

## LỜI CẢM ƠN

Trước hết em xin bày tỏ lòng biết ơn sâu sắc nhất tới cô giáo hướng dẫn Thạc sỹ Hồ Thị Hương Thơm – giảng viên khoa CNTT trường ĐHDL Hải Phòng là người đã tận tình giúp đỡ em rất nhiều trong suốt quá trình tìm hiểu nghiên cứu và hoàn thành đồ án tốt nghiệp này.

Em xin chân thành cảm ơn các thầy cô trong bộ môn công nghệ thông tin – trường ĐHDL 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.

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!

## MỤC LỤC

LỜI CẢM ƠN .....	1
MỤC LỤC .....	2
LỜI MỞ ĐẦU .....	4
CHƯƠNG 1: MỘT SỐ KHÁI NIỆM CƠ BẢN .....	5
1.1. KHÁI NIỆM GIẤU TIN TRONG ẢNH .....	5
1.1.1. <i>Khái niệm</i> .....	5
1.1.2. <i>Mô hình kỹ thuật giấu tin trong ảnh cơ bản</i> .....	5
1.2. TỔNG QUAN VỀ PHÁT HIỆN ẢNH CÓ GIẤU TIN .....	7
1.2.1. <i>Phân tích tin ẩn giấu (Steganalysis)</i> .....	7
1.2.2. <i>Các phương pháp phân tích</i> .....	7
1.2.2.1. <i>Phân tích trực quan</i> .....	7
1.2.2.2. <i>Phân tích định dạng ảnh</i> .....	7
1.2.2.3. <i>Phân tích thống kê</i> .....	8
1.3. KHÁI NIỆM ẢNH BITMAP .....	8
1.3.1. <i>Khái niệm</i> .....	8
1.3.2. <i>Cấu trúc ảnh BMP</i> .....	9
1.4. KỸ THUẬT GIẤU TIN TRÊN MIỀN LSB .....	8
1.4.1. <i>Khái niệm bit có trọng số thấp (LSB- Least significant bit)</i> .....	8
1.4.2. <i>Kỹ thuật giấu tin trên LSB</i> .....	9
1.5. LỘC THÔNG THẤP .....	9
CHƯƠNG 2: KỸ THUẬT PHÁT HIỆN ẢNH CÓ GIẤU TIN TRÊN MIỀN LSB BẰNG PHƯƠNG PHÁP THỐNG KÊ LLRT .....	11
2.1. GIỚI THIỆU .....	11
2.2. PHƯƠNG PHÁP PHÁT HIỆN .....	11
2.2.1. <i>Phát biểu lại bài toán theo bài toán phân lớp</i> .....	11
2.2.2. <i>Phân tích</i> .....	12

2.3. VÍ DỤ MINH HỌA .....	15
CHƯƠNG 3: CÀI ĐẶT VÀ THỬ NGHIỆM .....	23
3.1. MÔI TRƯỜNG CÀI ĐẶT.....	23
3.2. MỘT SỐ GIAO DIỆN CHƯƠNG TRÌNH .....	23
3.3. THỬ NGHIỆM .....	27
3.4 ĐÁNH GIÁ KỸ THUẬT PHÁT HIỆN .....	30
3.4.1 Độ đo đánh giá .....	30
3.4.2. Kết quả thử nghiệm đánh giá .....	31
KẾT LUẬN .....	35
TÀI LIỆU THAM KHẢO.....	36

## LỜI MỞ ĐẦU

Công nghệ thông tin và đặc biệt là sự phát triển của hệ thống mạng máy tính đã tạo nên môi trường mở và là phương tiện trao đổi, phân phối tài liệu một cách tiện lợi, nhanh chóng. Tuy nhiên cũng đặt ra một vấn đề về bảo vệ tài liệu, ngăn chặn việc đánh cắp và sao chép tài liệu một cách bất hợp pháp. Vấn đề an toàn và bảo mật thông tin hiện nay luôn nhận được sự quan tâm đặc biệt của nhiều nhà nghiên cứu trong nhiều lĩnh vực.

Giấu tin trong ảnh là một bộ phận chiếm tỷ lệ lớn nhất trong các chương trình ứng dụng, các phần mềm, hệ thống giấu tin trong dữ liệu đa phương tiện bởi lượng thông tin được trao đổi bằng ảnh là rất lớn. 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: nhận thức thông tin, xác định xuyên tạc thông tin, bảo vệ bản quyền tác giả, điều khiển truy nhập, giấu thông tin mật...

Đồ án trình bày về giấu và phát hiện ảnh có giấu thông tin. Trình bày về kỹ thuật phát hiện ảnh có giấu tin trên miền LSB bằng phương pháp thống kê LLRT.

Để nói rõ về nội dung này, đồ án của em được tổ chức gồm các chương:

Chương 1: Một số khái niệm cơ bản.

Chương 2: Giới thiệu kỹ thuật phát hiện ảnh có giấu tin trên miền LSB bằng phân tích thống kê tỉ lệ Logarit.

Chương 3: Cài đặt và thử nghiệm.

Kết luận:

Tài liệu tham khảo:

## CHƯƠNG 1: MỘT SỐ KHÁI NIỆM CƠ BẢN

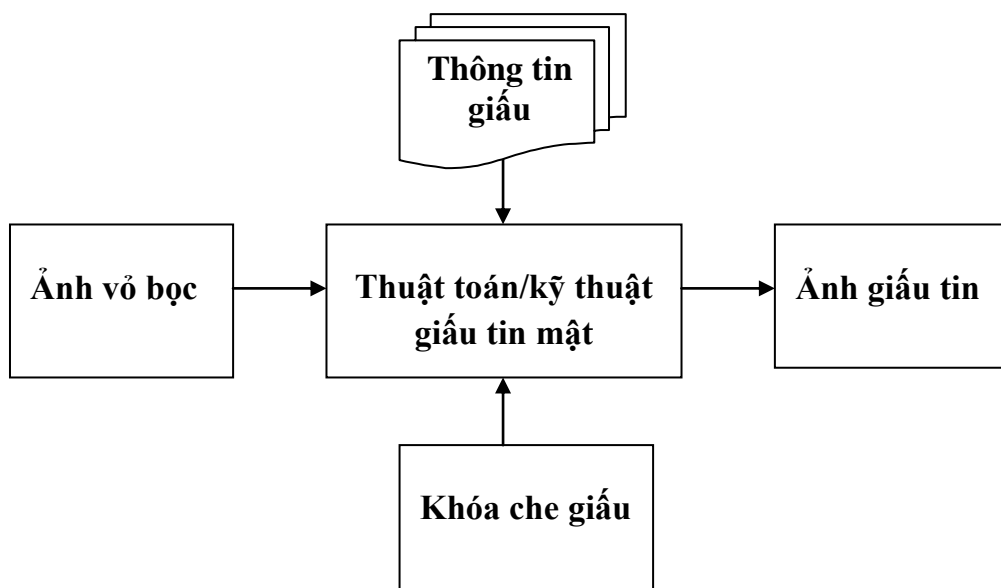
### 1.1. KHÁI NIỆM GIẤU TIN TRONG ẢNH

#### 1.1.1. Khái niệm

Giấu tin trong ảnh là một kỹ thuật giấu hoặc nhúng một lượng thông tin số nào đó vào trong một ảnh số.

#### 1.1.2. Mô hình kỹ thuật giấu tin trong ảnh cơ bản

Kỹ thuật giấu tin trong ảnh bao gồm hai quá trình đó là:  
Quá trình giấu (nhúng) tin vào ảnh.



**Hình 1.1.** Mô hình thuật toán giấu tin cơ bản

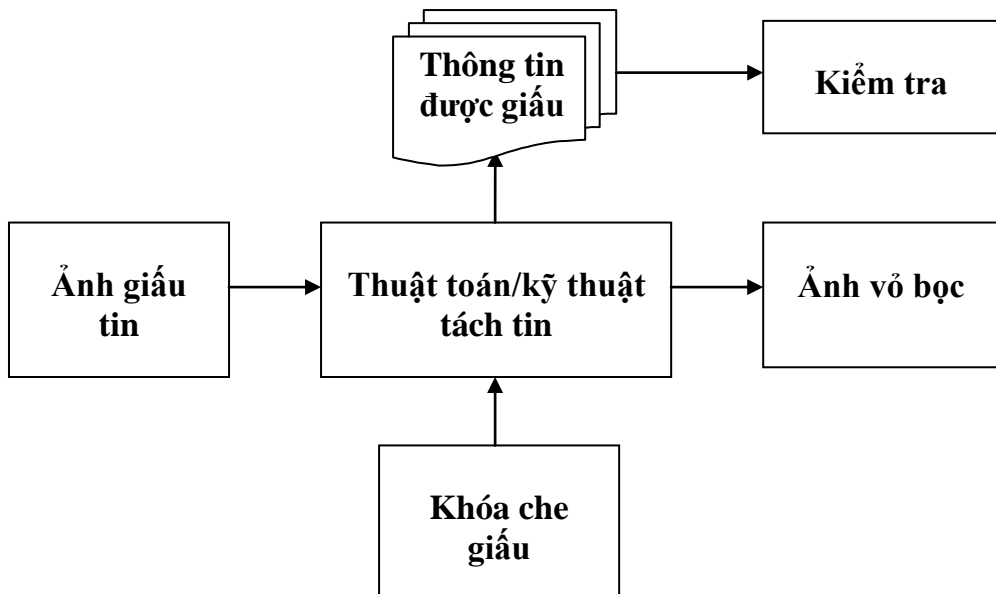
Input:

- Thông tin giấu: Tùy theo mục đích của người sử dụng mà thông tin giấu ở đây có thể là thông điệp, hình ảnh, video, âm thanh...
- Ảnh vô bọc: Là ảnh được chọn làm môi trường để giấu tin.

Output:

- Ảnh giấu đã được giấu tin

Quá trình tách tin từ ảnh giấu tin



**Hình 1.2.** Mô hình thuật toán tách tin ẩn giấu cơ bản

Input:

- Ảnh giấu tin.
- Khóa che giấu.

Output:

- Thông tin được giấu.
- Ảnh vô bọc ban đầu.

Quá trình giải mã được thực hiện thông qua thuật toán/kỹ thuật tách tin tương ứng với thuật toán/kỹ thuật nhúng tin cùng với khoá che giấu của quá trình nhúng. Kết quả thu được gồm ảnh gốc và thông tin đã giấu. Thông tin đã giấu được kiểm tra so sánh với thông tin ban đầu.

## 1.2. TỔNG QUAN VỀ PHÁT HIỆN ẢNH CÓ GIẤU TIN

### 1.2.1. Phân tích tin ẩn giấu (Steganalynis)

Steganalysis là kỹ thuật phát hiện sự tồn tại của thông tin ẩn giấu trong multimedia. Giống như thám mã, mục đích của steganalysis là phát hiện ra thông tin ẩn và phá vỡ tính bí mật của vật mang tin ẩn.

Phân tích ảnh có giấu thông tin thường dựa vào các yếu tố sau:

- Phân tích dựa vào các đối tượng đã mang tin.
- Phân tích bằng so sánh đặc trưng: So sánh vật mang tin chưa giấu tin với vật mang tin đã được giấu tin, đưa ra sự khác biệt giữa chúng.
- Phân tích dựa vào thông điệp cần giấu để dò tìm.
- Phân tích dựa vào các thuật toán giấu tin và các đối tượng giấu đã biết: Kiểu phân tích này phải quyết định các đặc trưng của đối tượng giấu tin, chỉ ra công cụ giấu tin (thuật toán) đã sử dụng.
- Phân tích dựa vào thuật toán giấu tin, đối tượng gốc và đối tượng sau khi giấu tin.

### 1.2.2. Các phương pháp phân tích

#### 1.2.2.1. Phân tích trực quan

Đây là phương pháp đơn giản nhất, phát hiện khả năng một ảnh có giấu tin hay không bằng việc phân tích ảnh một cách trực quan và tìm kiếm những điểm bất thường.

Thường dựa vào quan sát hoặc dùng biểu đồ histogram giữa ảnh gốc và ảnh chưa giấu tin để phát hiện ra sự khác biệt giữa hai ảnh căn cứ đưa ra vấn đề nghi vấn. Với phương pháp phân tích này thường khó phát hiện với ảnh có độ nhiễu cao và kích cỡ lớn.

#### 1.2.2.2. Phân tích định dạng ảnh

Phương pháp này rất rộng và thường dựa vào các dạng ảnh Bitmap để đoán nhận kỹ thuật giấu hay sử dụng, như các ảnh Bitmap thường hay sử dụng giấu trên miền LSB.

Có nhiều định dạng tệp tin ảnh khác nhau như BMP, GIF, JPEG. Mỗi loại có đặc điểm và cấu trúc định dạng tệp tin khác nhau. Do đó, khi thực hiện giấu tin,

chẳng hạn giấu tin theo LSB, sẽ cho sự thay đổi trên ảnh kết quả ở các điểm ảnh khác nhau. Và khi thực hiện phát hiện ảnh giấu tin cũng vậy.

### 1.2.2.3. Phân tích thống kê

Theo Plitzman và Westfeld, lý thuyết thống kê có thể áp dụng để phân tích thống kê các cặp giá trị (cặp giá trị điểm ảnh) để tìm sự khác biệt ở bit LSB.

Trước khi giấu tin, trên ảnh chứa thông điệp (cover image) thì mỗi cặp hai giá trị là phân phối không đều. Sau khi giấu tin, giá trị trong mỗi cặp có xu hướng trở nên bằng nhau. Hơn nữa, nếu các kỹ thuật giấu tin mật giấu các bit thông điệp một cách tuần tự vào các điểm ảnh liên tiếp nhau, bắt đầu từ góc trên trái thì ta sẽ quan sát được sự thay đổi đột ngột trong các thống kê.

## 1.3. KHÁI NIỆM ẢNH BITMAP

### 1.3.1. Khái niệm

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”, nét vẽ được thể hiện là các điểm ảnh. Qui ước màu đen, trắng tương ứng với các giá trị 0, 1.

### 1.3.2. Cấu trúc ảnh BMP

Cấu trúc một tệp ảnh BMP gồm có bốn phần:

- Bitmap File Header: Lưu trữ thông tin tổng hợp về tệp ảnh BMP.
- Bitmap Information: Lưu trữ thông tin chi tiết về ảnh bitmap.
- Color Palette: Lưu trữ định nghĩa của màu được sử dụng cho bitmap.
- Bitmap Data: Lưu trữ từng điểm ảnh của hình ảnh thực tế.

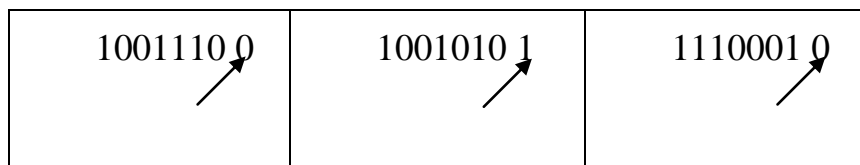
## 1.4. KỸ THUẬT GIẤU TIN TRÊN MIỀN LSB

### 1.4.1. Khái niệm bit có trọng số thấp (LSB- Least significant bit).

Bit có trọng số thấp là bit có ảnh hưởng ít nhất tới việc quyết định tới màu sắc của mỗi điểm ảnh, vì vậy khi ta thay đổi bit ít quan trọng của một điểm ảnh thì màu sắc của mỗi điểm ảnh mới sẽ tương đối gần với điểm ảnh cũ. Như vậy kỹ thuật tách bit trong xử lý ảnh được sử dụng rất nhiều trong quy trình giấu tin. Việc xác định LSB của mỗi điểm ảnh trong một bức ảnh phụ thuộc vào định dạng của ảnh và số bit màu dành cho mỗi điểm của ảnh đó.



Ví dụ: Tách bit cuối cùng trong 8 bit biểu diễn mỗi điểm ảnh của ảnh 256 màu



**Hình 1.3:** Mỗi điểm ảnh biểu diễn bởi 8 bit, bit cuối cùng được coi là bit ít quan trọng nhất tức là bit bên phải nhất

#### 1.4.2. Kỹ thuật giấu tin trên LSB

Các kỹ thuật giấu tin trên miền LSB thuộc vào nhóm giấu tin trong miền quan sát. Phương pháp này thường nhúng thông tin vào các bit có trọng số thấp của ảnh hay được áp dụng trên các ảnh bitmap không nén, các ảnh dùng bảng màu. Ý tưởng chính của phương pháp này là lấy từng bit của tin mật rải nó lên ảnh vỏ bọc, thay đổi bit có trọng số thấp của ảnh bằng các bit của tin mật. Vì khi thay đổi các bit có trọng số thấp không ảnh hưởng đến chất lượng ảnh, và mắt người không cảm nhận được sự thay đổi của ảnh đã giấu tin.

### 1.5. LỌC THÔNG THẤP

Lọc thông thấp thường được sử dụng để làm trơn nhiễu. Bộ lọc trên là bộ lọc tuyến tính theo nghĩa là điểm ảnh ở tâm cửa sổ sẽ được thay bởi tổ hợp các điểm lân cận chập với mặt nạ. Toán tử trung bình không gian là lọc thông thấp. Mỗi điểm ảnh được thay thế bằng trung bình trọng số của các điểm lân cận và được định nghĩa như sau:

$$v(m, n) = \sum_{(k, l) \in W} a(k, l) y(m - k, n - l)$$

Khi dùng các trọng số như nhau, phương trình trên sẽ trở thành:

$$v(m, n) = \frac{1}{N} \sum_{(k, l) \in W} y(m - k, n - l)$$

với :  $y(m, n)$ : ảnh đầu vào,

$v(m, n)$ : ảnh đầu ra,

$a(k, l)$  : là cửa sổ lọc.

với  $a_{k,l} = \frac{1}{N_w}$  và  $N_w$  là số điểm ảnh trong cửa sổ lọc  $W$ .

Lọc trung bình có trọng số chính là thực hiện chập ảnh đầu vào với nhân chập  $H$ . Nhân chập  $H$  trong trường hợp này có dạng:

$$H_{t1} = \frac{1}{8} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$H_b = \frac{1}{(b+2)^2} \begin{bmatrix} 1 & b & 1 \\ b & b^2 & b \\ 1 & b & 1 \end{bmatrix}$$

Giả sử đầu vào biểu diễn bởi ma trận  $X[m,n]$ .

Ảnh số thu được bởi lọc thông thấp  $Y=H \otimes X$ .

Để dàng nhận thấy khi  $b=1$ ,  $H_b$  chính là nhân chập  $H_{t1}$  (lọc trung bình). Để hiểu rõ hơn bản chất khử nhiễu cộng của các bộ lọc này, viết lại phương trình thu nhận ảnh dưới dạng:

$$X_{qs}[m,n] = X_{góc}[m,n] + \eta[m,n]$$

Trong đó  $\eta[m, n]$  là nhiễu cộng có phương sai  $\sigma_n^2$ . Như vậy, có:

$$Y[m, n] = \frac{1}{N_w} \sum_{k,l \in W} X_{qs}(m-k, n-l) + \eta[m, n]$$

$$\text{hay: } Y[m, n] = \frac{1}{N_w} \sum_{k,l \in W} X_{qs}(m-k, n-l) + \frac{\sigma_n^2}{N_w}$$

Như vậy, nhiễu cộng trong ảnh đã giảm đi  $N_w$  lần.

## CHƯƠNG 2: KỸ THUẬT PHÁT HIỆN ẢNH CÓ GIẤU TIN TRÊN MIỀN LSB BẰNG PHƯƠNG PHÁP THỐNG KÊ LLRT

### 2.1. GIỚI THIỆU

- Phương pháp phát hiện giấu tin trên miền LSB – logarithm likelihood ratio test (LLRT) được nghiên cứu bởi nhóm các nhà khoa học: K. Sullivan, O. Dabeer, U. Madhow, B.S. Manjunath, and S. Chandrasekaran. Tại trường đại học Santa Barbara-California, USA.

- Ý tưởng: Thuật toán phát hiện ảnh có ẩn giấu tin dựa trên lý thuyết Kullback- Leibler(  $D_{(p||q)}$  ).

Giả sử có ảnh A là một tín hiệu số được hiển thị dưới dạng ma trận hoặc biểu đồ, để xác định A có giấu tin ẩn hay không, ta thực hiện như sau:

+ Trường hợp thứ nhất: có ảnh B là ảnh gốc của ảnh A. Khi đó ta đem ảnh B giấu tin với tỉ lệ là  $R_0$  được ảnh C. Ta tính được độ lệch Kullback-Leibler giữa ảnh A với ảnh C ( $D_{(A||C)}$ ) và độ lệch Kullback-Leibler giữa ảnh B và ảnh A ( $D_{(B||A)}$ ). Với một ngưỡng  $T(\alpha)$  đã xác định trước, ta xác định được ảnh A có giấu tin khi:

$$D_{(A||C)} - D_{(B||A)} \leq T(\alpha) ,$$

+ Trường hợp thứ hai: không có ảnh gốc của ảnh A. Khi đó ta sẽ phải ước lượng trung bình để xây dựng ảnh gốc B từ ảnh A đã cho. Sau đó tiếp tục đem ảnh B vừa xây dựng được giấu tin với tỉ lệ  $R_0$ . Thực hiện như trường hợp trên.

### 2.2. PHƯƠNG PHÁP PHÁT HIỆN

#### 2.2.1. Phát biểu lại bài toán theo bài toán phân lớp

Kỹ thuật phát hiện LLRT là phương pháp phát hiện ảnh có giấu tin trên LSB dựa vào bài toán phân loại ảnh có giấu tin bằng kiểm định giả thuyết giữa hai giả thuyết:

$H_0$  (ảnh gốc – không giấu tin) và

$H_1$  (ảnh có giấu tin).

Với bài toán phân loại này chúng ta phải đi tìm ngưỡng hợp lý để có thể phân loại tốt, trong thống kê LLRT (kiểm định dựa trên tỉ lệ hợp lý loga) được cho

là bài toán tối ưu với sai số loại I ( bác bỏ  $H_0$  khi thực tế  $H_0$  đúng ) cho trước, cực tiểu hóa sai số loại II ( chấp nhận  $H_0$  khi thực tế  $H_0$  sai).

### 2.2.2. Phân tích

#### \* Mô hình thống kê cho kỹ thuật giấu trên LSB

Giả sử thông tin của mỗi ảnh được lưu trữ trên mảng một chiều. Ảnh gốc được biểu diễn là  $\{x_i\}_{k=1}^N$ , giá trị  $x_i$  được biểu diễn trên 8 bit,  $x_i \in \{0,1,\dots,255\}$ . Tiếp theo giấu thông tin trên LSB với tỉ lệ giấu R bit (R : Là tỉ lệ giữa độ dài thông điệp trên kích cỡ của ảnh). Chuỗi bit thông điệp kí hiệu là  $\{d_k\}_{k=1}^N$  (giả sử chuỗi được phân bố độc lập ngẫu nhiên - independent and identically distributed (i.i.d))

Khi đó:

$$\text{Xác suất } d_k \text{ có giá trị là 0: } P(d_k=0) = \frac{R}{2},$$

$$\text{Xác suất } d_k \text{ có giá trị là 1: } P(d_k=1) = \frac{R}{2},$$

$$\text{Xác suất } d_k \text{ rỗng là: } P(d_k=NULL) = (1-R), 0 < R \leq 1.$$

Nếu  $d_k = NULL$  thì không có dữ liệu ẩn trong  $x_i$ , nếu không thì  $d_k$  sẽ thay thế LSB của  $x_i$ . Với tỉ lệ giấu tin trên LSB là R, nếu gọi hàm khối xác suất (probability mass function - PMF) của  $x_i$  là  $p(n)$ ,  $n = 0,1, \dots, 255$ , thì hàm khối xác suất của các giá trị chẵn và giá trị lẻ của ảnh sau khi giấu tin trên LSB với tỉ lệ R là:

$$p_R(2l) = 1 - \frac{R}{2} p(2l) + \frac{R}{2} p(2l+1),$$

$$p_R(2l+1) = \frac{R}{2} p(2l) + 1 - \frac{R}{2} p(2l+1).$$

Với  $l = 0, 1, \dots, 127$ . Để thuận tiện, biểu diễn PMF thành 256 chiều theo vectơ  $p$ ,  $p_R$ , được phương trình tuyến tính sau:  $p_R = Q_R p$ , với  $Q_R$  là ma trận  $256 \times 256$ . Trong đó:

$$Q_R = \begin{bmatrix} 1 - \frac{R}{2} & \frac{R}{2} \\ \frac{R}{2} & 1 - \frac{R}{2} \end{bmatrix},$$

$$p = [p(2l) \quad p(2l+1)],$$

$$p_R = [p_R(2l) \quad p_R(2l+1)].$$

**\*Phát hiện ảnh có giấu tin khi có ảnh gốc**

Có A là ảnh cấp xám cần kiểm tra có giấu tin ẩn hay không.

Giả sử ảnh A được biểu diễn bằng một ma trận một chiều với  $\{x_k\}_{k=1}^N$  là giá trị mỗi điểm ảnh của A.  $x_k = \{0, 1, \dots, 255\}$ .

$q$  là xác suất xuất hiện mỗi giá trị  $x_k$  trong ảnh A.

B là ảnh gốc của ảnh A.

Ảnh B được biểu diễn bằng một ma trận một chiều với  $\{y_k\}_{k=1}^N$  là giá trị mỗi điểm ảnh của B.  $y_k = \{0, 1, \dots, 255\}$ .

$p$  là xác suất xuất hiện mỗi giá trị  $y_k$  trong ảnh B.

Từ ảnh B đem giấu tin với tỉ lệ  $R_o$  ta được ảnh C.

Ảnh C được biểu diễn bằng một ma trận một chiều với  $\{v_k\}_{k=1}^N$  là giá trị mỗi điểm ảnh của C.  $v_k = \{0, 1, \dots, 255\}$ .

$u$  là xác suất xuất hiện mỗi giá trị  $v_k$  trong ảnh C.

Phương pháp log likelihood ratio test (LLRT) sử dụng theo công thức Kullback – Leibler, ta có:

$$D(p||q) = \sum_{k=0}^{255} p_k \log \frac{p_k}{q_k} .$$

Khi đó xác định ảnh có giấu tin khi:

$$D(q||u) - D(p||q) \leq T(\alpha).$$

$T(\alpha)$  là ngưỡng để phân loại theo [3], nhóm tác giả chọn  $T(\alpha) = 0$  và  $R = 0,05$  sau khi kiểm tra thực nghiệm với hơn 4000 bức ảnh.

### **\*Phát hiện ảnh có giấu tin khi không có ảnh gốc**

A là ảnh cần kiểm tra, chúng ta không biết ảnh gốc B của ảnh A. Lúc này ảnh B sẽ được xây dựng bằng cách ước lượng từ ảnh A bằng phương pháp lọc thông thấp (lowpass - filter)(1.5). Như đã trình bày ở trên, lọc thông thấp là một thuật toán lọc nhiễu tín hiệu của ảnh. Nếu là ảnh gốc thì tín hiệu của các điểm ảnh là rất mịn, còn tín hiệu các điểm ảnh của ảnh có giấu tin bị nhiễu do tác động của việc giấu tin. Do vậy ta sẽ lọc nhiễu tín hiệu ảnh A để ước lượng được ảnh gốc B.

Sau khi xây dựng được ảnh gốc B, tiến hành làm theo những bước như trường hợp 1 để kiểm tra ảnh A có giấu tin hay không.

#### **❖ Nhận xét:**

Do trên thực tế ta không biết trước ảnh gốc nên thuật toán phát hiện ảnh giấu tin được xây dựng theo trường hợp 2.

#### **→ Thuật toán LLRT**

##### ***Đầu vào:***

- I: là một ảnh hoặc một tập ảnh cấp xám

##### ***Đầu ra:***

- Kết luận ảnh có giấu tin hay không, hoặc trong tập ảnh có ảnh nào giấu tin, ảnh nào không giấu tin.

##### ***Các bước thực hiện***

Bước 1: Đọc ảnh I, đọc giá trị điểm ảnh vào một ma trận  $A_{M \times N}$ .

Bước 2: Tính tần suất ( $h_A$ ) của ma trận A theo giá trị từ 0 đến 255.

Bước 3: Tính xác suất xuất hiện của mỗi giá trị ảnh trong ma trận A:

$$q = h_A / (M \times N)$$

Bước 4: Sử dụng ảnh I, ước lượng ảnh gốc G (bằng phương pháp lọc thông thấp). Đọc giá trị các điểm ảnh của ảnh G vào ma trận B.

Bước 5: Tính tần suất ( $h_G$ ) của ma trận B theo giá trị từ 0 đến 255.

Bước 6: Tính xác suất xuất hiện của mỗi giá trị ảnh trong ma trận B:

$$p = h_B / (M \times N)$$

Bước 7: Tính hàm  $D(p||q)$

$$D(p||q) = \sum_{k=0}^{255} p_k \log \frac{p_k}{q_k} .$$

Bước 8: : Giấu tin vào ảnh G với tỉ lệ  $R_0 = 0,05$ , được ảnh T. Đọc giá trị các điểm ảnh của ảnh T vào ma trận C.

Bước 9: Tính tần suất ( $h_T$ ) của ma trận C theo giá trị từ 0 đến 255.

Bước 10: Tính xác suất xuất hiện của mỗi giá trị ảnh trong ma trận C:

$$u = h_C|(M \times N)$$

Bước 11: Tính hàm  $D(q||u)$ .

$$D(q||u) = \sum_{k=0}^{255} q_k \log \frac{q_k}{u_k} .$$

Bước 12: Tính  $kq = D(q||u) - D(p||q)$ .

So sánh  $kq$  với  $T(\alpha) = 0$  :

Nếu  $kq \leq T(\alpha) \rightarrow$  kết luận ảnh I có giấu tin.

Nếu  $kq > T(\alpha) \rightarrow$  kết luận ảnh I không giấu tin.

### 2.3. VÍ DỤ MINH HỌA

Cho ảnh đầu vào :



**Hình 2.1:** tocdep.png

Đọc giá trị điểm ảnh vào ma trận  $A_{M \times N}$ :

Columns 460 through 468								
8	8	8	7	7	7	7	7	6
7	8	8	7	7	7	7	7	6
8	8	8	8	8	7	6	6	5
7	8	8	7	8	7	7	6	6
8	8	8	7	7	8	7	6	5
8	8	8	7	7	8	6	7	5
8	8	8	7	8	8	7	6	5
8	8	8	7	8	7	7	6	6
8	7	8	7	8	7	7	7	5
8	7	8	8	8	7	6	6	6
9	7	8	8	8	8	7	7	7
8	8	8	8	8	7	7	6	7
8	8	8	8	8	8	7	7	7
9	8	8	8	8	8	7	7	7

**Hình 2.2:** Ma trận A

Chuyển ma trận A thành ma trận 1 chiều.

11
10
10
10
10
10
11
10
10
10
10
11
10
11
10
11
10
10
10

**Hình 2.3:** Ma trận A(:)



Tính tần suất cho ma trận A theo giá trị từ 0 đến 255:

Columns 241 through 244	11	3	4	3
Columns 245 through 248	0	0	4	0
Columns 249 through 252	0	1	0	0
Columns 253 through 256	0	0	1	0

**Hình 2.4:** hist(A,0:255)

Tính xác suất xuất hiện của mỗi giá trị ảnh trong ma trận A:

Columns 241 through 245	0.0001	0.0000	0.0000	0.0000	0
Columns 246 through 250	0	0.0000	0	0	0.0000
Columns 251 through 255	0	0	0	0	0.0000
Column 256	0				

**Hình 2.5:**  $q = h_A/(M \times N)$

Lọc nhiễu tín hiệu vào ảnh I, ước lượng được tín hiệu ảnh gốc G:

Columns 463 through 468					
7	7	7	7	7	6
7	7	7	7	7	6
8	8	7	6	6	5
7	8	7	7	6	6
7	7	8	7	6	5
7	7	8	6	7	5
7	8	8	7	6	5
7	8	7	7	6	6
7	8	7	7	7	5
8	8	7	6	6	6
8	8	8	7	7	7

**Hình 2.6:** Ma trận B của ảnh gốc G

Chuyển ma trận B thành ma trận 1 chiều.

11
10
10
10
10
11
10
10
10
10
11
10
11
10
11
10
10
10

**Hình 2.7:** Ma trận B(:)

Tính tần suất ( $h_G$ ) của ma trận B theo giá trị từ 0 đến 255

Columns 241 through 244	11	3	4	3
Columns 245 through 248	0	0	4	0
Columns 249 through 252	0	1	0	0
Columns 253 through 256	0	0	1	0

**Hình 2.8:**  $\text{hist}(B,0:255)$

Tính xác suất xuất hiện của mỗi giá trị ảnh trong ma trận B

Columns 241 through 245	0.0001	0.0000	0.0000	0.0000	0
Columns 246 through 250	0	0.0000	0	0	0.0000
Columns 251 through 255	0	0	0	0	0.0000
Column 256	0				

**Hình 2.9:**  $p = h_B|(M \times N)$

Tính hàm  $D(p||q)$

```
D =
-1.1102e-016
```

**Hình 2.10:**  $D(p||q)$

Giấu tin vào ảnh G với tỉ lệ  $R_0 = 0,05$ , được tín hiệu ảnh T.

```
Columns 463 through 468
     6     6     6     7     7     6
     6     7     7     6     7     7
     8     8     6     6     7     5
     7     8     6     6     6     7
     6     7     9     6     6     4
     7     7     8     7     6     5
     7     9     9     7     7     5
     7     9     6     7     7     7
```

**Hình 2.11:** Ma trận C của ảnh T

Chuyển ma trận C thành ma trận 1 chiều.

```
11
10
10
10
10
11
10
10
10
10
11
10
11
10
11
10
10
10
```

**Hình 2.12:** Ma trận C(:)

Tính tần suất ( $h_T$ ) của ma trận C theo giá trị từ 0 đến 255.

Columns 241 through 244	11	3	4	3
Columns 245 through 248	0	0	4	0
Columns 249 through 252	0	1	0	0
Columns 253 through 256	0	0	1	0

**Hình 2.13:**  $\text{hist}(C,0:255)$

Tính xác suất xuất hiện của mỗi giá trị ảnh trong ma trận C:

Columns 241 through 245	0.0001	0.0000	0.0000	0.0000	0
Columns 246 through 250	0	0.0000	0	0	0.0000
Columns 251 through 255	0	0	0	0	0.0000
Column 256	0				

**Hình 2.14:**  $u = h_c|(M \times N)$

Tính hàm  $D(q||u)$ .

```
D2 =  
9.4827e-004
```

**Hình 2.15:**  $D(q||u)$

Tính  $kq = D(q||u) - D(p||q)$ .

```
kq =  
9.4827e-004
```

**Hình 2.16:**  $kq = D(q||u) - D(p||q)$ .

So sánh  $kq$  với  $T(\alpha) = 0$ :

$Kq > T(\alpha) \rightarrow$  ảnh không có tin ẩn giấu.

## CHƯƠNG 3: CÀI ĐẶT VÀ THỬ NGHIỆM

### 3.1. MÔI TRƯỜNG CÀI ĐẶT

Các thử nghiệm dùng để đánh giá thuật toán và kỹ thuật giấu và phát hiện được thực hiện trên môi trường MATLAB phiên bản 2009a.

Chạy trên máy tính cấu hình Pentium (R) Dual-Core CPU T4200 2.00GHz, bộ nhớ trong 1 Gb, bộ nhớ ngoài có dung lượng tổng khả dụng 15GB.

Quá trình thực nghiệm cần sự hỗ trợ của phần mềm xử lý ảnh Photoshop phiên bản CS2 8.0 để chuyển đổi dữ liệu ảnh từ màu sang ảnh cấp xám thuận tiện cho các thuật toán.

### 3.2. MỘT SỐ GIAO DIỆN CHƯƠNG TRÌNH

**Giao diện chương trình chính bao gồm:**

- Giấu tin trên miền LSB
  - o Giấu theo tỉ lệ
  - o Giấu theo văn bản
- Phát hiện ảnh giấu tin trên miền LSB
  - o Phát hiện 1 ảnh
  - o Phát hiện trên 1 tập ảnh



**Hình 3.1:** Form chính của chương trình



**Hình 3.2:** Chức năng giấu tin



**Hình 3.3:** Chức năng phát hiện ảnh giấu tin





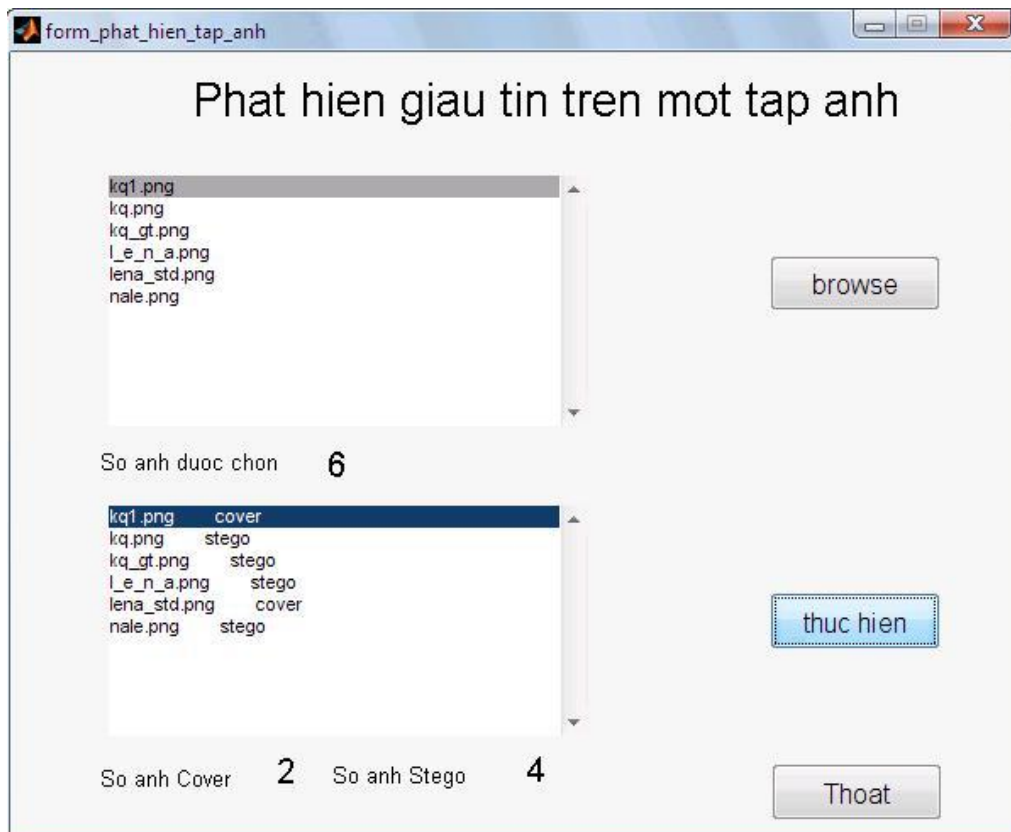
**Hình 3.4:** Giao diện giấu tin theo tỉ lệ



**Hình 3.5:** Giao diện giấu tin theo văn bản



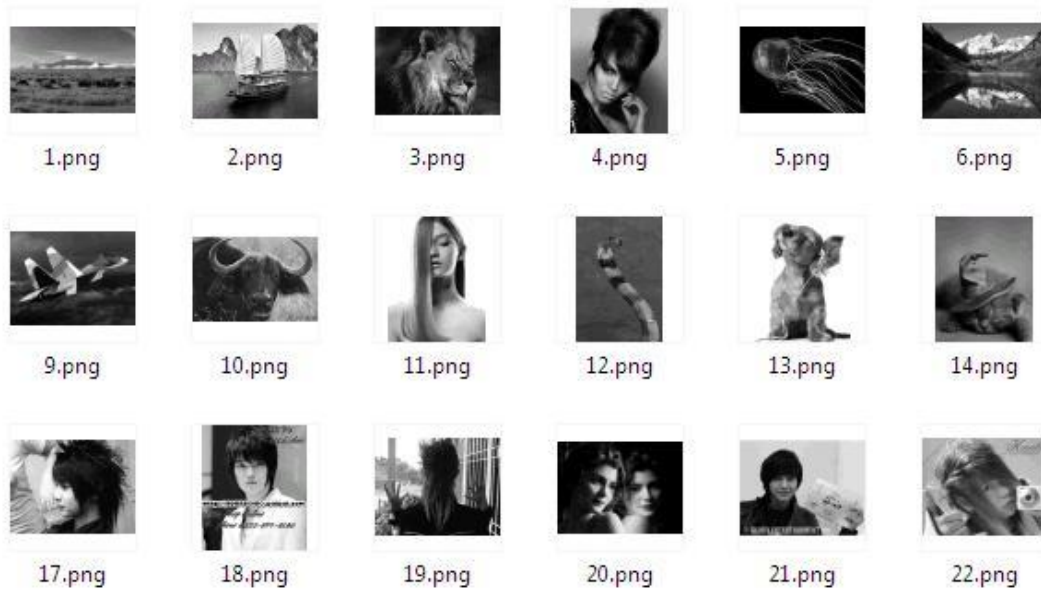
**Hình 3.6:** Giao diện kiểm tra giấu tin trên 1 ảnh



**Hình 3.7:** Giao diện phát hiện giấu tin trên 1 tập ảnh

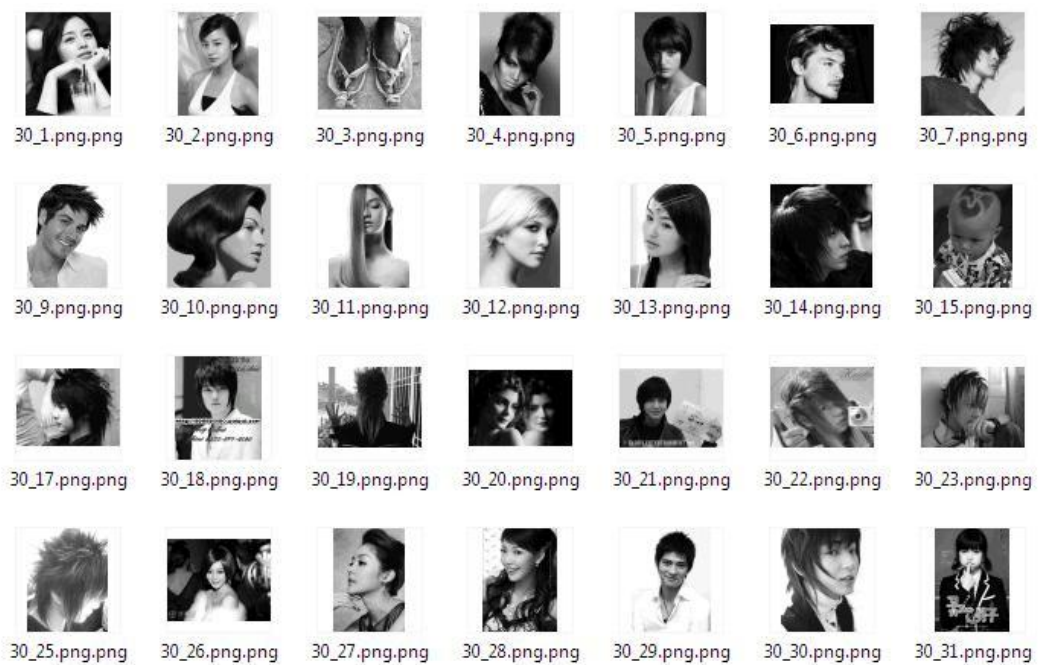
### 3.3. THỬ NGHIỆM

Thử nghiệm trên 100 ảnh xám không giấu tin:



**Hình 3.8:** 100 ảnh xám làm thử nghiệm

Thử nghiệm trên 100 ảnh xám giấu tin với tỉ lệ 30%:



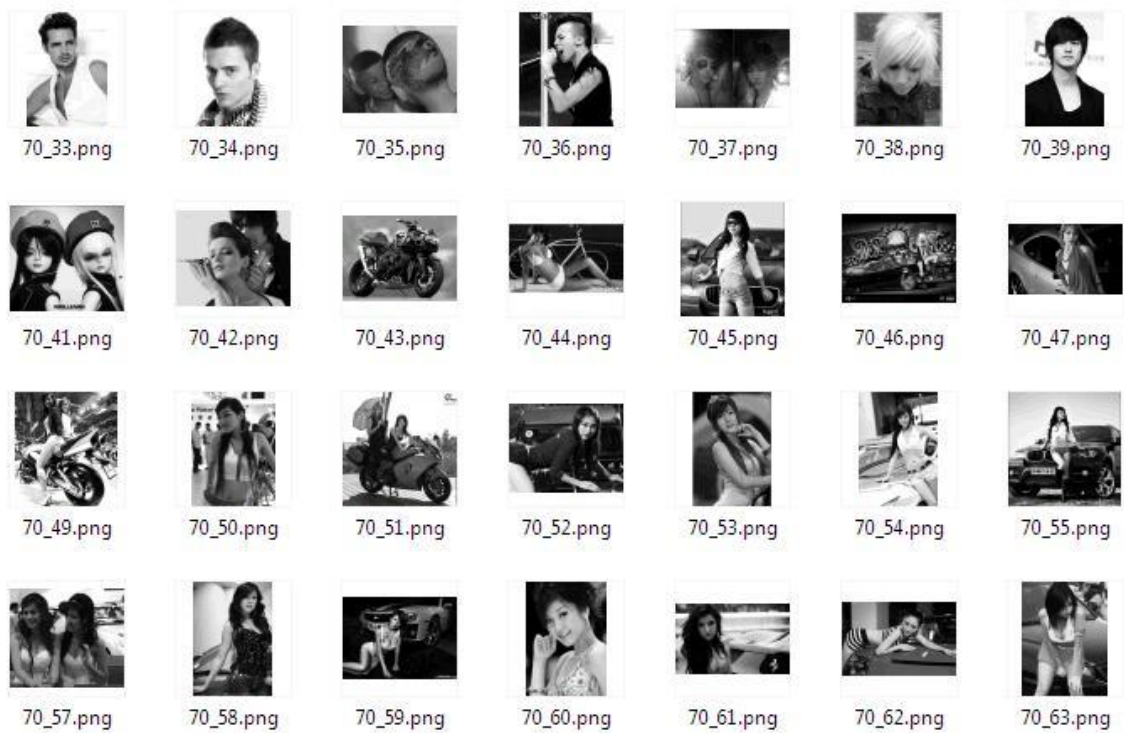
**Hình 3.9:** 100 ảnh xám giấu tin với tỉ lệ 30% làm thử nghiệm

Thử nghiệm trên 100 ảnh xám giấu tin với tỉ lệ 50%:



**Hình 3.10:** 100 ảnh xám giấu tin với tỉ lệ 50% làm thử nghiệm

Thử nghiệm trên 100 ảnh xám giấu tin với tỉ lệ 70%:



**Hình 3.11:** 100 ảnh xám giấu tin với tỉ lệ 70% làm thử nghiệm





### 3.4 ĐÁNH GIÁ KỸ THUẬT PHÁT HIỆN

#### 3.4.1 Độ đo đánh giá

Trong những thử nghiệm này, em sử dụng các độ đo đánh giá là: *precision*, *recall* và *f-measure* thường được áp dụng trong phân loại dữ liệu. *Precision* là độ đo tính chính xác và đúng đắn của việc phân loại. *Recall* là độ đo tính toàn vẹn của việc phân lớp.

Cụ thể cho bài toán phân loại ảnh có giấu tin và ảnh chưa giấu tin, giả sử ta có một tập ảnh đầu vào  $E$  (gồm cả ảnh giấu tin và ảnh chưa giấu tin) cần phân thành 2 tập con  $E_1$  (ảnh có giấu tin) và  $E_2$  (ảnh không giấu tin). Sau khi thực hiện phân lớp chúng ta được bảng sau:

		Kết quả phân lớp đạt đúng	
		$E_1$	$E_2$
Kết quả phân lớp đạt được	$E_1$	tp (true positive)	fp (false positive)
	$E_2$	fn (false negative)	tn (true negative)

Khi đó *precision* và *recall* được tính toán theo công thức sau:

$$Precision = \frac{tp}{tp + fp} \quad (3.1)$$

$$Recall = \frac{tp}{tp + fn} \quad (3.2)$$

Mặc dù *precision* và *recall* là những độ đo được dùng rộng rãi và phổ biến nhất, nhưng chúng lại gây khó khăn khi phải đánh giá các bài toán phân loại vì hai độ đo trên lại không tăng/giảm tương ứng với nhau. Bài toán đánh giá có *recall* cao có thể có *precision* thấp và ngược lại. Hơn nữa, việc so sánh mà chỉ dựa trên một mình *precision* và *recall* không phải là một ý hay. Với mục tiêu này,

độ đo *F-measure* được sử dụng để đánh giá tổng quát các bài toán phân loại. *F-measure* là trung bình điều hoà có trọng số của *precision* và *recall* và có công thức:

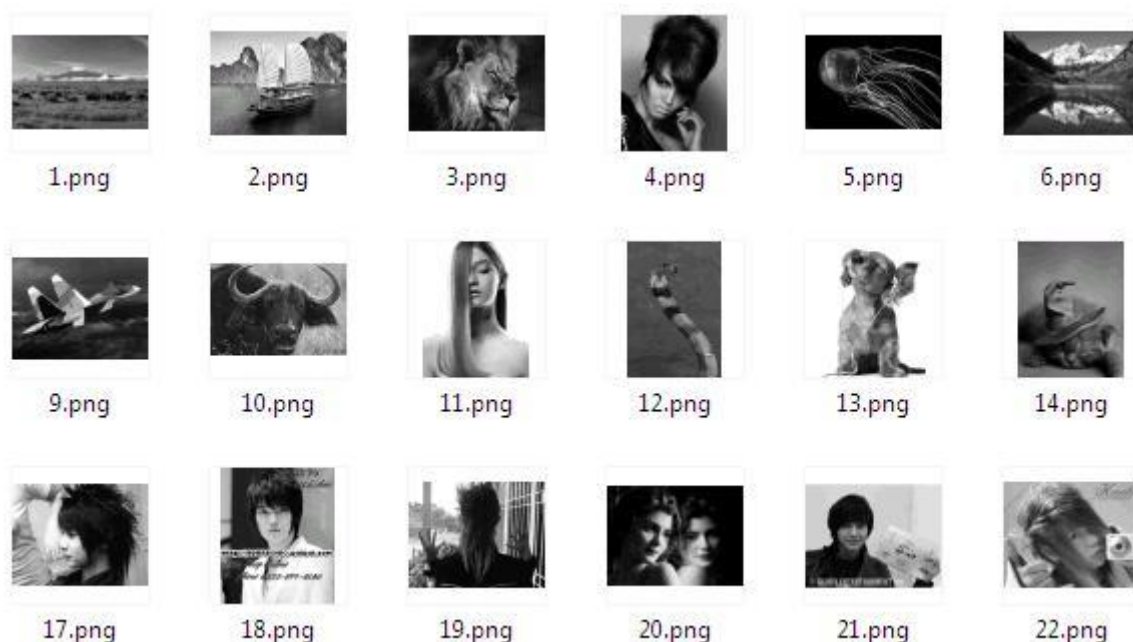
$$F_{\beta} = \frac{(\beta^2 + 1) \cdot \text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}}$$

trong đó  $\beta$  là một tham số có giá trị nằm giữa 0 và 1. Nếu  $\beta = 1$ , *F-measure* bằng với *precision* và nếu  $\beta = 0$ , *F-measure* bằng với *recall*. Giữa đoạn đó, giá trị  $\beta$  càng cao, độ quan trọng của *precision* càng cao so với *recall*. Ta sử dụng giá trị thường được dùng là  $\beta = 0.5$ , nghĩa là:

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (3.3)$$

### 3.4.2. Kết quả thử nghiệm đánh giá

Tập ảnh thử nghiệm D1 gồm 50 ảnh có giấu tin với lượng giấu 50%, 100%, kích thước tùy ý và D2 gồm 50 ảnh không giấu tin. Chọn tập ảnh D50\_percent gồm 100 ảnh trong đó có 50 ảnh có giấu tin với lượng giấu 50% (D1) và 50 ảnh không giấu tin (D2). Một tập ảnh khác D100\_percent gồm 100 ảnh với 50 ảnh có giấu với lượng giấu 100% (D1) và 50 ảnh không giấu tin(D2).



**Hình 3.13:** 50 ảnh xám không giấu tin làm thử nghiệm



**Hình 3.14:** 50 ảnh xám giấu tin với tỉ lệ 50% làm thử nghiệm



**Hình 3.15:** 50 ảnh xám giấu tin với tỉ lệ 100% làm thử nghiệm



		Kết quả phân lớp đúng	
		D <sub>1</sub>	D <sub>2</sub>
Kết quả phân lớp đạt được	D <sub>1</sub>	36	12
	D <sub>2</sub>	14	38

**Hình 3.16:** Bảng tổng hợp kết quả của tập thử nghiệm D50\_percent

Áp dụng công thức (5.1) và (5.2) và (5.3) ta có:

$$\text{Precision} = \frac{36}{36+12} = 0,75$$

$$\text{Recall} = \frac{36}{36+14} = 0,72$$

$$F - \text{measure} = 2 * \frac{0,75 * 0,72}{0,75 + 0,72} = 0,73$$

		Kết quả phân lớp đạt đúng	
		D <sub>1</sub>	D <sub>2</sub>
Kết quả phân lớp đạt được	D <sub>1</sub>	39	13
	D <sub>2</sub>	11	37

**Hình 3.17:** Bảng tổng hợp kết quả của tập thử nghiệm D100\_percent

Áp dụng công thức (3.1) và (3.2) và (3.3) ta có:

$$\text{Precision} = \frac{39}{39+13} = 0,75$$

$$\text{Recall} = \frac{39}{39+11} = 0,78$$

$$\text{F - measure} = 2 * \frac{0,75 * 0,78}{0,75+0,78} = 0,76$$

Độ đo Kỹ thuật	Precision	Recall	F – measure
Kỹ thuật phát hiện cho lượng giấu 50%	0.75	0.72	0.73
Kỹ thuật phát hiện cho lượng giấu 100%	0.75	0.78	0.76

**Hình 3.18:** Bảng thử nghiệm trên hai tập ảnh D50\_percent và D100\_percent

**Nhận xét:**

Với tỉ lệ phát hiện đúng trên 70%, phương pháp phát hiện ảnh giấu tin LLRT có độ chính xác chỉ đạt mức trung bình.

## KẾT LUẬN

Đồ án của em đã thực hiện những nhiệm vụ sau:

1. Trình bày một số khái niệm cơ bản về: Giấu tin trong ảnh, phát hiện giấu tin trong ảnh, tổng quan về ảnh Bitmap, kỹ thuật giấu tin trên LSB, lọc nhiễu tín hiệu bằng phương pháp lọc thông thấp
2. Kỹ thuật phát hiện ảnh có giấu tin trên miền LSB bằng phương pháp thống kê LLRT.

Đây là một trong những phương pháp phát hiện ảnh có giấu tin rất phổ biến hiện nay. Với sự phát triển một cách bùng nổ của ngành công nghệ thông tin hiện nay, chúng ta cũng phải bắt kịp sự phát triển của thế giới để có thể tự bảo vệ quyền lợi của bản thân, của quốc gia.

Do còn nhiều hạn chế về thời gian nghiên cứu nên đề tài này không tránh khỏi những thiếu sót, vì vậy em rất mong nhận được sự đóng góp ý kiến của các thầy cô cùng các bạn để báo cáo của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

## TÀI LIỆU THAM KHẢO

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

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

[3] K. Sullivan, O. Dabeer, U. Madow, B. S. Manujunath and S. Chandrasekaran (Sep. 2003), *LLRT Based Detection of LSB Hiding*, In Proc. IEEE International Conference on Image Processing (ICIP), Barcelona, Spain, pp. 497–500.

[4] <http://vi.wikipedia.org/wiki/>