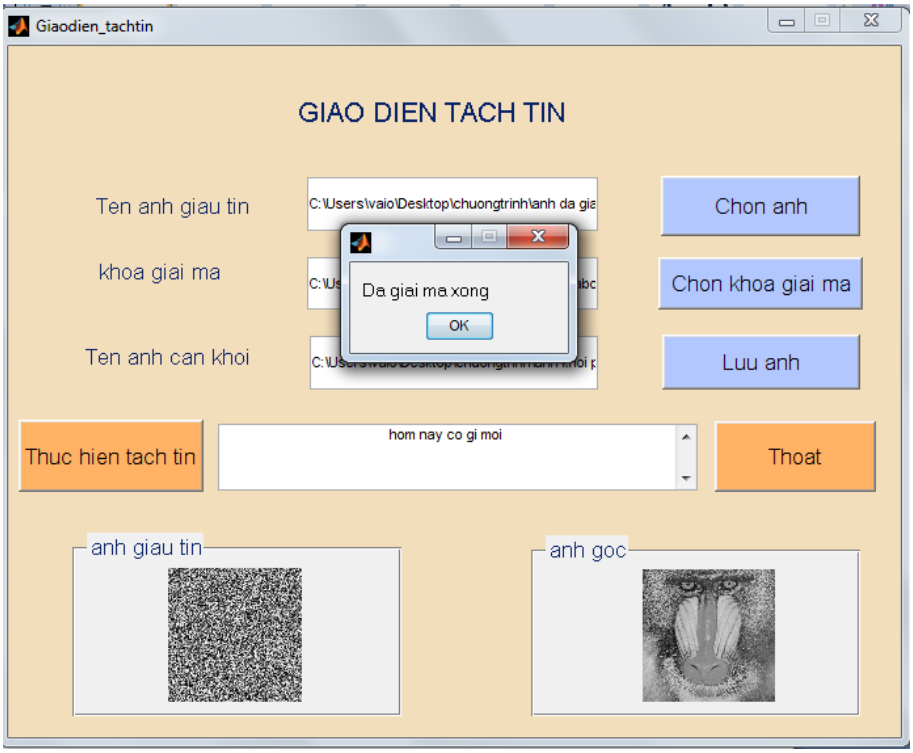

**Hình 3.13**Thư mục chứakhóa mã hóa ảnh

Sau khi chọn khóa xong ta kích vào phần "lưu ảnh" nhập tên ảnh cần được khôi phục.




**Hình 3.14** Thư mục chứa ảnh khôi phục sau khi tách tin

Sau khi đã lựa chọn xong đầu vào và đầu ra cho chương trình, chúng ta chọn nút "thực hiện tách tin".Chương trình sẽ thực hiện và đưa ra kết quả ảnh đã giấu tin ngay trên giao diện của chương trình




**Hình 3.15** Ảnh gốc xuất hiện sau khi thực hiện tách tin

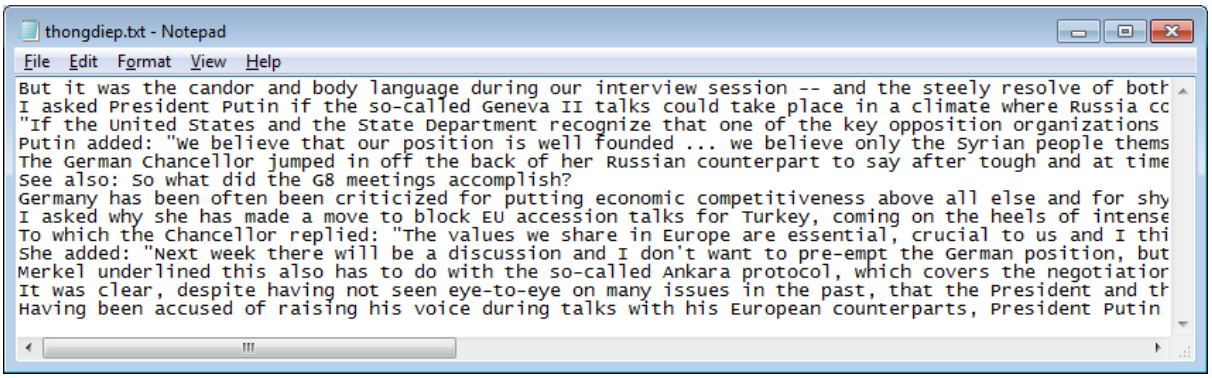
## 3. 3 Kết quả thực nghiệm và nhận xét

### *3. 3. 1 Kết quả thực nghiệm*

Thực nghiệm cho ta thấy kết quả giấu tin thuận nghịch cho ảnh đã mã hóa

**Bảng 3.1** đánh giá chất lượng trung bình PSNR với giá trị M, S khác nhau trên 3 ảnh lena.png, baboon.png, house.png (với L=10, cho cùng thông điệp có độ dài bằng 2168 bit như hình 3. 16)

| S / M | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1 | 71.2927 | 62.2795 | 62.3994 | 62.3411 |
| 2 | 56.3465 | 54.4783 | 55.3479 | 54.6745 |
| 3 | 50.2560 | 52.4036 | 53.9487 | 48.8480 |

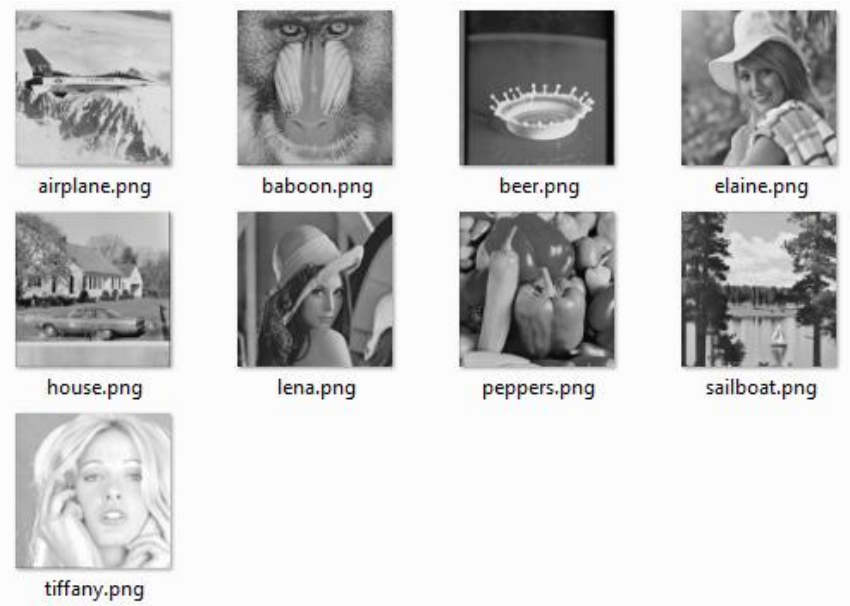



**Hình 3.16** Nội dung thông tin cần giấu vào 3 ảnh lena.png, baboon.png, house.png

**Bảng 3.2** Bảng đánh giá chất lượng PSNR giữa ảnh gốc và ảnh sau khi khôi phục trên 9 ảnh với

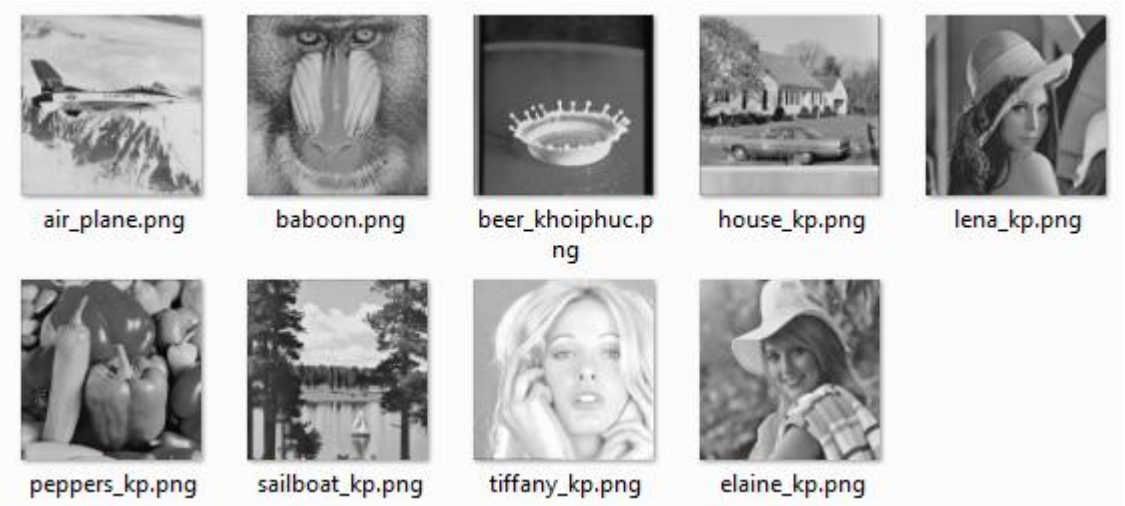
| Tên ảnh | Gía trị PSNR |
|---|---|
| Ari plane.png | 100dB |
| baboon.png | 78.3728 dB |
| Beer.png | 100dB |
| Elaine.png | 82.5023 dB |
| House.png | 79.8007 dB |
| Lena.png | 82.1915 dB |
| Peppers.png | 76.4047 dB |
| Sailboat.png | 72.9433 dB |
| Tiffny.png | 83.9911 dB |

Ảnh cấp xám 8 bit trước khi mã hóa




**Hình 3.17** Tậpảnh gốc trước khi chưa mã hóa

Ảnh sau khi tách thông tin và khôi phục




**Hình 3.18** Tậpảnh sau khi đã tách tin và khôi phục

### *3. 3. 2 Nhận xét*

Đánh giá PSNR(tỉ số tín hiệu trên nhiễu đỉnh) nếu độ nhiễu của ảnh PSNR ≥ 40dBthì hệ thống mắt người gần như không phân biệt được giữa ảnh gốc và ảnh khôi phục.

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.

Độ an toàn của kỹ thuật cao, phụ thuộc vào giá trịma trận mã hóaảnh r và khóa giấu tin LSB.

Qua thử nghiệm em nhận thấy kỹ thuật giấu tin thuận nghịch trong ảnh đã mã hóacó những ưu nhược điểm sau

Ưu điểm:

+ Khả năng bảo mật cao do khóa mãvà khóa giấu thông tin LSB do người nhận và người gửi biết với nhau. Phải có đầyđủ khóa mã hóa và khóa giải mã mới thực hiện được tách tin và khôi phụcảnh gốc.

Nhượcđiểm:

+ Quá trình giấu và tách tin chậm mất nhiều thời gian

+ Không có bước tính toán khóa giấu tin LSB và để tăng thêm độ an toàn cho dữ liệu.

# 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 việc che giấu thông tin trong ảnh số cũng như trình bày kỹ thuật giấu tin ảnhđã mã hóa.

Với kỹ thuật giấu tin trên ảnhđã mã hóa thì tính vô hình của thông tin sau khi giấu được đảm bảo, thông qua việc sử dụng một ma trận mã hóa và một khóa mã hóa LSB trong quá trình giấu và tách thông tin. Dùng phương pháp đánh giá PSNR để đánh giá chất lượng ảnh trước và sau khi khôi phục kết quả PSNR đạt được là khá cao.

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 kỹ thuật giấu tin thuận nghịch trên ảnhđã mã hóa.

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 của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

# TÀI LIỆU THAM KHẢO

[1]. Ni, Z., Shi, Y., Ansari, N., Su, W. (2003), "***Reversible data hiding***", Proc.ISCAS 2003, pp. 912–915.

[2]. J.H. Hwang, J. W. Kim, and J. U. Choi (2006), "***A Reversible WatermarkingBased on Histogram Shifting***", IWDW 2006, pp. 384-361.

[3]. XiNpeng Zhang, ***Separable Reversible Data Hiding in Encrypted Image,***IEEE transactions on information forensics and security, vol. 7, no. 2, april 2012

[4]. Nguyễn Xuân Huy, Trần Quốc Dũng, ***Giáo trình giấu tin và thuỷ vân ảnh***, Trung tâm thông tin tư liệu, TTKHTN - CN 2003

[5]. Ingemar Cox, Jeffrey Bloom, Matthew Miller, Ton Kalker, Jessica Fridrich, ***Digital Watermarking and Steganography***, Morgan Kaufmann, 2008

[6]. Jun Tian, *Reversible Watermarking by Difference Expansion*, Multimedia and Security Workshop at ACM Multimedia '02, December 6, 2002, Juan-les-Pins, France.

Đồán tốt nghiệp ngành CNTT liên quan đến kỹ thuật giấu tin:

[7].Dương Uông Hiên_lớp CT701, **"Nghiên cứu kỹ thuật giấu tin mật trên vùng biến đổi DWT"**, tiểu án tốt nghiệp ngành CNTT – 2008.

[8].Đỗ Trọng Phú – CT702, **"Nghiên cứu kỹ thuật giấu tin trên miềm biến đổi DFT"**, tiểu án tốt nghiệp ngành CNTT – 2008.

[9]. Hoàng Thị Huyền Trang – CT802 ,**"Nghiên cứu kỹ thuật phát hiện ảnh giấu tin trên miền biến đổi của ảnh"**, đồ án tốt nghiệp ngành CNTT – 2008.

[10]. Trần Đại Dương, **"Kỹ thuật giấu tin thuận nghịch trong ảnh bằng hiệu chỉnh hệ số wavelet"**, đồ án tốt nghiệp ngành CNTT.