**BAN CƠ YẾU CHÍNH PHỦ**

**HỌC VIỆN KỸ THUẬT MẬT MÃ**

****

ĐỒ ÁN TỐT NGHIỆP

**NGHIÊN CỨU CƠ CHẾ BẢO MẬT THÔNG TIN TRONG VIDEO THÍCH HỢP VÀ ỨNG DỤNG TRONG BẢO VỆ BẢN QUYỀN VIDEO.**

*Ngành: An Toàn Thông Tin*

*Mã đào tạo: 52.48.02.02*

*Sinh viên thực hiện:* **Trần Hữu Anh Văn.**

*MSSV:* **AT120949.**

*Lớp:* **AT12KU.**

*Giáo viên hướng dẫn:* **PGS.TS Nguyễn Thanh Bình.**

*Đơn vị:* **Trường Đại học Bách Khoa – ĐHQG TPHCM.**

TP. Hồ Chí Minh, năm 2020.

**BAN CƠ YẾU CHÍNH PHỦ**

**HỌC VIỆN KỸ THUẬT MẬT MÃ**

****

ĐỒ ÁN TỐT NGHIỆP

**NGHIÊN CỨU CƠ CHẾ BẢO MẬT THÔNG TIN TRONG VIDEO THÍCH HỢP VÀ ỨNG DỤNG TRONG BẢO VỆ BẢN QUYỀN VIDEO.**

*Ngành: An Toàn Thông Tin*

*Mã đào tạo: 52.48.02.02*

*Sinh viên thực hiện:* **Trần Hữu Anh Văn.**

*MSSV:* **AT120949.**

*Lớp:* **AT12KU.**

*Giáo viên hướng dẫn:* **PGS.TS Nguyễn Thanh Bình.**

*Đơn vị:* **Trường Đại học Bách Khoa – ĐHQG TPHCM.**

TP. Hồ Chí Minh, năm 2020.

# 

# LỜI CẢM ƠN

Em xin bày tỏ lòng biết ơn chân thành và sâu sắc đến PGS.TS Nguyễn Thanh Bình, Thầy đã dành nhiều thời gian tận tình chỉ bảo, hướng dẫn em trong suốt quá trình tìm hiểu, triển khai và nghiên cứu đề tài. Thầy là người đã định hướng và đưa ra nhiều góp ý quý báu trong quá trình em thực hiện luận văn này.

Em xin chân thành cảm ơn chân thành tới toàn thể các thầy giáo, cô giáo trong Khoa An toàn thông tin – Học Viện Kỹ Thuật Mật Mã đã dạy bảo tận tình, trang bị cho em những kiến thức quý báu, bổ ích và tạo điều kiện thuận lợi trong suốt quá trình em học tập và nghiên cứu tại trường.

Em cũng xin chân thành cảm ơn tới gia đình, bạn bè đã luôn bên em cổ vũ, động viên, giúp đỡ em trong suốt quá trình học tập và thực hiện đồ án.

Do có nhiều hạn chế về thời gian và kiến thức nên đồ án không tránh khỏi những thiếu sót, rất mong nhận được những ý kiến đóng góp quý báu của quý thầy cô và các bạn cùng quan tâm.

Cuối cùng em xin gửi lời chúc sức khỏe và thành đạt tới tất cả quý thầy cô, cùng toàn thể gia đình và bạn bè. Xin chân thành cảm ơn!

**SINH VIÊN THỰC HIỆN ĐỒ ÁN**

# LỜI CAM ĐOAN

Tôi xin cam đoan đồ án tốt nghiệp là công trình nghiên cứu của bản thân. Những phần sử dụng tài liệu tham khảo trong đồ án đã được nêu rõ trong phần tài liệu tham khảo. Các số liệu, kết quả trình bày trong đồ án là hoàn toàn trung thực, nếu sai tôi xin chịu hoàn toàn trách nhiệm và chịu mọi kỷ luật của bộ môn và nhà trường đề ra.

Hồ Chí Minh, tháng 12 năm 2020.

Sinh viên

# LỜI MỞ ĐẦU

Trong những năm gần đây, sự phát triển vượt bậc của công nghệ internet và sự mở rộng của các thiết bị tính toán mạnh mẽ không chỉ thúc đẩy thương mại điện tử đa phương tiện đi lên mà còn kích thích các nghệ sĩ chia sẻ và quảng bá tác phẩm của họ trực tuyến. Điều này rõ ràng ngụ ý sự hiện diện lớn trên web của dữ liệu đa phương tiện kỹ thuật số như âm thanh, hình ảnh và video. Tuy nhiên, với sự phổ biến và dễ sử dụng của các công cụ xử lý chuyên dụng đa phương tiện mạnh mẽ, những dữ liệu này có thể được tải xuống, dễ dàng sửa đổi, chiếm đoạt bất hợp pháp và sau đó phần lớn được phân phối lại hoặc thương mại hóa trên Internet. Bảo vệ quyền sở hữu trí tuệ của chủ sở hữu sau đó đã trở thành một mối quan tâm lớn.

Từ đó đồ án “ ***Nghiên cứu cơ chế bảo mật thông tin trong video thích hợp và ứng dụng bảo vệ bản quyền video***” mục đích tìm hiểu các kĩ thuật giấu tin trong video và phân tách thông tin bí mật để bảo vệ chủ sở hữu bản quyền.

# MỤC LỤC

[LỜI CẢM ƠN i](#_Toc60160098)

[LỜI CAM ĐOAN ii](#_Toc60160099)

[LỜI MỞ ĐẦU iii](#_Toc60160100)

[MỤC LỤC iv](#_Toc60160101)

[MỤC LỤC BẢNG BIỂU MẪU viii](#_Toc60160102)

[MỤC LỤC HÌNH VẼ ix](#_Toc60160103)

[CHƯƠNG I: GIỚI THIỆU 1](#_Toc60160104)

[1.1. Giới thiệu đề tài 1](#_Toc60160105)

[1.2. Mục tiêu nội dung đề tài 1](#_Toc60160106)

[1.3. Giới hạn đề tài 1](#_Toc60160107)

[1.4. Cấu trúc báo cáo 2](#_Toc60160108)

[CHƯƠNG II: CƠ SỞ LÝ THUYẾT VÀ CÁC NGHIÊN CỨU LIÊN QUAN 3](#_Toc60160109)

[2.1. Cơ sở lý thuyết 3](#_Toc60160110)

[2.1.1. Bảo mật thông tin giấu trong video 3](#_Toc60160111)

[2.1.2. Sơ đồ giấu tin tổng quát 3](#_Toc60160112)

[2.1.3. Kỹ thuật giấu tin mật trong video và các nghiên cứu liên quan 4](#_Toc60160113)

[2.1.3.1. Kỹ thuật giấu tin mật trong video 4](#_Toc60160114)

[2.1.3.2. Phương pháp thực hiện giấu tin bí mật (steganography) 6](#_Toc60160115)

[2.1.4.3. Các nghiên cứu liên quan 8](#_Toc60160116)

[2.1.4. Kỹ thuật đánh dấu thủy vân (watermark) và các nghiên cứu liên quan 9](#_Toc60160117)

[2.1.4.1. Kỹ thuật thủy vân 9](#_Toc60160118)

[2.1.5. Video và cấu trúc định dạng một số tệp video 12](#_Toc60160119)

[2.1.5.1. Khung và cấu trúc khung trong video 12](#_Toc60160120)

[2.1.5.2. Một số loại định dạng video phổ biến 13](#_Toc60160121)

[2.1.6. Hệ mật RSA 14](#_Toc60160122)

[2.1.6.1. Hoạt động 14](#_Toc60160123)

[2.1.6.2. Tạo khóa 15](#_Toc60160124)

[2.1.6.3. Mã hóa 15](#_Toc60160125)

[2.1.6.4. Giải mã 15](#_Toc60160126)

[2.2. Các nghiên cứu liên quan 16](#_Toc60160127)

[2.2.1. Truyền dữ liệu an toàn dựa trên video steganography (SLSB) 16](#_Toc60160128)

[2.2.1.1 Thuật toán mã hóa dựa trên kỹ thuật truyền bảo mật 16](#_Toc60160129)

[2.2.1.2. Thuật toán giải mã dựa trên kỹ thuật truyền bảo mật 16](#_Toc60160130)

[2.2.2. Kỹ thuật bit chủ yếu ít nhất Hash cho video steganography (HLSB) 16](#_Toc60160131)

[2.2.3. Video Steganography bằng LSB thay thế sử dụng các phương trình đa thức khác nhau (LSB Poly) 18](#_Toc60160132)

[2.2.2.1. Mã hóa để thay thế bằng các phương trình đa thức khác nhau 18](#_Toc60160133)

[2.2.2.2. Giải mã để thay thế bằng các phương trình đa thức khác nhau 19](#_Toc60160134)

[2.2.3. Stego Machine- Video Steganography sử dụng thuật toán LSB sửa đổi (MLSB) 20](#_Toc60160135)

[2.2.4. Video Steganography sử dụng thuật toán xem lại đối sánh LSB (LSBMR) 21](#_Toc60160136)

[2.2.4.1. Thuật toán mã hóa trong nội dung ẩn video LSBMR 21](#_Toc60160137)

[2.2.4.2. Thuật toán giải mã trong nội dung video LSBMR 22](#_Toc60160138)

[CHƯƠNG III: PHƯƠNG PHÁP ĐỀ XUẤT 23](#_Toc60160139)

[3.1. Yêu cầu bài toán 23](#_Toc60160140)

[3.2. Phương pháp đề xuất 23](#_Toc60160141)

[3.2.1. Thuật toán mã hóa và nhúng 24](#_Toc60160142)

[3.2.1.1. Input 24](#_Toc60160143)

[3.2.1.2. Output 25](#_Toc60160144)

[3.2.1.3. Quá trình thực hiện 25](#_Toc60160145)

[3.2.2. Sơ đồ mã hóa và nhúng 25](#_Toc60160146)

[3.2.3. Thuật toán giải mã và trích xuất 27](#_Toc60160147)

[3.2.3.1. Input 27](#_Toc60160148)

[3.2.3.2. Ouput 28](#_Toc60160149)

[3.2.3.3. Quá trình thực hiện 28](#_Toc60160150)

[3.2.4. Sơ đồ giải mã và trích xuất 29](#_Toc60160151)

[3.2.5. Phương pháp đánh giá 30](#_Toc60160152)

[3.2.5.1. Đánh giá chủ quan. 30](#_Toc60160153)

[3.2.5.2. Đánh giá khách quan. 30](#_Toc60160154)

[CHƯƠNG IV: CÀI ĐẶT, THỬ NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ 31](#_Toc60160155)

[4.1. Cài đặt 31](#_Toc60160156)

[4.1.1. Giao diện mã hóa nhúng 31](#_Toc60160157)

[4.1.2. Giao diện trích xuất/ giải mã 34](#_Toc60160158)

[4.1.3. Quá trình thực hiện 34](#_Toc60160159)

[4.1.3.1. Mã hóa/ nhúng thông điệp 34](#_Toc60160160)

[4.1.3.2. Trích xuất/ giải mã 35](#_Toc60160161)

[4.2. Tập dữ liệu đầu vào 35](#_Toc60160162)

[4.2.1. Bộ dữ liệu đầu vào 1 36](#_Toc60160163)

[4.2.2. Bộ dữ liệu đầu vào 2 37](#_Toc60160164)

[4.2.3. Bộ dữ liệu đầu vào 3 37](#_Toc60160165)

[4.2.4. Bộ dữ liệu đầu vào 4. 38](#_Toc60160166)

[4.2. Thí nghiệm đánh giá kết quả. 38](#_Toc60160167)

[4.3.1. Kết quả đạt được tương ứng bộ dữ liệu đầu vào 1 39](#_Toc60160168)

[4.3.1.1. Video đã nhúng 39](#_Toc60160169)

[4.3.1.2. Kết quả quá trình mã hóa/ nhúng 40](#_Toc60160170)

[4.3.1.3. Kết quả trích xuất/ giải mã 40](#_Toc60160171)

[4.3.2. Kết quả đạt được tương ứng bộ dữ liệu đầu vào 2 41](#_Toc60160172)

[4.3.2.1. Video đã nhúng 41](#_Toc60160173)

[4.3.2.2. Kết quả quá trình mã hóa/ nhúng 41](#_Toc60160174)

[4.3.2.3. Kết quả trích xuất/ giải mã 42](#_Toc60160175)

[4.3.3. Kết quả đạt được tương ứng bộ dữ liệu đầu vào 3 43](#_Toc60160176)

[4.3.3.1. Video đã nhúng 43](#_Toc60160177)

[4.3.3.2. Kết quả quá trình mã hóa/ nhúng 43](#_Toc60160178)

[4.3.3.3. Kết quả trích xuất/ giải mã 44](#_Toc60160179)

[4.3.4. Kết quả đạt được tương ứng bộ dữ liệu đầu vào 4 45](#_Toc60160180)

[4.3.4.1. Video đã nhúng 45](#_Toc60160181)

[4.3.4.2. Kết quả quá trình mã hóa/ nhúng 45](#_Toc60160182)

[4.3.4.3. Kết quả trích xuất/ giải mã 46](#_Toc60160183)

[4.3.5. Đánh giá kết quả 46](#_Toc60160184)

[4.3.5.1. Ưu điểm 46](#_Toc60160185)

[4.3.5.2. Nhược điểm 46](#_Toc60160186)

[CHƯƠNG V: KẾT LUẬN 47](#_Toc60160187)

[5.1. Kết quả đạt được 47](#_Toc60160188)

[5.2. Ưu nhược điểm phương pháp đề xuất 47](#_Toc60160189)

[5.3. Hướng mở rộng tương lai 47](#_Toc60160190)

[TÀI LIỆU THAM KHẢO 48](#_Toc60160191)

# MỤC LỤC BẢNG BIỂU MẪU

[Bảng 4. 1. Tập dữ liệu đầu vào. 35](#_Toc60095096)

[Bảng 4. 2. Thông số bộ dữ liệu đầu vào- file Text. 36](#_Toc60095097)

[Bảng 4. 3 Đánh giá kết quả tương ứng. 38](#_Toc60095098)

# MỤC LỤC HÌNH VẼ

[Hình 2. 1. Sơ đồ giấu tin tổng quát. 3](file:///C:\Users\bhanh\OneDrive\Desktop\Đồ%20án\word\Báo%20cáo%20đồ%20án.docx#_Toc59836971)

[Hình 2. 2. Sơ đồ trích xuất tổng quát. 4](file:///C:\Users\bhanh\OneDrive\Desktop\Đồ%20án\word\Báo%20cáo%20đồ%20án.docx#_Toc59836972)

[Hình 2. 3. Sơ đồ giấu tin tổng quát. 5](#_Toc59836973)

[Hình 2. 4. Sơ đồ giấu tin sử dụng thuật toán HLSB. 19](file:///C:\Users\bhanh\OneDrive\Desktop\Đồ%20án\word\Báo%20cáo%20đồ%20án.docx#_Toc59836974)

[Hình 2. 5. Sơ đồ trích xuất sử dụng thuật toán HLSB 20](#_Toc59836975)

[Hình 3. 1. Sơ đồ giấu tin phương pháp đề xuất. 24](file:///C:\Users\bhanh\OneDrive\Desktop\Đồ%20án\word\Báo%20cáo%20đồ%20án.docx#_Toc59836977)

[Hình 3. 2. Sơ đồ giấu tin bằng LSB. 27](file:///C:\Users\bhanh\OneDrive\Desktop\Đồ%20án\word\Báo%20cáo%20đồ%20án.docx#_Toc59836978)

[Hình 3. 3. Sơ đồ trích xuất thông điệp bằng thuật toán LSB. 29](file:///C:\Users\bhanh\OneDrive\Desktop\Đồ%20án\word\Báo%20cáo%20đồ%20án.docx#_Toc59836979)

[Hình 4. 1 Giao diện chương trình mã hóa nhúng. 31](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164460)

[Hình 4. 2 Giao diện chọn file Text chứa thông điệp 32](#_Toc60164461)

[Hình 4. 3 Giao diện chọn video mang thông điệp. 32](#_Toc60164462)

[Hình 4. 4 Mật khẩu mã hóa 33](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164463)

[Hình 4. 5 Lỗi khi thiếu thông tin đầu vào. 33](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164464)

[Hình 4. 6 Giao diện trích xuất giải mã. 34](#_Toc60164465)

[Hình 4. 7 Thông số bộ dữ liệu đầu vào 1. 36](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164466)

[Hình 4. 8 Thông số bộ dữ liệu đầu vào 2. 37](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164467)

[Hình 4. 9 Thông số bộ dữ liệu đầu vào 3. 37](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164468)

[Hình 4. 10 Thông số bộ dữ liệu đầu vào 4. 38](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164469)

[Hình 4. 11 Kết quả bộ dữ liệu đầu vào 1 - video nhúng thành công. 39](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164470)

[Hình 4. 12 Kết quả bộ dữ liệu đầu vào 1 - mã hóa/ nhúng thành công. 40](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164471)

[Hình 4. 13 Kết quả bộ dữ liệu đầu vào 1 - trích xuất/ giải mã thành công. 40](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164472)

[Hình 4. 14 Kết quả bộ dữ liệu đầu vào 2 - video nhúng thành công. 41](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164473)

[Hình 4. 15 Kết quả bộ dữ liệu đầu vào 2 - mã hóa/ nhúng thành công. 41](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164474)

[Hình 4. 16 Kết quả bộ dữ liệu đầu vào 2 - trích xuất/ giải mã thành công. 42](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164475)

[Hình 4. 17 Kết quả bộ dữ liệu đầu vào 3 - video nhúng thành công. 42](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164476)

[Hình 4. 18 Kết quả bộ dữ liệu đầu vào 3 - mã hóa/ nhúng thành công. 43](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164477)

[Hình 4. 19 Kết quả bộ dữ liệu đầu vào 3 - trích xuất/ giải mã thành công. 43](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164478)

[Hình 4. 20 Kết quả bộ dữ liệu đầu vào 4 - video nhúng thành công. 44](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164479)

[Hình 4. 21 Kết quả bộ dữ liệu đầu vào 4 - mã hóa/ nhúng thành công. 44](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164480)

[Hình 4. 22 Kết quả bộ dữ liệu đầu vào 4 - trích xuất/ giải mã thành công. 45](file:///C:\Users\bhanh\OneDrive\Desktop\baocaodoan.docx#_Toc60164481)

# CHƯƠNG I: GIỚI THIỆU

## Giới thiệu đề tài

Sự phát triển nhanh chóng của internet và công nghệ đa phương tiện đã kéo theo sự phổ biến ngày càng tăng của các video kỹ thuật số như TV mạng, video trực tuyến và video di động. Những ứng dụng video này làm phong phú thêm cuộc sống của mọi người và tạo ra nhiều tiện lợi. Tuy nhiên, sự gia tăng của video và chia sẻ video cũng gây ra nhiều vấn đề nghiêm trọng. Với sự trợ giúp của các công cụ chỉnh sửa video khác nhau, mọi người có thể dễ dàng chỉnh sửa nội dung video công khai một cách bất hợp pháp, cũng như sao chép và phán tán nó một cách tự do. Những hành vi này ảnh hưởng đến quyền lợi chính đáng của chủ sở hữu bản quyền. Để bảo vệ quyền sở hữu bản quyền video gốc hoặc cung cấp xác thực nội dung của các tác phẩm gốc đã trở thành vấn đề cấp bách cần phải đối mặt. Ý tưởng cơ bản là thêm một số thông tin bổ sung vào video gốc mà không ảnh hưởng đến khả năng hiển thị của video, điều này có thể cung cấp bằng chứng về bản quyền hoặc xác thực tính toán vẹn nội dung khi cần thiết.

## 1.2. Mục tiêu nội dung đề tài

Mục tiêu nghiên cứu của đề tài là thực hiện nghiên cứu cơ chế bảo mật thông tin trong video thích hợp và ứng dụng trong bảo vệ bản quyền video. Xây dựng cơ chế bảo mật thông tin trong video qua đó xây dựng ứng dụng để xác định định danh bảo vệ quyền video.

Để thực hiện được các mục tiêu trên, đồ án sẽ thực hiện các nội dung sau:

1. Tìm hiểu cơ chế bảo mật thông tin trong video.
2. Tìm hiểu các công trình nghiên cứu liên quan và phân tích ưu nhược điểm của chúng.
3. Đề xuất cơ chế bảo mật thông tin trong video.
4. Xây dựng ứng dụng bảo mật thông tin trong video để xác định bản quyền của tác giả.
5. Đánh giá và so sánh các cơ chế bảo mật trong bảo vệ bản quyền tác giả.

## 1.3. Giới hạn đề tài

Đối tượng nghiên cứu của đề tài là ảnh số ( khung hình tách ra của video), âm thanh ( âm thanh được tách ra của video ) của video. Đề tài sử dụng phương pháp LSB (Least Significant Bits) để thực hiện kỹ thuật giấu tin trong video. Thông điệp cần ẩn giấu chứa trong file văn bản (.txt) được mã hóa bằng thuật toán RSA. Kết quả mã hóa thông điệp được tiến hành nhúng bằng phương pháp LSB vào khung hình và âm thanh được tách ra từ video đầu vào. Kết quả của quá trình là video được nhúng thông điệp bí mật có chất lượng không thay đổi so với video gốc.

## 1.4. Cấu trúc báo cáo

Đồ án được tổ chức thành 5 chương với nội dung chính như sau

* ***Chương 1****: Tổng quan về đề tài nghiên cứu*. Chương này trình bày tổng quan những vấn đề nghiên cứu của luận án. Thứ nhất giới thiệu đề tài nghiên cứu. Thứ hai mục tiêu nội dung đề tài nghiên cứu hướng đến. Thứ ba giới hạn đề tài .
* ***Chương 2****: Cơ sở lý thuyết các nghiên cứu liên quan*. Chương này trình bày tổng quan về những cơ sở lý thuyết liên quan. Thứ nhất là khái niệm và phân loại bảo mật thông tin giấu trong video. Thứ hai là tổng quan về giấu thông tin. Thứ ba là tổng quan về watermark (đánh dấu thủy vân). Thứ tư là phân tích và đánh giá các kỹ thuật liên quan cơ chế bảo mật thông tin trong video.
* ***Chương 3****: Phương pháp đề xuất*. Chương này trình bày thứ nhất về yêu cầu bài toán. Thứ hai đưa ra thuật toán đề xuất, trình bày chi tiết về thuật toán này. Thứ ba đưa ra phương pháp đánh giá về thuật toán đề xuất.
* ***Chương 4****: Thử nghiệm và đánh giá kết quả*. Chương này trình bày quá trình mã hóa rồi đưa thông điệp bí mật vào video và quá trình trích xuất rồi giải mã để lấy ra thông điệp ban đầu. Đưa ra các số liệu kết quả thực nghiệm.
* ***Chương 5****: Kết luận*. Chương này trình bày kết quả đạt được của đồ án. Đưa ra ưu nhược điểm phương pháp mà đồ án đề xuất. Trình bày hướng mở rộng trong tương lai.

# CHƯƠNG II: CƠ SỞ LÝ THUYẾT VÀ CÁC NGHIÊN CỨU LIÊN QUAN

## 2.1. Cơ sở lý thuyết

## Bảo mật thông tin giấu trong video

Tiềm năng ngày càng tăng của truyền thông hiện đại cần các phương tiện bảo mật đặc biệt trong mạng máy tính. An ninh mạng ngày càng trở nên quan trọng khi thách thức trao đổi dữ liệu trên Internet tăng lên. Do đó, tính bảo mật và tính toàn vẹn của dữ liệu được yêu cầu bảo vệ chống lại việc truy cập và sử dụng trái phép. Điều này dẫn đến sự phát triển của lĩnh vực giấu thông tin.

Giấu thông tin là một sự bổ sung của định hướng ứng dụng thông tin đến tín hiệu đa phương tiện, mà không gây ra bấy kỳ biến dạng nào có thể cảm nhận được. Người dùng không biết nhưng máy tính thì có thể phát hiện ra. Phần lớn các nghiên cứu đã được thực hiện về giấu thông tin qua các phương pháp ẩn (steganography) và đánh dấu thủy vân (watermarking).

* Steganography: đề cập đến các phương pháp được sử dụng để truyền thông điệp được nhúng mà người quan sát không thể phát hiện thông điệp. Hay nhằm bảo mật thông điệp đem giấu vào “vật mang tin” (video).
* Watermarking: đảm bảo tính bảo mật của thông điệp bằng cách nhúng một “hình mờ” cho mục đích xác thực, một bước quan trọng để bảo vệ quan quyền và phát hiện giả mạo. Hay nhằm bảo vệ chính “vật mang tin” (video). Hình mờ được nhúng thường là có thể nhận ra và không thể xóa dễ dàng khỏi “vật mang tin”. Chúng thường được yêu cầu là nửa mỏng manh (semi-fragile) or mạnh mẽ (robust).

### Sơ đồ giấu tin tổng quát

Vật mang tin gốc C

Khóa mã K.

Thông tin cần giấu M

Kỹ thuật giấu tin ( thuật toán )

Vật mang thông tin “sau khi giấu” S

Hình 2. 1. Sơ đồ giấu tin tổng quát.

Sơ đồ giấu thông tin tổng quát gồm quá trình giấu tin và quá trình trích tin [1]. Khái quát quá trình giấu tin và trích tin như 2 sơ đồ trong hình 2.1 và 2.2 dưới đây. Trên hình 2 ngoài thông tin cần giấu M và vật mang tin gốc C, mỗi thuật toán nhúng tin đều được trang bị khóa mã K để nâng cao sự an toàn cho hệ thống. Vì trong các ứng dụng truyền thông tin, thông thường các “vật mang tin” S đều bị công khai. Do vậy việc sử dụng hệ thống trao đổi khóa bí mật trong thuật toán giấu tin rất quan trọng, ngoài việc bảo mật, nó còn phục vụ cho việc trích tin.

Sau khi nhúng tin, “vật mang tin” S được truyền trên các hệ thống thông tin có thể bí mật hoặc công khai. Ở phía người nhận, thủ tục trích tin được   
trình bày trong hình 3. Đối với người dùng hợp lệ, ngoài việc trích tin để nhận được thông tin M, người nhận còn phải kiểm tra xem tính xác thực và toàn vẹn của “vật mang tin” S xem có bị tấn công hay không?

Vật mang tin gốc C

Tách thông tin cần giấu M

Khóa mã K

Kỹ thuật giấu tin (thuật toán)

“Vật mang tin” (sau khi giấu) S

Hình 2. 2. Sơ đồ trích xuất tổng quát.

Như đã trình bày ở trên, giấu tin là phương pháp nhúng thông điệp M vào vật mang tin gốc C để nhận được đối tượng S. Khi nhúng M vào C, các kỹ thuật giấu tin thường phải biến đổi theo một thuật toán nào đó để nhận được S. Do vậy, giữa S và C bao giờ cũng có sự sai khác nhất định. Sự sai khác này có thể được phát hiện bằng các chương trình, hoặc bằng hệ thống thị giác đối với dữ liệu dạng hình ảnh, hoặc bằng hệ thống thính giác đối với dữ liệu âm thanh. Thuật toán có tính che giấu càng cao thì càng khó phát hiện. Nói cách khác, sự sai khác giữa vật mang tin gốc C và “vật mang tin” S càng ít thi tính che giấu càng cao [2].

### Kỹ thuật giấu tin mật trong video và các nghiên cứu liên quan

#### Kỹ thuật giấu tin mật trong video

Có nhiều giao thức và kỹ thuật nhúng khác nhau cho phép “giấu dữ liệu” trong đối tượng đã cho. Tuy nhiên, tất cả các giao thức và kỹ thuật phải đáp ứng một số các yêu cầu sau:

* Thứ nhất: tính toàn vẹn của thông tin ẩn sau khi nó được nhúng vào bên trong đối tượng stego phải đúng. Thông điệp bí mật không được thay đổi theo bất kỳ cách nào, chẳng hạn như bổ sung thông tin được them vào, mất thông tin hoặc thay đổi thông tin bí mật sau đó.
* Thứ hai: đối tượng Stego phải không thay đổi hoặc hầu như không thay đổi với mắt thường. Nếu đối tượng Stego thay đổi đáng kể và có thể nhận thấy thì bên thứ ba có thể biết và cố gắng trích xuất hoặc loại bỏ nó.
* Thứ ba: trong watermark, những thay đổi trong đối tượng Stego không được ảnh hưởng đến watermark. Tưởng tượng nếu bạn có một bản sao bất hợp pháp, bạn muốn chỉnh sửa kích thước, thời lượng,… thì hình mờ phải tồn tại sau những thao tác này. Nếu không thì quá dễ để loại bỏ hình mờ và điểm Steganography sẽ bị hỏng.
* Cuối cùng giả sử kênh tấn công biết có thông tin ẩn bên trong đội tượng Stego.

Cover video

Secret text

Original

cover

Key

Secret text

Stego Object

Decoder

Encoder

Hình 2. 3. Sơ đồ giấu tin tổng quát.

Bước đầu tiên trong việc nhúng và ẩn thông tin là chuyển cả thông điệp bí mật và che thông điệp vào bộ mã hóa. Bên trong bộ mã hóa, một hoặc một số giao thức sẽ được triển khai để nhúng thông tin bí mật vào bìa tin nhắn. Loại giao thức sẽ phụ thuộc vào thông tin bạn đang cố gắng nhúng và bạn đang nhúng thông tin đó vào. Ví dụ, bạn sẽ sử dụng một giao thức hình ảnh để nhúng thông tin vào bên trong hình ảnh.

Một khóa thường cần thiết trong quá trình nhúng. Điều này có thể ở dạng công cộng hoặc riêng tư khóa để bạn có thể mã hóa thông điệp bí mật bằng khóa cá nhân của mình và người nhận có thể giải mã nó bằng khóa công khai của bạn. Khi nhúng thông tin theo cách này, bạn có thể giảm cơ hội của kẻ tấn công bên thứ ba nắm được đối tượng stego và giải mã nó để tìm ra thông tin bí mật.

Nói chung, quá trình nhúng sẽ chèn một dấu, M, vào một đối tượng, I. Một khóa, K, thường được tạo ra bởi một trình tạo số ngẫu nhiên được sử dụng trong quá trình nhúng và kết quả đối tượng được đánh dấu Ĩ, được tạo ra.

Sau khi đi qua bộ mã hóa, một đối tượng stego sẽ được tạo ra. Một đối tượng stego là đối tượng bìa ban đầu với thông tin bí mật được nhúng bên trong. Đối tượng này sẽ trông gần giống với đối tượng che vì nếu không, kẻ tấn công bên thứ ba có thể thấy thông tin.

Sau khi tạo ra đối tượng stego, nó sẽ được gửi đi qua một số kênh liên lạc, chẳng hạn như email, đến người nhận dự định để giải mã. Người nhận phải giải mã stego để họ xem thông tin bí mật. Quá trình giải mã đơn giản là ngược lại của quá trình mã hóa. Đó là việc trích xuất dữ liệu bí mật từ một đối tượng stego. Trong quá trình giải mã, đối tượng stego được đua vào hệ thống. Khóa công khai hoặc khóa riêng tư có thể giải mã khóa gốc được sử dụng bên trong quá trình mã hóa cũng cần thông tin bí mật có thể được giải mã. Tùy thuộc vào kỹ thuật mã hóa, đôi khi đối tượng bảo vệ ban đầu cũng cần thiết trong quá trình giải mã. Nếu không, ở đó có thể không có cách nào để trích xuất thông tin bí mật từ đối tượng stego.

Sau khi quá trình giải mã hoàn thành, thông tin bí mật được nhúng trong đối tượng stego (giấu tin) sau đó có thể được trích xuất và xem. Quá trình giải mã chung một lần nữa yêu cầu một khóa K, cái này thời gian cùng với một vật có khả năng dánh dấu là Ĩ'. Cũng bắt buộc phải có dấu M, là đang được kiểm tra hoặc đối tượng ban đầu I, và kết quả sẽ là dấu được truy xuất từ tân ngữ hoặc biểu thị khả năng M có mặt trong Ĩ'. Các loại mạnh mẽ khác nhau hệ thống đánh dấu sử dụng các đầu vào và đầu ra khác nhau [3].

#### Phương pháp thực hiện giấu tin bí mật (steganography)

Bằng cách tận dụng nhận thức của con người, có thể nhúng dữ liệu vào trong một tệp. Đối với ví dụ, với các tệp âm thanh che tần số xảy ra khi hai âm có tần số tương tự được phát cùng một lúc. Người nghe chỉ nghe thấy âm to hơn trong khi âm trầm hơn che mặt. Tương tự, che dấu thời gian xảy ra khi tín hiệu mức thấp xảy ra ngay lập tức trước hoặc sau một âm thanh mạnh hơn vì chúng ta phải mất thời gian để thích nghi với tần số mới của thính giác. Điều này cung cấp một điểm rõ ràng trong tệp để nhúng nhãn hiệu.

Tuy nhiên, nhiều định dạng được sử dụng cho phương tiện kỹ thuật số tận dụng khả năng nén các tiêu chuẩn như MPEG để giảm kích thước tệp bằng cách loại bỏ các phần không được nhận biết bởi người dùng. Do đó, nhãn hiệu nên được nhúng vào các phần của tệp để đảm bảo nó tồn tại trong quá trình nén.

Việc nhúng nhãn rõ ràng vào các phần quan trọng của tệp sẽ dẫn đến giảm chất lượng vì một số thông tin sẽ bị mất. Một kỹ thuật đơn giản liên quan đến việc nhúng nhãn hiệu vào các bit ít quan trọng nhất sẽ giảm thiểu sự biến dạng. Tuy nhiên nó cũng làm cho nó tương đối dễ dàng xác định vị trí và loại bỏ dấu hiệu. Một cải tiến là chỉ nhúng nhãn hiệu vào các bit quan trọng của dữ liệu được chọn ngẫu nhiên trong tệp.

Trong phần này, một số kỹ thuật che giấu thông tin khác nhau sẽ được thảo luận và đã kiểm tra. Các phương tiện liên quan khác nhau từ hình ảnh đến văn bản thuần túy. Trong khi một số kỹ thuật có thể được sử dụng để ẩn một loại thông tin nhất định, trong hầu hết các trường hợp, thông tin khác nhau có thể bị ẩn tùy thuộc vào hạn chế không gian.

Kỹ thuật tệp nhị phân: nếu chúng ta đang cố giấu một số thông tin bí mật bên trong tệp nhị phân, liệu bí mật thông tin là hình mờ bản quyền hoặc chỉ là văn bản bí mật đơn giản, chúng tôi đang phải đối mặt với vấn đề rằng bất kỳ thay đổi nào đối với tệp nhị phân đó sẽ khiến việc thực thi tệp đó thay đổi. Chỉ cần thêm một lệnh đơn lẻ sẽ làm cho việc thực thi khác nhau và do đó chương trình có thể không hoạt động bình thường và có thể làm hỏng hệ thống.

Bạn có thể thắc mắc tại sao mọi người lại muốn nhúng thông tin vào bên trong các tệp nhị phân, vì ở đó rất nhiều loại định dạng dữ liệu khác mà chúng tôi có thể nhúng thông tin vào. Lý do chính cho đây là mọi người muốn bảo vệ bản quyền của họ bên trong một chương trình nhị phân. Tất nhiên là có các phương tiện khác để bảo vệ bản quyền trong phần mềm, chẳng hạn như khóa nối tiếp, nhưng nếu bạn đã tìm kiếm trên Internet, các trình tạo chính cho các chương trình phổ biến được cung cấp rộng rãi và do đó sử dụng các khóa nối tiếp có thể không đủ để bảo vệ bản quyền của tệp nhị phân [3].

Một phương pháp để nhúng hình mờ trong tệp nhị phân hoạt động như sau. Đầu tiên, chúng ta hãy xem xét các dòng mã sau đã được trích xuất từ tệp nhị phân:

A = 2; B = 3; C = B + 3; D = B + C;

Ví dụ trên tương đương với:

B = 3; A = 2 C = B + 3; D = B + C;

B = 3; C = B + 3; A = 2; D = B + C;

B = 3; C = B + 3; D = B + C; A = 2;

Việc khởi tạo B, C, D phải được thực hiện theo cùng một thứ tự nhưng A thì có thể khai báo tùy ý.

Để nhúng hình mờ W = {w1, w­2, w3, w4…, wn} trong đó wi Є {0, 1}. Đầu tiên chúng ta chia mã nguồn thành n khối. Mỗi khối này sau đó được đại diện bởi wi và điều này chứa giá trị 0 hoặc 1. Nếu wi là 0, thì khối mã mà nó biểu diễn sẽ không thay đổi. Tuy nhiên, nếu wi là 1, thì bạn sẽ tìm kiếm hai câu lệnh bên trong khối và chuyển đổi chúng kết thúc.

Sử dụng phương pháp này, hình mờ có thể được nhúng bằng cách thực hiện các thay đổi đối với mã nhị phân điều đó không ảnh hưởng đến việc thực thi tệp. Để giải mã và trích xuất hình mờ, bạn sẽ cần có tệp nhị phân gốc. Bằng cách so sánh các tệp được đánh dấu và tệp gốc, sau đó bạn có thể phát hiện các công tắc câu lệnh và do đó trích xuất hình mờ được nhúng. Phương pháp này là rất đơn giản nhưng không có khả năng chống lại các cuộc tấn công. Nếu kẻ tấn công có nhiều phiên bản khác nhau của các tệp được đánh dấu thì anh ta có thể phát hiện hình mờ và do đó có thể xóa nó .

Tóm lại kỹ thuật Steganography phải đảm bảo:

* Lượng thông tin M cần giấu trong tệp video là đủ lớn.
* Sự thay đổi “video mang tin” so với video gốc là ít.
* Trước khi được giấu, thông điệp cần phải được mã hóa bằng thuật toán mã hóa nào.

#### 2.1.4.3. Các nghiên cứu liên quan

B.SUNEETHA đã đề xuất trong công trình của mình hệ thống dựa trên Cryptography và Steganography để ẩn dữ liệu trong video bằng cách mã hóa nó với mã ASCII và cung cấp một lớp bảo mật bổ sung [4]. Mật mã cung cấp sự riêng tư trong khi Steganography nhằm mục đích cung cấp bí mật. Kousik Dasgupta, JK Mandal và Paramartha Dutta đã đề xuất một kỹ thuật LSB dựa trên băm bảo mật cho video steganography sử dụng các tệp video bìa trong miền không gian để ẩn sự hiện diện của dữ liệu nhạy cảm bất kể định dạng của nó. Phân tích hiệu suất của kỹ thuật LSB dựa trên băm sau khi so sánh với kỹ thuật LSB là tốt hơn. A. Swathi, Tiến sĩ SAK Jilani đã đề xuất trong bài báo của mình sự thay thế LSB sử dụng phương trình đa thức được phát triển để ẩn thông tin trong các khung cụ thể của video và ở vị trí cụ thể của khung bằng cách thay thế LSB sử dụng phương trình đa thức. Tại đây thông tin sẽ được nhúng dựa trên chìa khóa. Key ở dạng phương trình đa thức với các hệ số khác nhau. Bằng cách sử dụng điều này, khả năng nhúng các bit vào ảnh bìa có thể được tăng lên [5].

Mritha Ramalingam đã đề xuất một phương pháp LSB bảo mật hơn, trong đó tệp video được sử dụng làm phương tiện lưu trữ để ẩn tin nhắn bí mật mà không ảnh hưởng đến cấu trúc tệp và nội dung của tệp video. Vì chất lượng video suy giảm dẫn đến thay đổi có thể nhìn thấy trong video có thể dẫn đến việc không đạt được các mục tiêu của Steganography [6]. Ashawq T. Hashim và cộng sự đã đề xuất một kỹ thuật Mã hóa lai và Mật mã trong đó có hai phương pháp ẩn được sử dụng, phương pháp đầu tiên là Bit ít có ý nghĩa nhất (LSB) và phương pháp thứ hai là Biến đổi Wavelet Haar (HWT). Công việc này dựa trên sự kết hợp giữa kỹ thuật ghi chép và mật mã để tăng mức độ bảo mật và làm cho hệ thống phức tạp hơn bị đánh bại bởi những kẻ tấn công.

R. Shanthakumari và Dr.S. Malliga trong công việc được đề xuất của họ đã tuyên bố một thuật toán được xem lại đối sánh LSB (LSBMR) chọn nhúng các vùng theo kích thước của thông điệp bí mật và sự khác biệt giữa hai pixel liên tiếp trong ảnh bìa [7].

### Kỹ thuật đánh dấu thủy vân (watermark) và các nghiên cứu liên quan

#### Kỹ thuật thủy vân

Watermarking mẫu bit được chèn vào tệp bìa xác định thông tin bản quyền của tệp (tác giả, quyền, v.v.). Tên này đến từ các hình mờ mờ có thể nhìn thấy được in trên văn phòng phẩm xác định nhà sản xuất văn phòng phẩm. Mục đích của hình mờ kỹ thuật số là cung cấp bản quyền bảo vệ tài sản trí tuệ ở định dạng kỹ thuật số [8].

Trong cả hai kỹ thuật watermark (hình mờ mỏng manh và hình mờ nửa mảnh), có thể áp dụng tính toàn vẹn của thưbằng tổng kiểm tra cho bit quan trọng nhất và nhúng nó thành các pixel được chọn một cách ngẫu nhiên bằng cách sử dụng kỹ thuật LSB. Ngoài ra, hình ảnh có thể được phân vùng thành các khối biến kích thước và mỗi khối được xác định duy nhất theo kích thước, do đó khi kích thước khối thay đổi, điều này sẽ rõ ràng sau đó hình mờ cũng thay đổi [9]. Vì watermarking không cung cấp một con đường cho một sơ đồ xử lý hình ảnh bất kỳ những thay đổi đối với hình ảnh có thể không quan trọng để phát hiện, như khi tính toàn vẹn của thông điệp cần được giám sát. Vì thế watermarking sẽ không đáng tin cậy để đảm bảo bí mật hoặc trong tính toàn vẹn của thông điệp bí mật [10].

Watermarking là một kỹ thuật mà thông qua đó thông tin được truyền đi mà không làm giảm chất lượng củatín hiệu ban đầu. Kỹ thuật này bao gồm hai khối: khối nhúng và chiết xuất. Hệ thống có một khóa nhúng như trong trường hợp in mật mã. Mấu chốt làđược sử dụng để tăng cường bảo mật, điều này không cho phép bất kỳ người dùng trái phép để thao tác hoặc trích xuất dữ liệu. Các đối tượng nhúng được gọi là hình mờ, hình mờ phương tiện nhúng được gọi là tín hiệu gốc hoặc vỏ bọc đối tượng và đối tượng sửa đổi được gọi là tín hiệu nhúng hoặc dữ liệu hình mờ [9].

Khối nhúng, được hiển thị trong hình 2.2 bao gồm hình mờ, tín hiệu gốc (hoặc đối tượng che) và khóa watermark làm đầu vào (tạo tín hiệu nhúng hoặc dữ liệu hình mờ) [11]. Trong khi, đầu vào cho khối trích xuất là đối tượng được nhúng, khóa và đôi khi là watermark như minh họa trong hình 2.3 [11].

Kỹ thuật tạo hình mờ không sử dụng hình mờtrong quá trình khai thác được gọi là "đánh dấu nước mù”. "Watermarking mù vượt trội hơn các watermarking khác liên quan đến hình mờ để trích xuất dưới dạng tín hiệu hình mờ và chìa khóa đủ để tìm ra bí mật được nhúng thông tin [12].

Kỹ thuật đánh dấu nước đã phát triển đáng kể từ nguồn gốc của nó [12]. Do sự phát triển của công nghệ, phương tiện truyền dẫn đã được thay đổi. Watermarking là được sử dụng trong các phương tiện kỹ thuật số như hình ảnh và âm thanh. Âm thanh watermarking khá khó khăn hơn watermarking hình ảnh do tính năng động tối cao của hệ thống thính giác của con người (HAS) qua hệ thống thị giác của con người (HVS) [12]. Các ứng dụng của Watermarking được sử dụng để bảo vệ Quyền sở hữu và bằng chứng về quyền sở hữu, Xác thực và phát hiện giả mạo, In ngón tay, Giám sát phát sóng, Kiểm soát sao chép và kiểm soát truy cập, hãng thông tin, ứng dụng y tế và Giám sát lưu lượng hàng không [12].

##### Hệ thống thủy vân riêng

Hệ thống đánh dấu riêng có thể được chia thành nhiều loại khác nhau nhưng tất cả đều yêu cầu bản gốc hình ảnh. Hệ thống loại I sử dụng I để giúp định vị dấu trong Ĩ 'và xuất ra dấu. Hệ thống loại II cũng yêu cầu M và chỉ cần đưa ra câu trả lời có hoặc không cho câu hỏi "hiện Ĩ" chứa dấu M”.

Hệ thống đánh dấu bán riêng hoạt động giống như Loại II ngoại trừ chúng không yêu cầu hình ảnh gốc và chỉ cần trả lời câu hỏi tương tự. Hệ thống đánh dấu riêng tiết lộ ít thông tin và yêu cầu khóa bí mật để phát hiện dấu hiệu. Nhiều hệ thống hiện tại thuộc loại này và chúng thường được sử dụng để chứng minh quyền sở hữu vật chất tại tòa án.

##### Hệ thống đánh dấu công khai (đánh dấu mù)

Hệ thống đánh dấu công cộng không yêu cầu I hoặc M nhưng trích xuất n bit từ Ĩ' đại diện cho đánh dấu. Hệ thống đánh dấu công khai có nhiều ứng dụng hơn và các thuật toán thường có thể được sử dụng trong các hệ thống riêng.

##### Hệ thống đánh dấu bất đối xứng (đánh dấu khóa công khai)

Hệ thống đánh dấu không đối xứng cho phép bất kỳ người dùng nào đọc được phần đánh dấu nhưng ngăn họ loại bỏ nó. Steganography có thể được chia thành hai loại đó là Fragile và Robust. Dưới đây là phần mô tả định nghĩa của hai loại này:

* *Mỏng manh (fragile)*

Mật mã mỏng manh liên quan đến việc nhúng thông tin vào một tệp sẽ bị hủy nếu tệp đã được sửa đổi. Phương pháp này không phù hợp để ghi lại chủ sở hữu bản quyền của tệp vì nó có thể rất dễ bị xóa nhưng rất hữu ích trong các tình huống mà điều quan trong là phải chứng mình rằng tệp không có bị giả mạo, chẳng hạn như sử dụng một tệp làm bằng chứng trước tòa án pháp luật. Các kỹ thuật nhúng mật mã mỏng manh có xu hướng dễ thực hiện hơn các phương pháp nhúng mật mã mạnh mẽ.

* *Mạnh mẽ (Robust)*

Đánh dấu mạnh mẽ nhawfwm mục đích nhúng thông tin vào một tệp mà không thể dễ dàng bị phá hủy. Mặc dù không có dấu hiệu nào la thực sự không thể phá hủy nhưng một hệ thống có thể được coi là mạnh mẽ nếu lượng những thay đổi cân thiết để xóa dấu sẽ làm cho tệp vô dụng. Do đó nên được ẩn trong một phần của tệp nơi có thể dễ dàng nhận ra việc xóa nó.

* + Có hai kiểu dánh dấu mạnh mẽ chính:

Lấy dấu vân tay liên quan đến việc ẩn một số nhận dạng duy nhất cho khách đã mua tệp ban đầu và do đó được phép sử dụng nó. Nên tệp được tìm thấy thuộc quyền sở hữu của người khác, chủ sỡ hữu bản quyền có thể sử dụng dấu vân tay để xác định khách hàng nào vi phạm thỏa thuận cấp phép bằng cách phân phối một bản sao của tệp.

Hình mờ (watermark) xác định chủ sở hữu bản quyền của tệp, không phải khách hàng. Trong khi dấu vân tay được sử dụng để xác định những người vi phạm thỏa thuận cấp phép hình mờ giúp truy tố những người có một bản sao bất hợp pháp. Tốt nhất nên lấy dâu vân tay được sử dụng nhưng để sản xuất hàng loạt đĩa CD, DVD… việc cung cấp cho mỗi đĩa một dấu vâng tay riêng biệt thì khó khăn. Nên hình mờ được ẩn để ngăn chặn việc phát hiện và xóa chúng (hình mờ ẩn không được nhận thấy). Tuy nhiên điuề này không phải luôn luôn như vậy HÌnh mờ có thể nhìn thấy được sử dụng va thường có dạng mô hình trực quan phủ trên hình ảnh. Việc sử dụng hiển thị hình mờ tương tự như việc sử đụng hình mờ ở các dạng không phải kỹ thuật số (chẳng hạn như hình mờ trên tiền của Anh).

### Video và cấu trúc định dạng một số tệp video

Hầu hết các định dạng đa phương tiện đều có hai khía cạnh. Định dạng vùng chứa xác định cấu trúc của tệp. Nó chứa một hoặc nhiều luồng, cùng với thông tin về cách phát chúng. Nó cũng có thể chứa siêu dữ liệu như tiêu đề, chủ bản quyền và ngày tạo. Mã hóa dữ liệu video được xác định bằng codec. Lưu trữ rõ ràng từng pixel của mọi khung hình của một bộ phim độ nét cao sẽ yêu cầu dung lượng lưu trữ khổng lồ. Các nhà phát triển phần mềm đã nghĩ ra nhiều kế hoạch để nén video mà không bị suy giảm rõ ràng, do đó, có rất nhiều codec cạnh tranh. Hầu hết các vùng chứa, đều hỗ trợ lựa chọn codec.

#### Khung và cấu trúc khung trong video

Video bao gồm một loạt các ảnh bitmap trực giao hiển thị trong liên kết nhanh với tốc độ không đổi. Trong cấu trúc của video những ảnh này được gọi là khung hình. Chúng ta đo tốc độ khung hình được hiển thị trong mỗi giây (FPS).

Vì mỗi khung hình là một ảnh kỹ thuật số trực giao bitmap bao gồm một raster các điểm ảnh (pixel). Nếu nó có chiều rộng W pixel và chiều cao H pixel ta nói rằng kích thước khung hình là W x H pixels. Pixels chỉ có một thuộc tính màu sắc của chúng. Màu sắc của một điểm ảnh được biểu diễn bởi một giá trị cố định các bit. Các bit hơn các biến thể tinh tế của màu sắc là có thể được sao chép. Đây được gọi là độ sâu màu (CD) của video.

Ví dụ video có thể có thời gian (T) 1 giờ (3600 giây), kích thước khung hình 640 x 480 (R x C) ở độ sâu màu 24bit và tỷ lệ khung hình 25 fps. Video ví dụ này có các thuộc tính sau:

* Pixel mỗi khung hình = 640 \* 480 = 307.200
* Bit trên mỗi khung hình = 307.200 \* 24 = 7.372.800 = 7, 37 Mbits
* Tỷ lệ bit (BR) = 7.37 \* 25 = 184, 25 Mbits / sec
* Kích thước video (VS) = 184 Mbits / sec \* 3600 giây = 662.400 Mbits

= 82.800 MB = 82, 8 GB

Các đặc tính quan trọng nhất là tốc độ bit và kích thước video. Các công thức liên quan giữa hai thuộc tính đó với tất cả các thuộc tính khác là:

BR = W \* H \* CD \* FPS

VS = BR \* T = W \* H \* CD \* FPS \* T

(Đơn vị là: BR theo bit/s, W và H theo điểm ảnh, CD bằng bit, VS theo bit, T

theo giây)

Trong khi một số công thức thứ cấp là:

* pixels\_per\_khung hình = W \* H
* pixels\_per\_second = W \* H \* FPS
* bits\_per\_khung hình = W \* H \* CD

#### Một số loại định dạng video phổ biến

* *Định dạng AVI (Audio Video Interle)*

Định dạng AVI (Audio Video Interle) là một định dạng số đa phương tiện do  
Microsoft giới thiệu vào tháng khoảng 11/1992 như một chuẩn video dành cho  
Windows. Tệp AVI có thể chứa cả dữ liệu âm thanh và video trong một tệp, cho phép đồng bộ với phát lại audio – video. Đặc điểm của tệp AVI là dạng video không nén, chính vì vậy hình ảnh của video dạng này khá đẹp và sắc nét, đây là đặc tính ưu điểm và đồng thời cũng là nhược điểm của định dạng này, vì không hình ảnh và âm thanh của nó được nén nên dung lượng của một tệp AVI thường khá lớn (một tệp video avi khoảng 60 phút sẽ có dung lượng khoảng trên dưới 10Gb)

* *Định dạng FLV (Flash video)*

Tệp FLV là một dạng file nén từ các file video khác để tải lên trang web với dung lượng nhỏ, tuy nhiên chất lượng của hình ảnh không bằng được file gốc (MP4, WAV…). Tệp FLV được lựa chọn cho việc nhúng video trong web, đây là định dạng hay được sử dụng bởi ứng dụng trên web như: Youtube, Google Video, Yahoo!Video,…

* *H.264/MPEG – 4 Part 10 hay AVC (Advanced Video Coding)*

Đây là một chuẩn mã hóa/giải mã video và định dạng tệp video đang được sử dụng rộng rãi nhất hiện này vì khả năng ghi, nén và chia sẽ video phân giải cao. Tệp này có dung lượng thấp nhưng mang lại chất lượng rất cao.

* *MPEG-4 Part 14 hoặc MP4*

Là định dạng thường được sử dụng để lưu trữ video và âm thanh, nhưng cũng có thể được sử dụng để lưu trữ dữ liệu khác như phụ đề và hình ảnh. MP4 cho phép truyền tải trên Internet.

* *MKV (hay còn gọi là Matroska)*

Trái ngược với nhiều định dạng video khác, tệp tin MKV không phải là một định dạng nén âm thanh hoặc video. Tệp MKV là một tệp thực sự chứa đa phương tiện, nó có thể kết hợp âm thanh, video và phụ đề vào một tập tin duy nhất ngay cả khi chúng sử dụng mã khác nhau. Bạn có thể có một tập MKV sử dụng video VP8 với âm thanh Vorbis, hoặc phổ biến hơn sử dụng H.264 cho video và một cái gì đó giống MP3 hay ACC cho âm thanh.

### Hệ mật RSA

[RSA](http://people.csail.mit.edu/rivest/Rsapaper.pdf) là một hệ mã hóa bất đối xứng được phát triển bởi Ron Rivest, Adi Shamir và Leonard Adleman (tên của nó cũng chính là tên viết tắt của 3 tác giả này) và được sử dụng rộng rãi trong công tác mã hoá và công nghệ chữ ký điện tử. Trong hệ mã hóa này, public key có thể chia sẻ công khai cho tất cả mọi người. Hoạt động của RSA dựa trên 4 bước chính: sinh khóa, chia sẻ key, mã hóa và giải mã.

#### Hoạt động

Thuật toán RSA bao gồm bốn bước: [quan trọng](https://en.wikipedia.org/wiki/Key_(cryptography)) thế hệ, phân phối chìa khóa, mã hóa, giải mã và.

Một nguyên tắc cơ bản đằng sau RSA là nhận xét rằng thực tế là tìm ra ba số nguyên dương rất lớn *e*, *d* và *n* , sao cho với [lũy thừa mô-đun](https://en.wikipedia.org/wiki/Modular_exponentiation) cho tất cả các số nguyên *m* (với 0 ≤ *m* < *n* ):{\ displaystyle (m ^ {e}) ^ {d} \ equiv m {\ pmod {n}}}

(me)d≡ m (mod n)

và việc biết *e* và *n*, hoặc thậm chí *m*, có thể cực kỳ khó tìm ra *d*. Thanh [ba](https://en.wikipedia.org/wiki/Triple_bar) (≡) ở đây biểu thị [sự đồng dạng mô-đun](https://en.wikipedia.org/wiki/Modular_arithmetic) .

Ngoài ra, đối với một số phép toán, thuận tiện là thứ tự của hai lũy thừa có thể được thay đổi và quan hệ này cũng ngụ ý:{\ displaystyle (m ^ {d}) ^ {e} \ equiv m {\ pmod {n}}}

(md)e≡ m (mod n)

RSA liên quan đến *khóa công khai* và [*khóa riêng tư*](https://en.wikipedia.org/wiki/Private_key) . Khóa công khai có thể được mọi người biết đến và nó được sử dụng để mã hóa tin nhắn. Mục đích là các thư được mã hóa bằng khóa công khai chỉ có thể được giải mã trong một khoảng thời gian hợp lý bằng cách sử dụng khóa riêng. Khóa công khai được biểu diễn bởi các số nguyên *n* và *e*; và, khóa riêng tư, bởi số nguyên *d* (mặc dù *n* cũng được sử dụng trong quá trình giải mã, vì vậy nó cũng có thể được coi là một phần của khóa riêng). *m* đại diện cho thông điệp (được chuẩn bị trước đó với một kỹ thuật nhất định được giải thích bên dưới).

#### Tạo khóa

Gồm 2 khóa:

* Khóa bí mật.
* Khóa công khai.

#### Mã hóa

Sau khi Bob lấy được khóa công khai của Alice, anh ta có thể gửi một tin nhắn *M* cho Alice.

Để làm điều đó, trước tiên anh ta biến *M* (nói đúng ra là bản rõ không đệm) thành một số nguyên *m* (nói đúng ra là bản rõ có đệm), sao cho 0 ≤ *m* < *n* bằng cách sử dụng một giao thức có thể đảo ngược được thỏa thuận được gọi là [đệm kế hoạch](https://en.wikipedia.org/wiki/RSA_(cryptosystem)#Padding_schemes) . Sau đó, anh ta tính toán bản mã *c*, sử dụng khóa công khai *e* của Alice, tương ứng với

{\ displaystyle m ^ {e} \ equiv c {\ pmod {n}}}me≡ c (mod n)

Điều này có thể được thực hiện một cách hợp lý nhanh chóng, ngay cả đối với những số rất lớn, bằng cách sử dụng [lũy thừa mô-đun](https://en.wikipedia.org/wiki/Modular_exponentiation) . Bob sau đó truyền *c* cho Alice.

#### Giải mã

Alice có thể khôi phục *m* từ *c* bằng cách sử dụng số mũ *d* khóa riêng của mình bằng tính toán{\ displaystyle c ^ {d} \ equiv (m ^ {e}) ^ {d} \ equiv m {\ pmod {n}}}

{\ displaystyle m ^ {e} \ equiv c {\ pmod {n}}}cd≡ (me)d≡ m (mod n)

Với *m*, Alice có thể khôi phục thông điệp ban đầu *M* bằng cách đảo ngược sơ đồ đệm.

## Các nghiên cứu liên quan

### 2.2.1. Truyền dữ liệu an toàn dựa trên video steganography (SLSB)

Trong Truyền dữ liệu an toàn Kỹ thuật Steganography hoạt động dựa trên kỹ thuật nén được đánh giá sao cho dữ liệu được được nhúng trong các pixel thành phần dọc và ngang. Để đánh giá khung hình, có ba loại hình ảnh (hoặc khung hình) được sử dụng trong nén video: I-frame, P-frame và B-frame được xác định dựa trên lượng dữ liệu nén. Chúng có những đặc điểm khác nhau: Các khung hình (được mã hóa nội bộ) không yêu cầu các khung video khác giải mã nhưng ít có khả năng nén nhất. Khung P- (Dự đoán) sử dụng dữ liệu từ các khung hình trước đó để giải nén và có khả năng nén nhiều hơn các khung hình I. Khung B- (Dự đoán Bi) sử dụng cả trước và chuyển tiếp khung tham chiếu dữ liệu để có được lượng dữ liệu nén cao hơn.

#### 2.2.1.1 Thuật toán mã hóa dựa trên kỹ thuật truyền bảo mật

* Bước 1: Lấy đầu vào tệp video bìa hoặc luồng.
* Bước 2: Đọc thông tin yêu cầu của file video bìa.
* Bước 3: Chia video thành các khung.
* Bước 4: Nén khung nơi dữ liệu sẽ được chèn bằng bất kỳ kỹ thuật nén nào (DCT).
* Bước 5: Ẩn dữ liệu bằng thuật toán LSB.

#### 2.2.1.2. Thuật toán giải mã dựa trên kỹ thuật truyền bảo mật

* Bước 1: Nhập tập tin hoặc luồng video stego.
* Bước 2: Đọc thông tin cần thiết từ video stego.
* Bước 3: Chia video thành các khung.
* Bước 4: Sử dụng vector chuyển động, chọn khung nơi ẩn dữ liệu.
* Bước 5: Dữ liệu được trích xuất từ ​​các LSB của khung đã xác định.

Kỹ thuật Truyền Dữ liệu Bảo mật đã cho cung cấp dung lượng cao và không thể nhận thấy, cho con người nhìn thấy bí mật ẩn giấu thông tin. Bằng cách nhúng dữ liệu vào các hình ảnh chuyển động, chất lượng của video được tăng lên. Video nén được sử dụng cho truyền dữ liệu vì nó có thể chứa một lượng lớn dữ liệu. Thiếu tính bảo mật.

### 2.2.2. Kỹ thuật bit chủ yếu ít nhất Hash cho video steganography (HLSB)

Kỹ thuật Hash dựa trên bit ít quan trọng nhất (HLSB) cho Video Steganography đã được sử dụng, ở đây dữ liệu bí mật được ẩn trong LSB của các khung bìa. Trong kỹ thuật này, tám bit dữ liệu bí mật được chia theo định dạng 3, 3, 2 và được nhúng vào pixel RGB giá trị của khung video. Một hàm băm được sử dụng để chọn vị trí chèn trong các bit LSB. Ví dụ: giá trị pixel RGB của khung bìa dưới đây:

**R: 10110111** **G: 10010100** **B: 11001001**

và một byte thông báo sẽ được chèn vào LSB dưới dạng: **10001001**

LSB là bit thấp nhất trong một chuỗi các số nhị phân, vì vậy trong trường hợp này đối với R, nó sẽ là 1, 0 đối với G và 1 đối với B. Kỹ thuật được đề xuất là được áp dụng trong bốn LSB thấp nhất trong mỗi giá trị pixel. Vì vậy, LSB cho các giá trị RGB ở trên là:

**R: 0111** **G: 0100** **B: 1001**

Thông báo được nhúng trong các nhóm 3, 3 và 2 ở các vị trí RGB LSB tương ứng. Các vị trí có được từ hàm băm hàm cho trong phương trình k = p% n. Giá trị của n số bit LSB cho kịch bản hiện tại là 4. Sử dụng hàm băm cho phép vị trí của chèn k được trả về cho một lần lặp cụ thể là

k = 1,2,3 đối với R. k = 4,1,2 đối với G k = 3,4 đối với B.

Xem xét các vị trí chèn ở trên, các bit từ thông báo được chèn vào bốn vị trí LSB và kết quả là pixel RGB giá trị như được đưa ra bên dưới.

**R: 10111001** **G: 10011000 B: 11001001**

Do đó, tất cả tám bit của thông báo được nhúng trong ba byte và số bit thực sự thay đổi là năm trên hai mươi bốn bit.Hơn nữa, năm bit này được phân phối ngẫu nhiên trong đó làm tăng tính mạnh mẽ của chương trình.

Để giải mã tin nhắn, người dùng hợp lệ làm theo bước ngược lại. Vì hàm băm được biết đến với mục đích của người dùng, nó sẽ tính toán k giá trị để lấy vị trí chèn. Lấy cùng một giá trị RGB được nhúng như trên,

**R: 10111001** **G: 10011000** **B: 11001001**

Hàm băm sẽ trả về k giá trị sau cho lần lặp cụ thể này.

k = 1,2,3 đối với R. k = 4,1,2 đối với G k = 3,4 đối với B

sử dụng k giá trị này đại diện cho bốn vị trí LSB, dữ liệu của thông điệp bí mật được tìm thấy như bên dưới,

**10001001**

Giống như dữ liệu của thông điệp bí mật như đã xét ở trên.

Luồng trình tự của thuật toán được thể hiện trong hình 2.4

Kỹ thuật HLSB được áp dụng cho các tệp AVI, tuy nhiên nó có thể hoạt động với bất kỳ định dạng nào khác với một số thay đổi. Đối với video nén định dạng tệp như MPEG video cần được giải nén thì kỹ thuật này có thể được áp dụng cho video không nén. Trong khi cho các tệp Flash Video FLV có thể áp dụng kỹ thuật này mà không cần sửa đổi. Nó kém an toàn hơn vì dữ liệu được ẩn trực tiếp trong video.

### 2.2.3. Video Steganography bằng LSB thay thế sử dụng các phương trình đa thức khác nhau (LSB Poly)

Kỹ thuật chèn bit quan trọng nhất (LSB) hoạt động trên bit LSB của tệp phương tiện để ẩn bit thông tin. Trong kỹ thuật này, một lược đồ ẩn dữ liệu sẽ được phát triển để ẩn thông tin trong các khung cụ thể của video và ở vị trí cụ thể của khung bằng cách thay thế LSB bằng phương trình đa thức. Đầu tiên đọc tín hiệu video gốc và văn bản, sau đó nhúng văn bản vào tín hiệu video cho chuyển đổi dữ liệu văn bản sang định dạng nhị phân. Chuyển đổi nhị phân được thực hiện bằng cách lấy giá trị ASCII của ký tự và chuyển đổi các giá trị ASCII đó sang định dạng nhị phân. Biểu diễn nhị phân của các mẫu tín hiệu bao phủ được chèn vào biểu diễn nhị phân của văn bản. Các bit LSB của tín hiệu video được thay thế bằng các bit nhị phân của dữ liệu và tín hiệu được mã hóa này được gọi là tín hiệu stego đã sẵn sàng cho truyền qua internet. Thông báo mà chúng tôi muốn ẩn được chuyển đổi thành ASCII và sau đó được chuyển đổi thành tệp nhị phân của nó biểu diễn với mỗi từ bao gồm 8 bit. Các bit này được thay thế trong các bit quan trọng nhất của biểu diễn nhị phân của mỗi hình ảnh mẫu. Ở đây, các phương trình đa thức được sử dụng để tìm vị trí chèn bit dữ liệu trong tệp video.

#### 2.2.2.1. Mã hóa để thay thế bằng các phương trình đa thức khác nhau

Đầu tiên, lấy video gốc mà chúng ta phải nhúng hình ảnh, sau đó chuyển đổi tệp video thành số khung hình, xem xét từng khung hình như một hình ảnh. Tại đây đặt giá trị bộ đếm thành các khung. Sau đó chuyển đổi dữ liệu văn bản sang định dạng nhị phân. Chuyển đổi nhị phân được thực hiện bằng cách lấy giá trị ASCII của mỗi ký tự và chuyển đổi các giá trị ASCII đó sang định dạng nhị phân. Đặt giá trị bộ đếm bằng độ dài của thông điệp nhị phân, để vòng lặp lặp lại nhiều lần. Bit LSB của pixel hình ảnh được thay thế bằng dữ liệu nhị phân. Mã hóa này hình ảnh được gọi là video stego đã sẵn sàng để truyền qua internet.

#### 2.2.2.2. Giải mã để thay thế bằng các phương trình đa thức khác nhau

Stego Frame

Secret Message

Stegoganography using HLSB

Cover video file

Collect information of the video

Cover frames

Frames

Other frames

Secret Message

Hình 2. 4. Sơ đồ giấu tin sử dụng thuật toán HLSB.

Đầu tiên hãy lấy hình ảnh được mã hóa LSB. Đặt bộ đếm theo độ dài của dữ liệu nhị phân. Sau đó trích xuất dữ liệu nhị phân từ LSB được mã hóa hình ảnh bằng cách trích xuất các bit LSB của các pixel hình ảnh. Để tạo tệp văn bản từ nhóm dữ liệu nhị phân tất cả các bit nhị phân.

Frame

Secret Message

Destegoganography using HLSB

Stego video file

Collect information of the video

Stego frames

Stego Frames

Other frames

video

Hình 2. 5. Sơ đồ trích xuất sử dụng thuật toán HLSB

### 2.2.3. Stego Machine- Video Steganography sử dụng thuật toán LSB sửa đổi (MLSB)

Máy stego được phát triển để ẩn dữ liệu chứa văn bản trong tệp video và lấy thông tin ẩn. Điều này có thể được thực hiện bởi nhúng tệp văn bản vào tệp video theo cách sao cho video không bị mất chức năng bằng Bit ít quan trọng nhất (LSB) phương pháp sửa đổi. Thông báo được ẩn bên trong tệp của nhà cung cấp dịch vụ được mã hóa cùng với một khóa để cung cấp tính mạnh mẽ cho.

Thuật toán bit ít quan trọng nhất (LSB) được sử dụng để che giấu dữ liệu trong tệp video. Ưu điểm chính của phương pháp mã hóa LSB là tốc độ bit kênh hình mờ rất cao và độ phức tạp tính toán thấp. Tính mạnh mẽ của mã hóa LSB nhúng hình mờ, tăng khi tăng độ sâu LSB. Trong phương pháp này, các sửa đổi được thực hiện đối với các bit ít quan trọng nhất của tệp sóng mang pixel riêng lẻ, do đó mã hóa dữ liệu ẩn. Ở đây mỗi pixel có chỗ cho 3bit thông tin bí mật, một trong mỗi giá trị RGB. Sử dụng hình ảnh 24bit, có thể ẩn ba bit dữ liệu trong giá trị màu của mỗi pixel bằng hình ảnh 1024x768 pixel; nó cũng vậy có thể ẩn lên đến 2.359.296 bit. Mắt người không thể dễ dàng phân biệt màu 21bit và màu 24bit. Như một ví dụ đơn giản về Thay thế LSB, hãy tưởng tượng "ẩn" ký tự 'A' trên tám byte sau của tệp sóng mang:

(00100111 11101001 11001000)

(00100111 11001000 11101001)

(11001000 00100111 11101001)

Chữ cái 'A' được biểu diễn ở định dạng ASCII là chuỗi nhị phân 10000011. Tám bit này có thể được "ghi" vào LSB của mỗi tám byte mang như sau (các LSB được in nghiêng và in đậm):

(0010011 ***1*** 1110100 ***0*** 1100100 ***0*** )

(0010011 ***0*** 1100100 ***0*** 1110100 ***0*** )

(1100100 ***1*** 0.010.011 ***1*** 11.101.001).

Với sự thay đổi nhỏ như vậy về màu sắc của hình ảnh video, mắt người sẽ rất khó phân biệt do đó độ bền cao cho hệ thống.

### 2.2.4. Video Steganography sử dụng thuật toán xem lại đối sánh LSB (LSBMR)

Thuật toán LSB Matching Revisited (LSBMR) cho Video Steganography chọn các vùng nhúng theo kích thước bí mật thông báo và sự khác biệt giữa hai pixel liên tiếp trong ảnh bìa [7].

#### Thuật toán mã hóa trong nội dung ẩn video LSBMR

* Bước 1: Chia Video thành các Khung, tệp video bìa được phân tách thành số khung trong đó thông điệp bí mật sẽ là ẩn. Phím chia sẻ được sử dụng để chọn khung ẩn tin nhắn.
* Bước 2: Tính khóa bằng Thuật toán Diffie Hellman. Phương thức trao đổi khóa Diffie-Hellman cho phép hai bên có không có kiến ​​thức trước về nhau để cùng thiết lập khóa bí mật dùng chung trên một kênh liên lạc an toàn.
* Bước 3: Trong Nhúng văn bản, sơ đồ trước tiên khởi tạo một số tham số, được sử dụng để xử lý trước dữ liệu tiếp theo và lựa chọn vùng và sau đó ước tính dung lượng của các vùng đã chọn đó. Nếu các vùng đủ lớn để che giấu bí mật đã cho, sau đó ẩn dữ liệu được thực hiện trên các vùng đã chọn. Cuối cùng, nó thực hiện một số xử lý bài đăng để có được hình ảnh stego
* Bước 4: Sau khi ẩn dữ liệu, ảnh thu được được chia thành các khối BZ \* BZ không chồng lên nhau. Các khối sau đó được xoay bởi một số độ ngẫu nhiên dựa trên khóa. Quá trình này rất giống với Bước 1 ngoại trừ độ ngẫu nhiên là ngược lại. Sau đó chúng ta nhúng hai tham số vào một vùng đặt trước chưa được sử dụng để ẩn dữ liệu.

#### Thuật toán giải mã trong nội dung video LSBMR

* Bước 1: Để trích xuất dữ liệu, trước tiên hãy trích xuất thông tin bên, tức là kích thước khối BZ và ngưỡng t từ ảnh stego. Sau đó làm giống hệt như Bước 1 trong nhúng dữ liệu.
* Bước 2: Hình ảnh stego được chia thành các khối Bz \* Bz và các khối sau đó được xoay theo độ ngẫu nhiên dựa trên khóa bí mật key1. Hình ảnh kết quả được sắp xếp lại dưới dạng vector hàng V. Cuối cùng, đơn vị nhúng thu được bằng cách chia V thành không chồng chéo khối có hai pixel liên tiếp.
* Bước 3: Di chuyển các đơn vị nhúng có sự khác biệt tuyệt đối lớn hơn hoặc bằng ngưỡng T theo tỷ lệ giả đặt hàng dựa trên khóa khóa bí mật2.
* Thuật toán LSBMR do tỷ lệ thay thế thấp, giá trị MSE thấp làm cho nó an toàn khi so sánh với thuật toán LSB. Dự kiến ​​rằng ý tưởng có thể được mở rộng bằng cách nhúng văn bản vào các khung khác nhau của cùng một video.

# CHƯƠNG III: PHƯƠNG PHÁP ĐỀ XUẤT

Kỹ thuật giấu tin (còn gọi là bảo mật thông tin được giấu) trong video yêu cầu cần thiết đối với sự phát triển của kỹ thuật mật mã. Trong nghiên đồ án tập trung tìm hiểu về kỹ thuật giấu tin trong video. Từ thuật toán giấu tin đã được công bố và thuật toán cải tiến của nó trước đây, đồ án trình bày một thuật toán giấu tin mới có hiệu quả cao hơn.

## Yêu cầu bài toán

Ngày nay công nghệ phát triển, việc chia sẻ/ chỉnh sửa video một cách phổ biến qua Mạng xã hội như facebook, zalo… Ai cũng có thể chỉnh sửa video mà không cần có nhiều kiến thức công nghệ thông tin. Vậy nên việc định danh video là quan trọng, nếu muốn bảo vệ quyền sở hữu cho người sáng tạo video. Nếu trong trường hợp tranh chấp thì định danh này sẽ đóng vai trò quan trọng giúp bảo vệ người sáng tạo ra video.

Yêu cầu: đoạn định danh này (thông điệp bí mật) được nhúng vào trong video và sau quá trình chỉnh sửa cơ bản thì nó không bị loại bỏ. Và khi cần thiết có thể trích xuất ra định danh này. Quá trình nhúng này đảm bảo không làm thay đổi chất lượng của video.

## Phương pháp đề xuất

Đã có nhiều thuật toán giấu tin vào video được giới thiệu nhưng phổ biến nhất và được ứng dụng rộng rãi nhất là các thuật toán chèn các thông tin ẩn vào bit có ý nghĩa thấp nhất (Least Significant Bit – LSB) trong phần dữ liệu khung hình và audio của video.

Về cơ bản, trong xử lý ảnh mỗi pixel nói chung được lưu dưới dạng 8 bit hay 24 bit. Với biểu diễn 24 bit, mỗi pixel trải trên 3 byte, mỗi byte ứng với các màu đỏ, xanh da trời và xanh lá cây (RGB). Các màu là sự kết hợp của 3 màu trên. Mỗi byte có giá trị từ 0 - 255 ứng với cường độ màu. Màu tối nhất có giá trị 0, màu sáng nhất có giá trị 255.

Có rất nhiều phương pháp để che giấu thông tin trong các hình ảnh số nhưng phương pháp dụng nhất là chèn bit có trọng số nhỏ nhất (LSB). Ví dụ ta có: 11110110 là một số nhị phân 8 bit. Bit tận cùng bên phải ( bit 0 ) được gọi là bit LSB vì sự thay đổi của nó có ảnh hưởng ít nhất đến giá trị của số. Dữ liệu nhị phân của thông báo mật sẽ bị chia nhỏ ra và thay thế các bit LSB của từng byte trong giá trị RGB của từng Pixel. Sỡ dĩ chúng ta chỉ thay thế các bit LSB là vì sau khi thay đổi các giá trị trong RBG chỉ dao động 1 đơn vị, dùng mắt thường sẽ khó phát hiện ra sự thay đổi của bức ảnh, giúp đạt được mục đích ẩn dấu.  
Bởi mỗi pixel được tạo nên từ 3 kênh màu (RGB), nên ta có thể dấu tối đa 3 bit của thông báo mật.

Tóm gọn lại thì ta có các bước để dấu một thông điệp ẩn vào một bức ảnh bằng phương pháp LSB:

* Chuyển thông điệp bí mật sang dạng nhị phân.
* Thay từng 3bit của thông điệp vào LSB của các pixel liên tiếp trong ảnh cho đến khi hết thông điệp.
* Thay thế khung ảnh mới trong phần khung ảnh ban đầu.

Phần này đề xuất một thuật toán kết hợp các khái niệm Steganography (giấu thông tin), đối xứng mã hóa và song song hóa. Kiếm trúc của cách tiếp cận được đề xuất thể hiện trong sơ đồ phần tiếp theo.

TEXT

Audio

freams

Video Frames

Split

T1

T2

T3

Tm

**…**

Audio 1

Audio 2

Audio 3

Audio m

**…**

Embed

Frame 1

Frame 2

Frame 3

Frame m

Embed

Embed

Embed

Encrypt

Encrypt

Encrypt

Encrypt

**…**

Stego Video

Video

Key

Hình 3. 1. Sơ đồ giấu tin phương pháp đề xuất.

### Thuật toán mã hóa và nhúng

#### Input

* Tệp video.
* Thông điệp bí mật.
* Khóa mã hóa.

#### Output

* Video đã giấu tin.

#### Quá trình thực hiện

* Bước 1: Thông điệp bí mật được đọc và nội dung được chia thành các khối Ti (I = 1, 2, … m) để mã hóa.
* Bước 2: Sử dụng khóa bí mật mã hóa khóa khối Ti bằng thuật toán RSA.
* Bước 3: Tệp video cần đánh dấu bản quyền được tách thành video frames (khung hình) và audio frames bằng bộ thư viện FFMPEG.
* Bước 4: Chia Audio frames thành các phần rồi nhúng thông điệp bí mật đã hóa ở B2 bằng thuật toán LSB.
* Bước 5: Nhúng thông điệp bí mật đã mã hóa ở B2 vào video frames (khung hình) đã tách ở B3 bằng thuật toán LSB.
* Bước 6: Kết hợp kết quả audio frames và video frames thu được từ 2 bước trên thành video đã định bản quyền (stenography video).

### Sơ đồ mã hóa và nhúng

##### Mã hóa thông điệp

Luồng của quá trình mã hóa tạo ra một khối dữ liệu được phân chia. Dữ liệu này được XOR với một khóa bí mật. Đây là dữ liệu được mã hóa, được thể hiện trong hình 3.2

##### Nhúng thông diệp vào frame bằng LSB

Phương pháp chèn LSB phổ biến và đơn giản được sử dụng để nhúng thông tin vào Video frames. Các luồng riêng lẻ trong quá trình này nhúng thông tin đã được mã hóa vào LSB của khung hình.

##### Tách video bằng FFMPEG

FFmpeg là framework đa phương tiện hàng đầu, có thể giải mã , mã hóa , chuyển mã , mux , demux , truyền phát , lọc và chơi khá nhiều thứ mà con người và máy móc đã tạo ra. Nó hỗ trợ các định dạng cổ xưa khó hiểu nhất cho đến thời điểm hiện tại. Cho dù chúng được thiết kế bởi một số ủy ban tiêu chuẩn, cộng đồng hay một công ty. Nó cũng có tính di động cao: FFmpeg biên dịch, chạy và vượt qua cơ sở hạ tầng thử nghiệm của chúng tôi FATE trên Linux, Mac OS X, Microsoft Windows, BSDs, Solaris, v.v. trong nhiều môi trường xây dựng, kiến ​​trúc máy và cấu hình.

Ffmpeg là một công cụ chuyển đổi âm thanh và video rất nhanh cũng có thể lấy từ nguồn âm thanh / video trực tiếp. Nó cũng có thể chuyển đổi giữa các tỷ lệ mẫu tùy ý và thay đổi kích thước video nhanh chóng với bộ lọc polyphase chất lượng cao.

Ffmpeg đọc từ một số lượng tùy ý "tệp" đầu vào (có thể là tệp thông thường, đường ống, luồng mạng, thiết bị lấy, v.v.), được chỉ định bởi -itùy chọn và ghi vào một số "tệp" đầu ra tùy ý, được chỉ định bởi một url đầu ra đơn giản. Bất kỳ thứ gì được tìm thấy trên dòng lệnh không thể được hiểu là một tùy chọn được coi là một url đầu ra.

### 3.2.3. Thuật toán giải mã và trích xuất

Tách video gốc

Nhúng text vào audio.

Audio được ẩn text.

Output: Video mới đã ẩn text

(audio + video frames)

Run

ádasda

Audio

Audio được tách.

Chuyển sang nhị phân.

Áp dụng thuật toán LSB

Thông điệp bí mật

(INPUT: Text)

Chia text thành 16 khối bằng khau.

Mã hóa tất cả khối của Text bằng RSA.

Khóa mã hóa bí mật.

(INPUT: password)

Kết hợp Frame.

Thay thế Frames vào video

Video frames được ẩn text.

Chọn frame để ẩn dữ liệu

Cắt mỗi khung thành x phần

Ẩn khối của thông điệp cần mã hóa trong mỗi lần cắt

Kết hợp các khối được frame đã ẩn thông điệp.

Frames Video

Frames được tách.

Chọn frame.

Áp dụng thuật toán LSB

Input: Chọn video định dạng .avi

Hình 3. 2. Sơ đồ giấu tin bằng LSB.

#### 3.2.3.1. Input

* Stego- video
* Khóa mã hóa.

#### 3.2.3.2. Ouput

* Thông điệp bí mật.

#### 3.2.3.3. Quá trình thực hiện

* Bước 1: Nhận tệp video đã giấu thông điệp như là input.
* Bước 2: Check khóa mã hóa. Nếu đúng thực hiện các bước tiếp theo. Sai thì kết thúc chương trình.
* Bước 3: Tách tệp video đầu vào thành video frames và audio bằng FFMEPG.
* Bước 4: Chia Audio frames thành các phần rồi trích xuất thông điệp bí mật bằng thuật toán LSB.
* Bước 5: Chia Video frames thành các phần rồi trích xuất thông điệp bí mật bằng thuật toán LSB.
* Bước 6: Kết hợp kết quả thông điệp thu được từ audio frames và video frames thành thông điệp bí mật.

### 3.2.4. Sơ đồ giải mã và trích xuất

OUTPUT: Thông điệp gốc.

Run

INPUT: Chọn video cần tách Text.

Tách (ffmpeg)

Video frames

Audio

Chọn frame có thông điệp

Cắt mỗi frame thành 4 phần.

Trích xuất thông điệp từ mỗi lần cắt

Đúng

Giải mã thông điệp thu được từ mỗi frame

Kết hợp tất cả các thông điệp.

Thông điệp trong video frames

Thông điệp trong audio.

Chọn audio có thông điệp

Cắt mỗi audio thành 4 phần.

Trích xuất thông điệp từ mỗi lần cắt

Giải mã thông điệp thu được từ mỗi audio.

Kết hợp tất cả các thông điệp.

OUTPUT:

Kết thúc chương trình.

sai

PASSWORDRD

Hình 3. 3. Sơ đồ trích xuất thông điệp bằng thuật toán LSB.

Sơ đồ hình 3.3. quá trình giải mã nhận stego-video như đầu vào. Để trích xuất, các khung được cung cấp theo phương pháp trước đó được giải thích ở phần 1 với các giá trị ban đầu giống nhau. Việc trích xuất dữ liệu mã hóa được xử lý song song, tương. Điều này thực hiện sau quá trình giải mã song song.  
Sơ đồ được thể hiện trong hình A cho thấy hoạt động giải mã và hình B được tóm tắt phương pháp giải mã.

### 3.2.5. Phương pháp đánh giá

#### 3.2.5.1. Đánh giá chủ quan.

Dựa vào mắt thường khi nhúng thông điệp vào video sẽ không hiển thị, không làm thay đổi chất lượng hình hình video.

#### 3.2.5.2. Đánh giá khách quan.

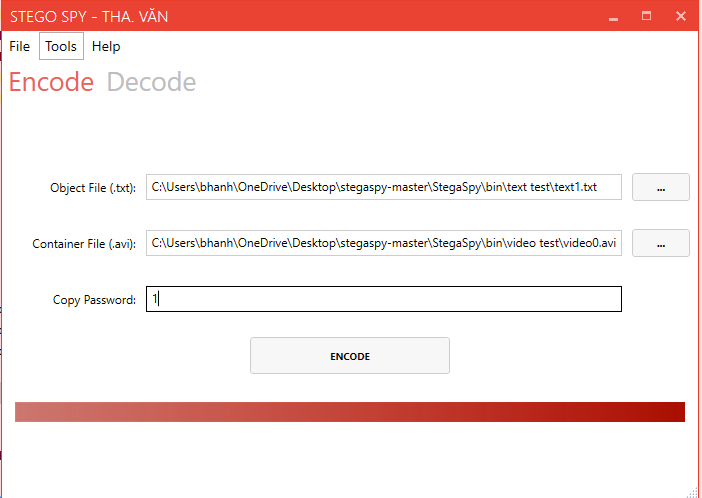
Thực hiện đánh giá kết quả sử dụng phương pháp liệt kê theo kết quả hiện thực đạt được thông qua các thông số độ phân giải, tốc độ khung hình trên giây, kích thước video,… sau khi nhúng thông điệp bí mật.

Đánh giá độ an toàn phương pháp bằng cách liệt kê, thử nghiệm vài bộ mật khẩu sai (hoặc thiếu mật khẩu).

# CHƯƠNG IV: CÀI ĐẶT, THỬ NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ

## 4.1. Cài đặt

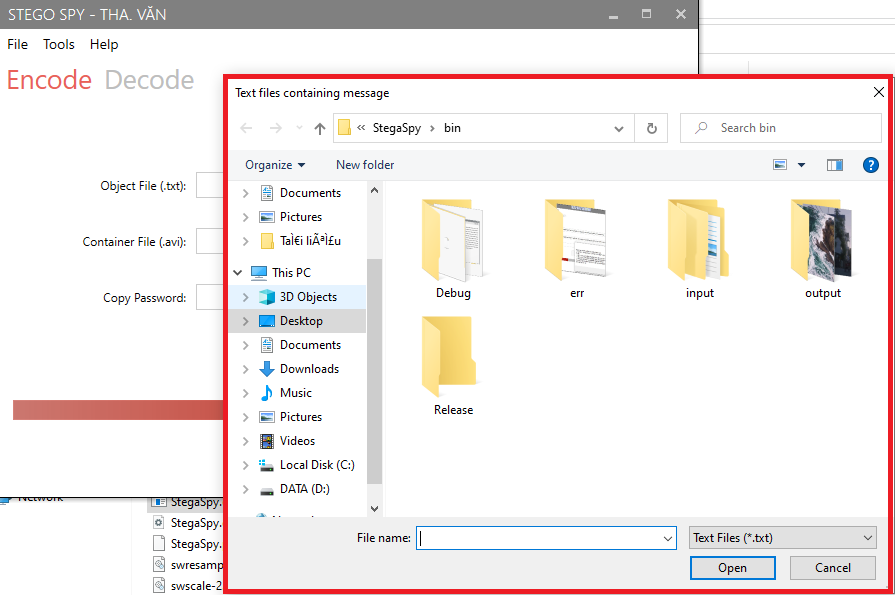
### 4.1.1. Giao diện mã hóa nhúng



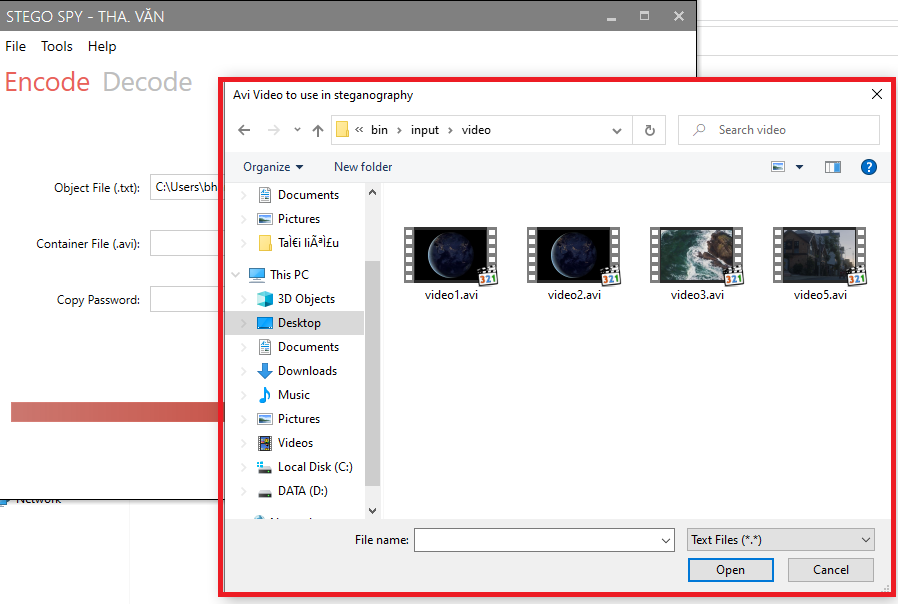
Hình 4. 1 Giao diện chương trình mã hóa nhúng.

* + Object File: chọn đường dẫn đến thư mục chứa tập tin .txt. Tập tin chứa thông điệp bí mật cần nhúng.
  + Container File: chọng đường dẫn đến tệp video được nhúng thông điệp.
  + Copy Password: nhập password mong muốn, nếu quá trình encode ( mã hóa/ nhúng) thành công thì chương trình sẽ sinh khóa, giữ password mới sinh này để sử dụng trong quá trình decode ( trích xuất/ giải mã).
  + Encode: kích chuột vào nút này để thực hiện quá trình mã hóa/ nhúng.
  + Tools: kích chuột vào sẽ xuất hiện giao diện video info để check thông số video.
  + Kích chuột vào nút … để chọn đường dẫn đến file video, file text.

Hình 4.1 thể hiện chi tiết giao diện của ứng dụng được thực hiện với các chức năng cụ thể như sau:

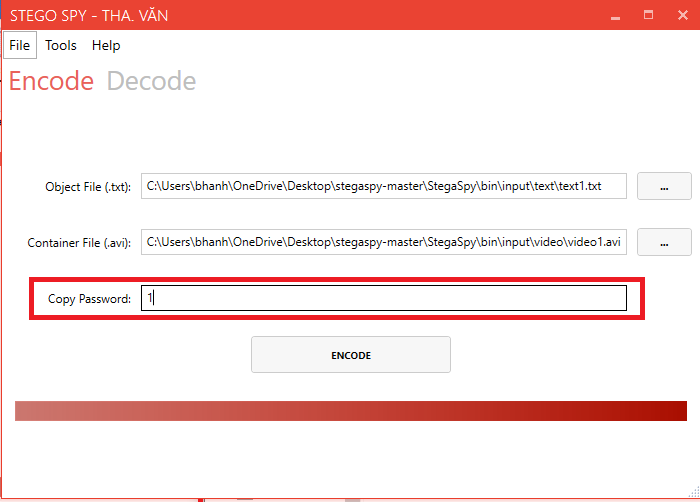
- Bước 1: Khi chọn button … dòng Object File sẽ xuất hiện giao diện như hình 4.2, để chọn file Text chứa thông điệp cần ẩn.

Hình 4. 2 Giao diện chọn file Text chứa thông điệp

* + Bước 2: Khi chọn button … dòng Container file sẽ xuất hiện giao diện tương tự hình 4.2 nhưng để chọn video cần mang thông điệp (hình 4.3).

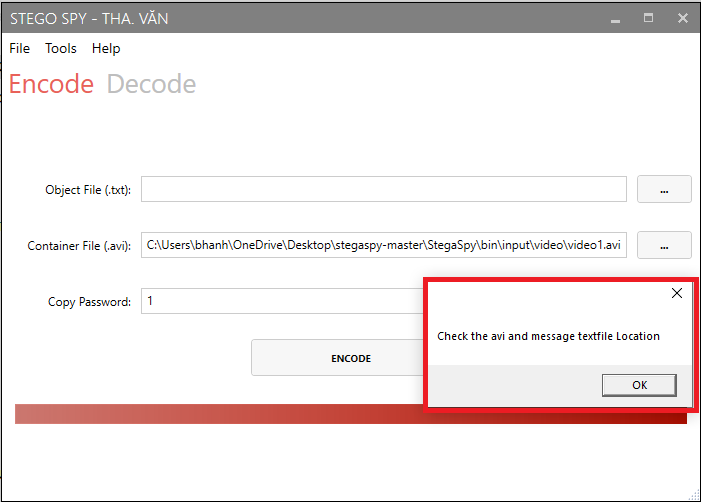
Hình 4. 3 Giao diện chọn video mang thông điệp.

* + Bước 3: Điền mật khẩu vào ô Password, như hình 4.4.



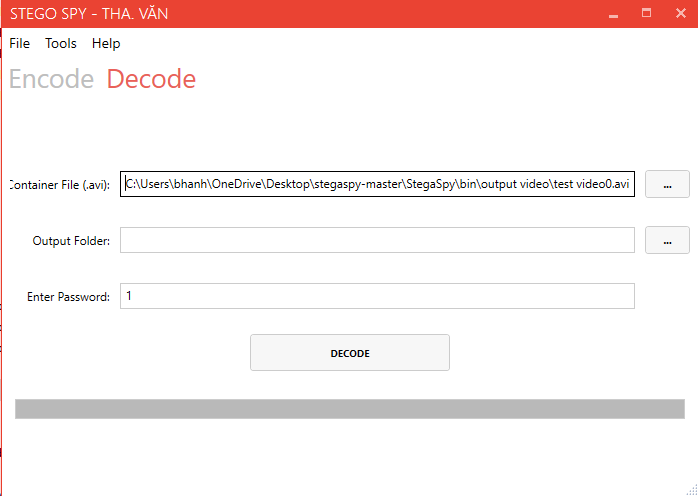
Hình 4. 4 Mật khẩu mã hóa

* + Bước 4: Nhấn button Encode để thực hiện quá trình mã hóa/ nhúng.
  + Bước 5: Đảm bảo chọn đường dẫn đến file text và file video để thực hiện quá trình này, nếu không chương trình sẽ hiện lỗi như hình 4.5.



Hình 4. 5 Lỗi khi thiếu thông tin đầu vào.

### Giao diện trích xuất/ giải mã



Hình 4. 6 Giao diện trích xuất giải mã.

* Chú thích hình 4.2:
  + Tương tự như hình 4.1
  + Container file: ở đây nhận video đã nhúng thông điệp (kết quả của quá trình encode) làm video đầu vào.
  + Output Folder: nơi xuất file .text ( hiện tại chưa có chức năng này)
  + Enter Password: quá trình encode thành công sẽ sinh ra khóa này.
    - Nếu nhập đúng thông tin này thì sẽ thực hiện quá trình Decode.
    - Nếu sai sẽ kết thúc chương trình.

Quá trình trích xuất/ giải mã thực hiện tương tự như quá trình mã hóa/ nhúng ở phần 4.1.1

### 4.1.3. Quá trình thực hiện

#### 4.1.3.1. Mã hóa/ nhúng thông điệp

* Bước 1: chọn đường dẫn đến file text chứa nội dung thông điệp.
* Bước 2: chọn đường dẫn đến video cần định danh bản quyền (file .avi)
* Bước 3: Click chuột vào Encode.
  + Bước 3.1: nếu thành công thì thông điệp được nhúng vào video.
  + Bước 3.2: xuất password được sinh ra ở phần Passwor Enter (cho dễ quan sát và sử dụng trong quá trình trích xuất/ giải mã (Decode).
  + Bước 3.3: xuất ra màn hình nhúng thành công.

#### Trích xuất/ giải mã

* Bước 1: chọn video mang thông điệp bí mật (file .avi).
* Bước 2: nhập password đã được tạo ở bước 3.2 quá trình mã hóa nhúng.
* Bước 3: click vào decode.
* Bước 3.1: nếu đúng thông điệp bí mật sẽ được xuất ra màn hình.
* Bước 3.2: nếu sai kết thúc chương trình.

## 4.2. Tập dữ liệu đầu vào

Gồm 4 bộ dữ liệu đầu vào. Mỗi bộ dữ liệu bao gồm 1 file video và 1 file chứa thông điệp bí mật định dạng .text, ngoài ra còn password (dùng dể sinh khóa sau khi nhúng thành công.). Chi tiết bộ dữ liệu đầu vào gồm mục File video và File text bên dưới.

**File video**

* Dung lượng lần lượt là: 334MB, 593MB, 889MB,791MB.
* Thời lượng lần lượt là: 30s, 30s, 05s, 10s.
* Chất lượng hình ảnh lần lượt là: 720p, 1080p, 720p.
* Số lượng khung hình trên giây là: 30fps

Bảng 4. 1. Tập dữ liệu đầu vào.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Video1 | Video2 | Video3 | Video4 |
| Length | 30s | 30s | 05s | 10s |
| Width | 480 | 640 | 1920 | 1280 |
| Height | 270 | 360 | 1080 | 720 |
| Data rate | 400 Kbps | 400 Kbps | 400 Kbps | 400 Kbps |
| Total birate | 400 Kbps | 400 Kbps | 400 Kbps | 400 Kbps |
| Frame rate | 30 Fps | 30 Fps | 30 Fps | 30 Fps |
| Size: | 334 MB | 593 MB | 889 MB | 791 MB |

**File text**

Có số lượng ký tự và dung lượng tăng dần lần lượt là

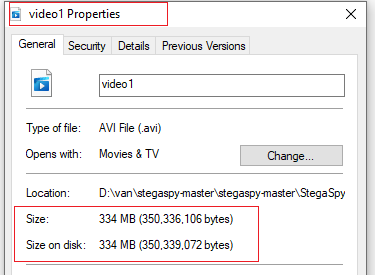
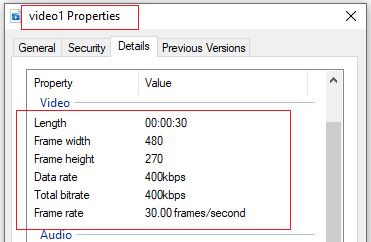
* Dung lượng là: 1kb, 5kb, 7kb, 45kb.
* Số lượng ký tự là: 39, 8918, 12,458, 49832.
* Bảng mã là: Unicode

Bảng 4. 2. Thông số bộ dữ liệu đầu vào- file Text.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Text1 | Text 2 | Text 3 | Text 4 |
| Size | 1 KB | 5 KB | 7 KB | 45 KB |
| Count | 39 | 8918 | 12458 | 49832 |
| Standard | Unicode | Unicode | Unicode | Unicode |

### 4.2.1. Bộ dữ liệu đầu vào 1

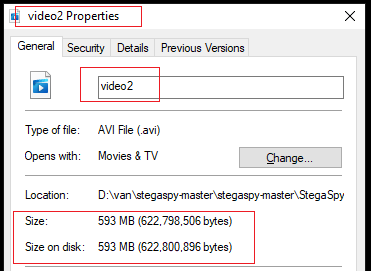
Bộ dữ liệu đầu vào gồm 1 file text chứa thông điệp và file video. File text được viết bằng bảng mã Unicode, gồm 39 ký tự. File video dài 30s, độ phân giải 480x 270, tốc độ khung hình trên giây 30 fps.



Hình 4. 7 Thông số bộ dữ liệu đầu vào 1.

### 4.2.2. Bộ dữ liệu đầu vào 2

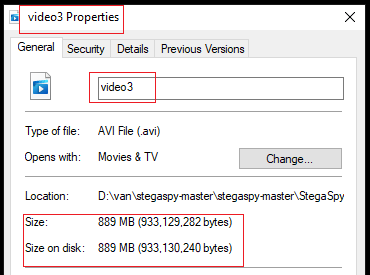
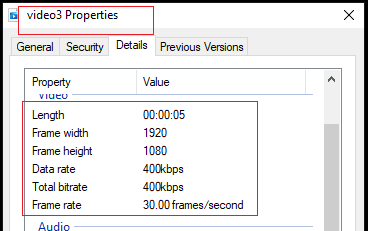
Bộ dữ liệu đầu vào gồm 1 file text chứa thông điệp và file video. File text được viết bằng bảng mã Unicode, gồm 8918 ký tự. File video dài 30s, độ phân giải 640x360, tốc độ khung hình trên giây 30 fps.



Hình 4. 8 Thông số bộ dữ liệu đầu vào 2.

### 4.2.3. Bộ dữ liệu đầu vào 3

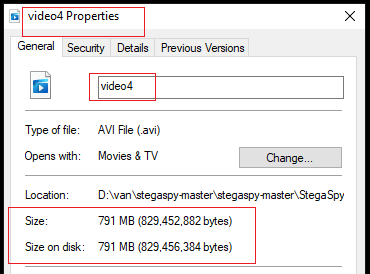
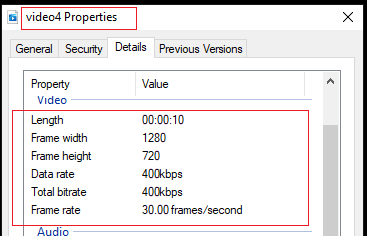
Bộ dữ liệu đầu vào gồm 1 file text chứa thông điệp và file video. File text được viết bằng bảng mã Unicode, gồm 12458 ký tự. File video dài 30s, độ phân giải 1920x1080, tốc độ khung hình trên giây 30 fps.



Hình 4. 9 Thông số bộ dữ liệu đầu vào 3.

### 4.2.4. Bộ dữ liệu đầu vào 4.

Bộ dữ liệu đầu vào gồm 1 file text chứa thông điệp và file video. File text được viết bằng bảng mã Unicode, gồm 49832 ký tự. File video dài 10s, độ phân giải 1280x720p, tốc độ khung hình trên giây 30 fps.



Hình 4. 10 Thông số bộ dữ liệu đầu vào 4.

## Thí nghiệm đánh giá kết quả.

Quá trình thực hiện tương tự như phần 4.1 cài đặt phần mềm thu được kết quả như bảng 3.

Bảng 4. 3. Đánh giá kết quả tương ứng.

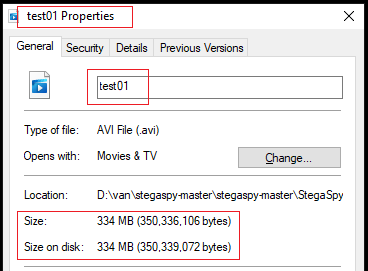
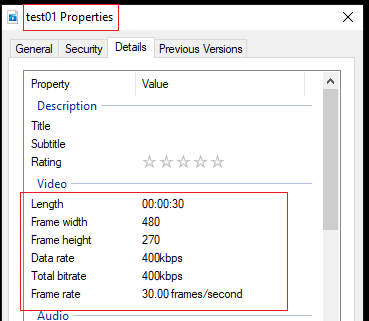
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Video1 | Test1 | Video2 | Test2 | Video3 | Test3 | Video4 | Test 4 |
| Length | 30s | V | 30s | V | 05s | V | 10s | V |
| Width | 480 | V | 640 | V | 1920 | V | 1280 | V |
| Height | 270 | V | 360 | V | 1080 | V | 720 | V |
| Data rate | 400 Kbps | V | 400 Kbps | V | 400 Kbps | V | 400 Kbps | V |
| Total birate | 400 Kbps | V | 400 Kbps | V | 400 Kbps | V | 400 Kbps | V |
| Frame rate | 30 Fps | V | 30 Fps | V | 30 Fps | V | 30 Fps | V |
| Size: | 334 MB | V | 593 MB | V | 889 MB | V | 791 MB | V |

* Chú thích sơ đồ 3:
  + V là không đổi.
  + Length: độ dài video/ thời lượng video.
  + Width: chiều rộng
  + Height: chiều cao.
  + Frame rate: tốc độ khung hình / giây.

### 4.3.1. Kết quả đạt được tương ứng bộ dữ liệu đầu vào 1

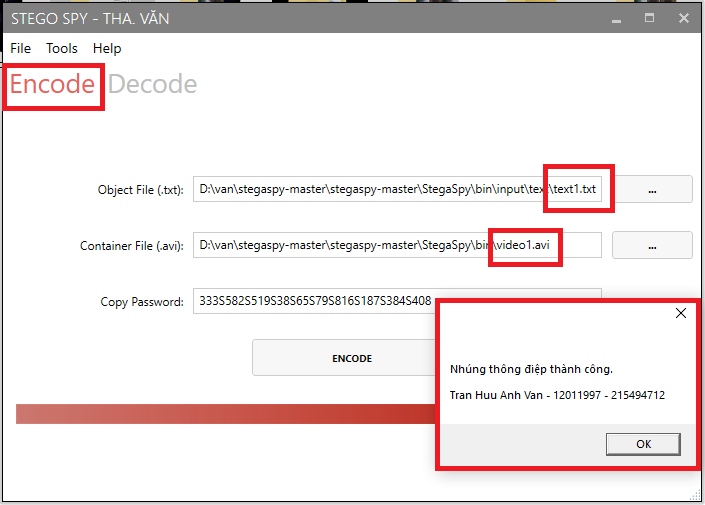
#### 4.3.1.1. Video đã nhúng

Video thu được sau quá trình thực nghiệm nhúng thông điệp bí mật có độ dài 39 ký tự vào có chất lượng và kích thước không đổi.

Hình 4. 11 Kết quả bộ dữ liệu đầu vào 1 - video nhúng thành công.

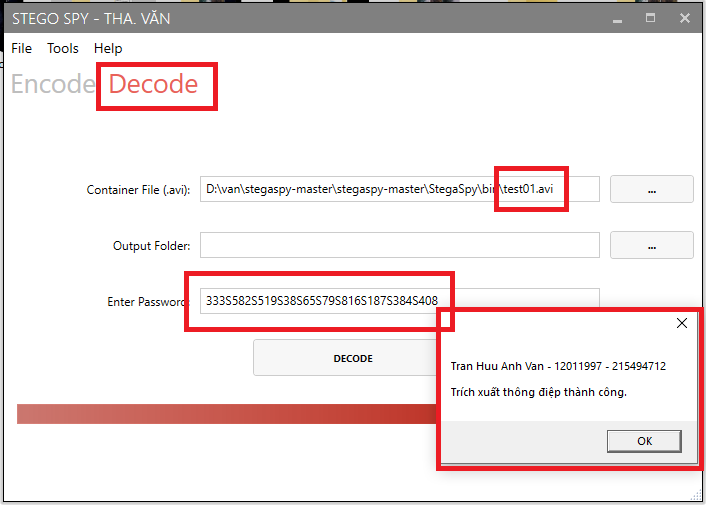
#### 4.3.1.2. Kết quả quá trình mã hóa/ nhúng



Hình 4. 12 Kết quả bộ dữ liệu đầu vào 1 - mã hóa/ nhúng thành công.

#### 4.3.1.3. Kết quả trích xuất/ giải mã

Thông được trích xuất được có bảng mã viết bằng Unicode, số lượng thông điệp tương đối giống với lúc nhúng do kiểm tra bằng mắt thường.

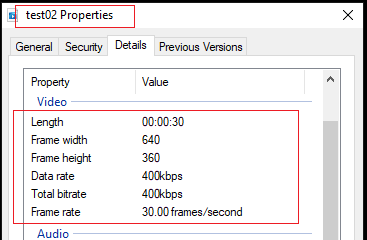
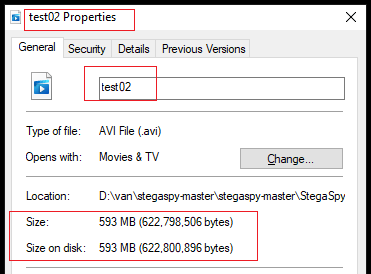


Hình 4. 13 Kết quả bộ dữ liệu đầu vào 1 - trích xuất/ giải mã thành công.

### 4.3.2. Kết quả đạt được tương ứng bộ dữ liệu đầu vào 2

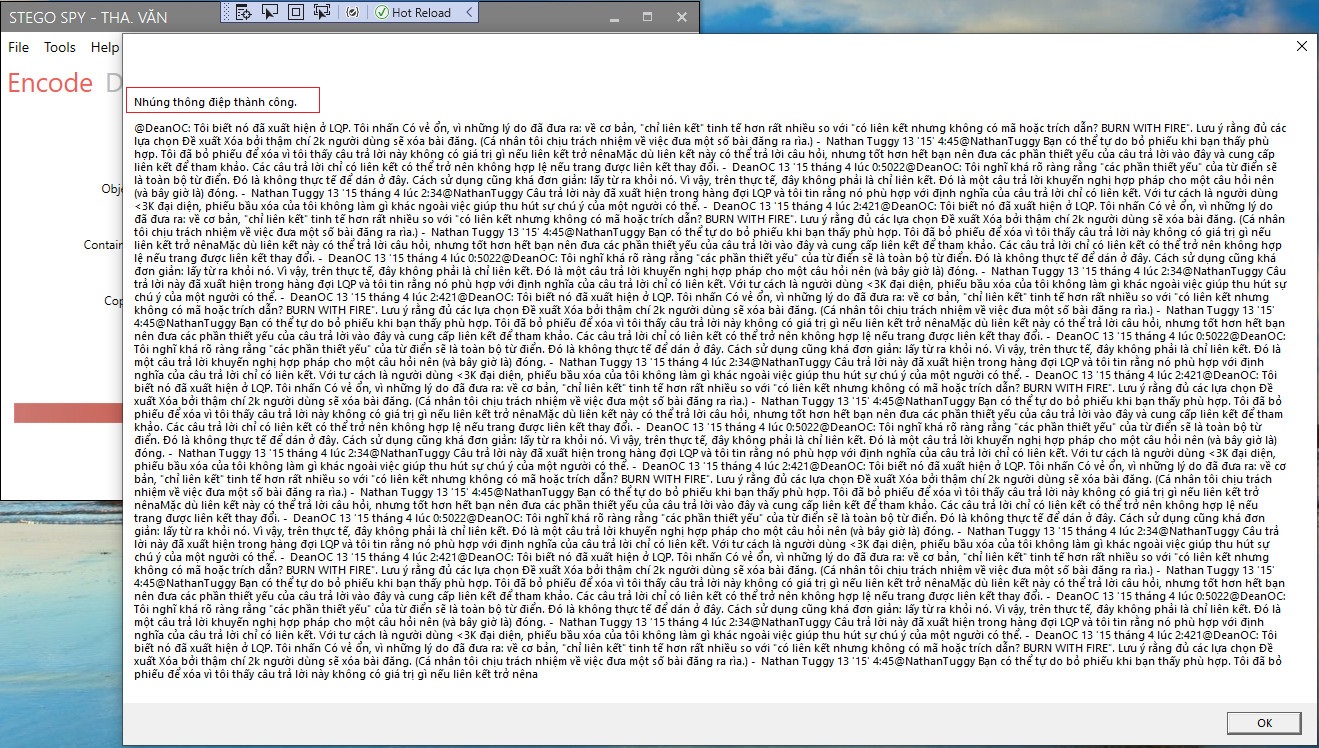
#### 4.3.2.1. Video đã nhúng

Video thu được sau quá trình thực nghiệm nhúng thông điệp bí mật có độ dài 12460 ký tự vào có chất lượng và kích thước không đổi.



Hình 4. 14 Kết quả bộ dữ liệu đầu vào 2 - video nhúng thành công.

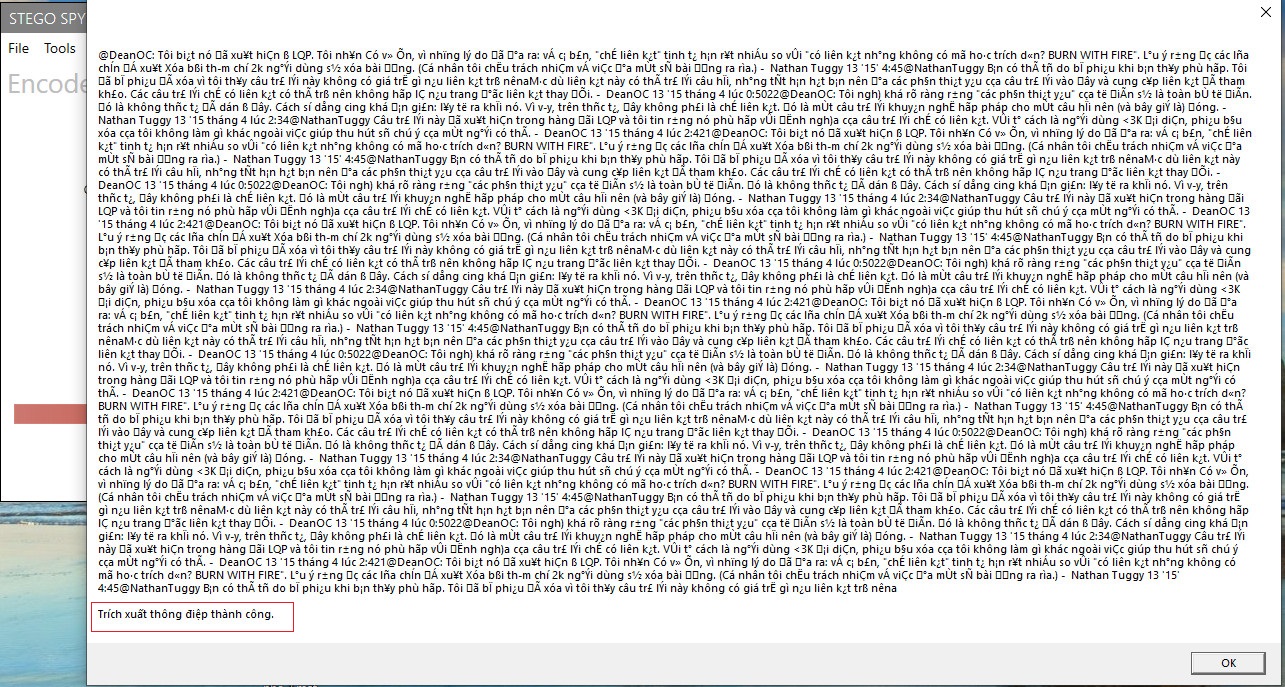
#### 4.3.2.2. Kết quả quá trình mã hóa/ nhúng



Hình 4. 15 Kết quả bộ dữ liệu đầu vào 2 - mã hóa/ nhúng thành công.

#### 4.3.2.3. Kết quả trích xuất/ giải mã

Thông được trích xuất được có bảng mã viết bằng Unicode, số lượng thông điệp tương đối giống với lúc nhúng do kiểm tra bằng mắt thường. Lỗi font chữ.

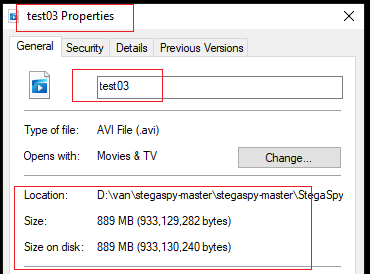
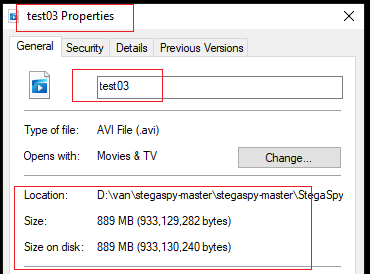


Hình 4. 16 Kết quả bộ dữ liệu đầu vào 2 - trích xuất/ giải mã thành công.

### 4.3.3. Kết quả đạt được tương ứng bộ dữ liệu đầu vào 3

#### 4.3.3.1. Video đã nhúng

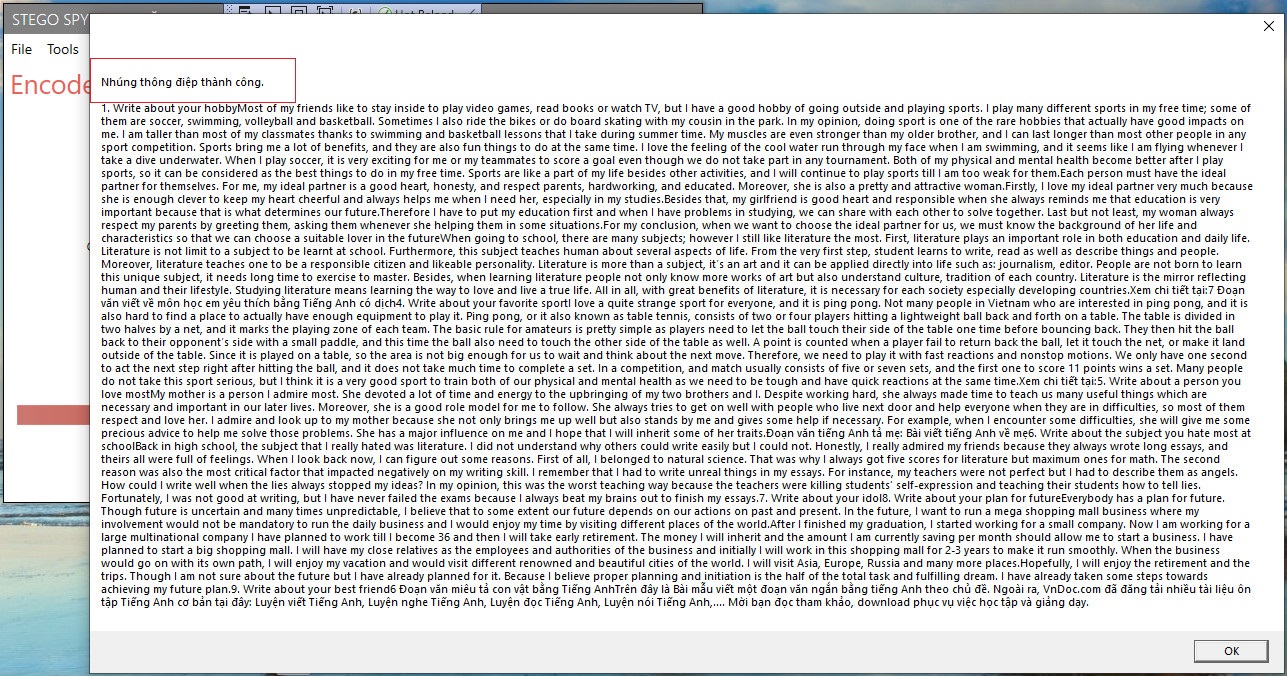
Video thu được sau quá trình thực nghiệm nhúng thông điệp bí mật vào có chất lượng không đổi.



Hình 4. 17 Kết quả bộ dữ liệu đầu vào 3 - video nhúng thành công.

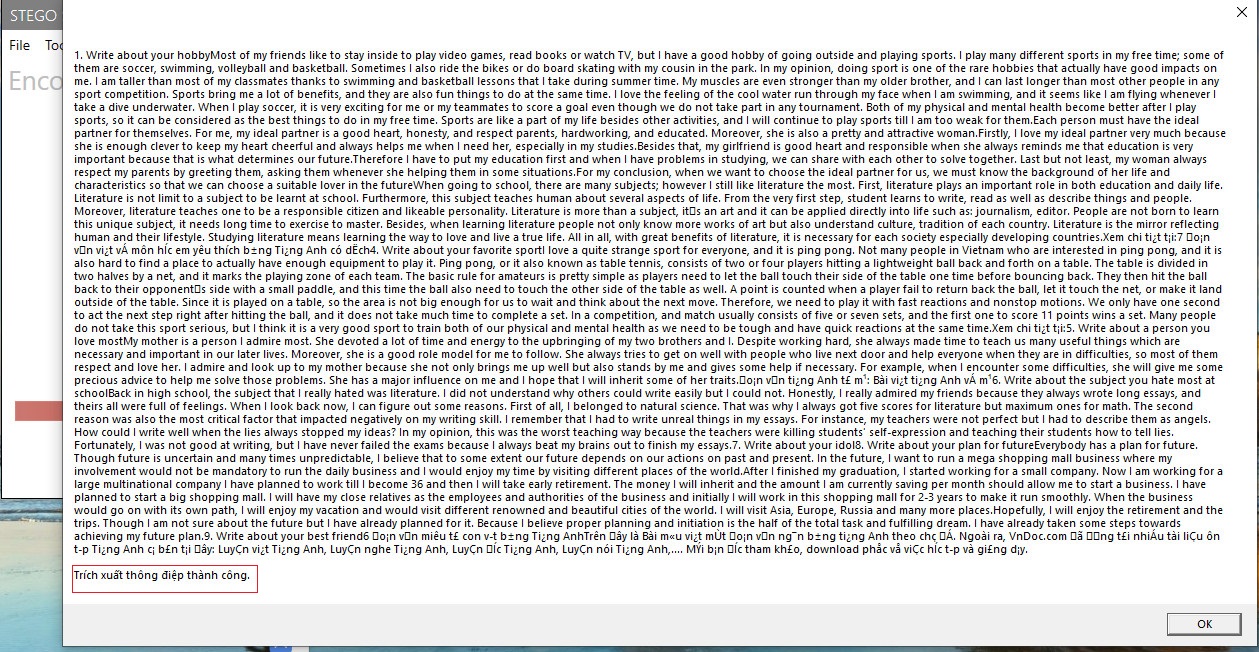
#### 4.3.3.2. Kết quả quá trình mã hóa/ nhúng

#### 4.3.3.3. Kết quả trích xuất/ giải mã



Hình 4. 18 Kết quả bộ dữ liệu đầu vào 3 - mã hóa/ nhúng thành công.

Thông được trích xuất được có bảng mã viết bằng Unicode, số lượng thông điệp tương đối giống với lúc nhúng do kiểm tra bằng mắt thường.

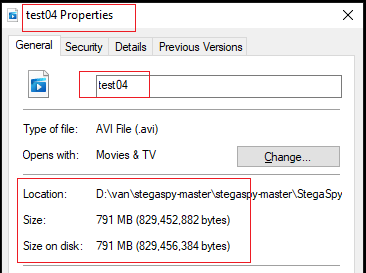
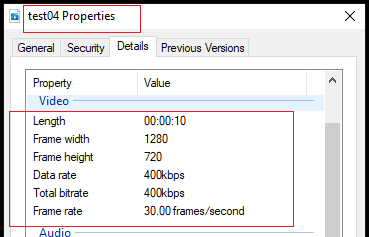


Hình 4. 19 Kết quả bộ dữ liệu đầu vào 3 - trích xuất/ giải mã thành công.

### 4.3.4. Kết quả đạt được tương ứng bộ dữ liệu đầu vào 4

#### 4.3.4.1. Video đã nhúng

Video thu được sau quá trình thực nghiệm nhúng thông điệp bí mật vào có chất lượng không đổi.



Hình 4. 20 Kết quả bộ dữ liệu đầu vào 4 - video nhúng thành công.

#### 4.3.4.2. Kết quả quá trình mã hóa/ nhúng



Hình 4. 21 Kết quả bộ dữ liệu đầu vào 4 - mã hóa/ nhúng thành công.

#### 4.3.4.3. Kết quả trích xuất/ giải mã

Thông được trích xuất được có bảng mã viết bằng Unicode, số lượng thông điệp tương đối giống với lúc nhúng do kiểm tra bằng mắt thường.

Hình 4. 22 Kết quả bộ dữ liệu đầu vào 4 - trích xuất/ giải mã thành công.

### 4.3.5. Đánh giá kết quả

Để thử nghiệm phương pháp được đề xuất, sử dụng video không nén (định dạng Avi) gồm 30 khung hình và thông tin văn bản được mã hóa rồi nhúng khác nhau. Kích thước của thông điệp bí mật khác nhau đa dạng từ 39 đến 49832 ký tự. Kết quả được mô tả ở hình trên.

#### 4.3.5.1. Ưu điểm

* Nhúng được lượng thông tin vào video.
* Trích xuất thành công thông tin nhúng.
* Không làm thay đổi chất lượng video.

#### 4.3.5.2. Nhược điểm

* Chưa quản lý được thông điệp xuất ra.
* Lỗi font chữ khi thông điệp đầu vào là tiếng việt nhiều ký tự.

# CHƯƠNG V: KẾT LUẬN

Kỹ thuật giấu thông tin trong video là một trong những hướng nghiên cứu chính của phương pháp giấu thông tin hiện nay và đạt được những kết quả khả quan. Đề tài nghiên cứu này đã tìm hiểu và trình bày một số kỹ thuật giấu tin điển hình. Thực nghiệm đã giấu tin trên khung hình và âm thanh. Trong thực tế chúng ta có thể lựa chọn chỉ giấu trong khung hình của video hoặc chỉ giấu trên âm thanh của video, tùy mục đích cụ thể trong thực tế. Do hạn chế về thời gian nên trong đề tài chỉ nghiên cứu kỹ thuật giấu tin chưa quản lý được thông điệp xuất ra.

## 5.1. Kết quả đạt được

Đồ án đã đạt được các nội dung sau đây:

1. Tìm hiểu cơ chế bảo mật thông tin trong video.
2. Tìm hiểu các công trình nghiên cứu liên quan và phân tích ưu nhược điểm của chúng.
3. Đề xuất cơ chế bảo mật thông tin trong video.
4. Xây dựng ứng dụng bảo mật thông tin trong video để xác định bản quyền của tác giả.
5. Đánh giá và so sánh các cơ chế bảo mật trong bảo vệ bản quyền tác giả.

## 5.2. Ưu nhược điểm phương pháp đề xuất

Tuy là phương pháp chèn vào bit ít quan trọng được sử dụng rộng rãi nhưng khi áp dụng chưa có bổ sung những công nghệ mới vào. Chỉ sử dụng phương pháp cũ. Phương pháp LSB thuần này còn nhiều hạn chế. Dễ làm mất các thông tin đã nhúng. Đây là phương pháp giấu tin ít bền vững.

## 5.3. Hướng mở rộng tương lai

Hướng nghiên cứu tiếp theo của đề tài sẽ nghiên cứu phương pháp giấu tin bền vững trên video định dạng khác như: Mpeg, vmv... Xây dựng thêm nhiều chức năng cho phần mềm. Quản lý thông điệp đâu ra, phát triển theo hướng hệ thống Content ID Youtube.

# TÀI LIỆU THAM KHẢO

1. Hoàng Văn Thức, (2011), *“Hệ tiêu chuẩn tham số an toàn cho hệ mật RSA và ứng dụng”, luận án tiến sỹ, Viện KHCN Quân sự.*
2. Lê Hải Triều, (2019), *“Phương pháp nghiên cứu bảo mật thông tin giấu trong ảnh số”, luận án tiến sỹ, Học viện công nghệ bưu chính viễn thông, trang 14-15.*
3. Jonathan Cummins, Patrick Diskin, Samuel Lau and Robert Parlett, (*2004), “Steganography and Digital Watermarking”, School of Computer Science, The University of Birmingham.*
4. B.SUNEETHA, CH.HIMA BINDU & S.SARATH CHANDRA, (2003), “SECURED DATA TRANSMISSION BASED VIDEO STEGANOGRAPHY”, *International Journal of Mechanical and Production Engineering (IJMPE) ISSN No.: 2315-4489, Vol-2, Iss-1*
5. A. Swathi 1, Dr. S.A.K Jilani, (2012), *”Video Steganography by LSB Substitution Using Different”, Polynomial national Journal Of Computational Engineering Research (ijceronline.com) Vol. 2 Issue. 5*
6. Ashawq T. Hashim, Dr.Yossra H. Ali & Susan S. Ghazoul, (2011) *“Developed Method of Information Hiding in Video AVI File Based on Hybrid Encryption and Steganography” Engg.and tech journal, vol 29,No.2,2011.*
7. R. Shanthakumari1 and Dr.S. Malliga, (2014), *“Video Steganography Using LSB Matching Revisited Algorithm”, IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 16, Issue 6, Ver. IV(Nov – Dec. 2014), pp. 01- 06.*
8. S. B. Kumar, D. Bhattacharyya, P. Das, D. Ganguly and S. Mukherjee, *(2008), “A tutorial review on Steganography”, International Conference on Contemporary Computing (IC3-2008), Noida, India, pp. 105-114.*
9. Atoum, M. S. Rababah, A. and Al-attili, A. I, (2011), *“New Technique for Hiding Data in Audio Files” International Journal of Computer Science andNetwork Security, 11(4), 173-177.*
10. Atoum, M. S. Ibrahim, S. Sulong, G. and Ahmed, A.(2013), *“New Secure Scheme in Audio Steganography (SSAS)”, Australian Journal of Basic and Applied Sciences, 7(6), 250–256.*
11. Atoum, M. S., Ibrahim, S., Sulong, G., Zeki, A and Abubakar, A. (2013), *“Exploring the Challenges of MP3 Audio Steganography”, Proceding IEEE from 2nd international Conference on Advanced Computer Science Applications and Technologies (ACSAT), Sarawak, Malaysia.*
12. Atoum, M. S. (2015, August), *“New MP3 Steganography Data Set”, In IT Convergence and Security (ICITCS), 2015 5th International Conference on (pp. 1-7), IEEE.*
13. R. Shanthakumari1 and Dr. S. Malliga, (Nov - Dec.2014), “*Video Steganography Using LSB Matching Revisited Algorithm”, IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 16, Issue 6, Ver. IV(Nov – Dec. 2014), pp. 01-06.*