Bài thực hành: Phát hiện giấu tin trong ảnh sử dụng thuật toán DCT

1. Mục đích

Bài thực hành này hướng dẫn phát hiện một ảnh đã được thực hiện giấu tin bằng thuật toán DCT và các công cụ khác. Sau khi hoàn thành bài lab sinh viên sẽ hiểu hơn về cách phát hiện ảnh đã được giấu tin cũng như quá trình trao đổi thông điệp được giấu tin.

2. Yêu cầu đối với sinh viên

Có kiến thức cơ bản về thuật toán DCT.

Có kiến thức cơ bản về ngôn ngữ Python.

3. Nội dung thực hành

3.1 Khởi động bài lab

Cài đặt bài lab tại: https://github.com/NgocTaif/detect_stego_code

Giản nén và chuyển vào thư mục */labtainer/trunk/labs.*

Trên terminal, gõ:

labtainer -r detect_stego_dct

Sau khi khởi động xong, một terminal ảo xuất hiện.

3.2 Nhiệm vụ 1: So sánh ma trận pixel và màu sáng của ảnh.

Trong bài thực hành đã được chuẩn bị sẵn một hai file ảnh *image1.png và image2.png* để ta tiến hành thực hiện so sánh, đánh giá và phát hiện điểm bất thường. Ta có thể sử dụng lệnh *fim* để có thể xem ảnh.

Trong các phương pháp giấu tin, đặc biệt là các phương pháp thay đổi miền tần số của ảnh như thuật toán DCT, thường sẽ thay đổi một lượng rất nhỏ các giá trị pixel trong ảnh để không ảnh hưởng trực quan đến mắt người. Tuy nhiên, nếu bạn có ảnh gốc để so sánh, việc kiểm tra ma trận điểm ảnh (pixel matrix) có thể phát hiện, xác định có hay không pixel thay đổi, giúp nghi ngờ có quá trình giấu tin.

Trước tiên, ta cần tạo hai ma trận điểm ảnh của hai ảnh *image1.png* và *image2.png*.

Trên terminal, ta nhập lệnh:

python3 create_img_matrix.py image1.png <tên_file_output>.txt

python3 create_img_matrix.py image2.png <tên_file_output>.txt

Kết quả đầu ra sẽ được lưu tại file .txt đầu ra do sinh viên tự tạo tên file.

Sau khi đã tạo thành công hai ma trận điểm ảnh cho hai ảnh, để so sánh xem chúng có sự khác biệt trong các hệ số điểm ảnh hay không, trên terminal, ta tiếp tục nhập:

python3 compare_matrix.py <file_matrix_image1>.txt <file_matrix_image2>.txt

Kết quả đầu ra có điều gì bất thường hay không, hãy lưu ý và ghi nhớ lại kết quả.

Một cách tiếp theo để có thể phát hiện ra ảnh đã được giấu tin hay không, ta có thể thực hiện tạo ảnh "chênh lệch" (difference image) để trực quan hóa. Ảnh "chênh lệch" sẽ hiển thị những vùng trong ảnh bị thay đổi so với ảnh gốc. Những vùng màu tối → không có sự thay đổi (pixel giống nhau). Những vùng màu sáng hoặc nhiễu lốm đốm → có sự thay đổi (pixel khác nhau). Đây là cách dễ dàng để "nhìn thấy" vùng có thể bị giấu tin mà mắt thường không phân biệt được giữa hai ảnh.

Trên terminal, ta gõ lệnh:

convert image1.png image2.png -compose difference -composite <ảnh_chênh_lệch>.png

Kết quả sẽ tạo một ảnh "chênh lệch", trong đó tên file ảnh đầu ra do sinh viên tự đặt tên.

Sinh viên hãy thực hiện xem ảnh "chênh lệch" bằng lệnh *fim* và phát hiện điều gì khác thường hay không? Nếu có hãy phân tích kết quả, điều đó sẽ hỗ trợ nhận biết có bao nhiêu khối ma trận điểm ảnh được sử dụng để giấu tin.

3.3 Nhiệm vụ 2: Phát hiện bất thường.

Khi đã thực hiện các phân tích và đánh giá tại nhiệm vụ 1, sinh viên đã xác định và kết luận được ảnh có điểm bất thường, có thực hiện giấu tin trong ảnh bằng thuật toán DCT.

Tại bài thực hành này, trong ảnh được giấu tin, người giấu đã thực hiên giấu một file chứa key giải mã trong ảnh. Sinh viên hãy sử dụng các câu lệnh *strings, binwalk* để phát hiện ra điều đó, và tiến hành lấy và đọc key giải mã được giấu bên trong bằng công cụ *binwalk*.

Sau khi đã trích xuất và lấy được file key giải mã thành công, hãy đọc file này xem nội dung bên trong của chúng là gì, điều này sẽ giúp sinh viên trả lời các câu hỏi trong nhiệm vụ 3.

3.4 Nhiệm vụ 3: Trả lời câu hỏi.

Sau quá trình phân tích, đánh giá và phát hiện các điểm bất thường ở các nhiệm vụ trên, sinh viên đã có được kết quả trong quá trình phát hiện ảnh được giấu tin trong bài thực hành.

Sinh viên hãy trả lời các câu hỏi trong file question.txt để hoàn thành bài lab: *nano question.txt*

- *How many the matrix blocks that was hided?*
- *How many the different rows between the two matrices?*
- *What is the index of the last matrix block?*

3.5 Kết thúc bài lab

Trên terminal, gõ:

stoplab

Khi bài lab kết thúc, một tệp lưu kết quả được tạo và lưu vào một vị trí được hiển thị bên dưới.

Sinh viên cần nộp file *.lab* để chấm điểm.

Để kiểm tra kết quả khi trong khi làm bài thực hành sử dụng lệnh:

checkwork

Trong quá trình làm bài sinh viên cần thực hiện lại bài lab, sử dụng lệnh:

labtainer –r detect_stego_dct