

Another Photo Editor

5조

201411257 컴퓨터공학과 강정모

201411307 컴퓨터공학과 이한강

201611284 컴퓨터공학과 이유진

1. Outline – Another photo Editor (이미지 편집기)

- 포맷 변환, 대칭, 회전 기능을 갖는 이미지 편집 프로그램
- 자체적인 image DRM 기능 제공

※ 기능 세부

- 포맷 변환 : jpg ↔ png ↔ bmp 포맷 간 변환
- 대칭 : 상하좌우 대칭
- 회전 : 90°, 180°, 270°, 360° 회전
- 자체적인 image DRM 기능(Steganography 기법 활용)
 - Image 생성자가 고유한 자신의 signature image를 삽입하는 기능
 - Image 내의 삽입된 signature image를 확인할 수 있는 기능

※ 직접 구현 (python)

- 포맷 변환, 대칭, 회전, Image DRM 기능

※ open source

- Python image library
(디코딩 결과 나온 RGB 값들을 numpy array에 넣어 image 생성 후 화면에 출력하기 위해 사용)
- Scipy library
(DCT 등 압축 연산 시 수식 지원을 위해 사용)

2. 최종산출물 형태 및 기능

- 최종 산출물 : 이미지 편집 기능을 제공하는 SW
- UI: terminal에서 실행되는 text 기반의 UI

※ I/O

- input : 명령어 command, 편집 대상이 되는 이미지
- output : 결과 message, 편집 처리 된 이미지

- 1) 포맷 변환
 - 결과 message 출력 및 변환된 이미지 파일 속성 출력
- 2) 대칭, 회전, DRM
 - 성공 여부 출력 및 별도의 창을 띄워 변환된 이미지 출력

[포맷 변환]

<input> - 명령어 command, 편집 대상이 되는 이미지

```
jungmo_k@JungMo-K-MacBookPro ~ % transform ./aaa.jpg bbb.png
```

<output> - 결과 출력, message로 변환된 파일 속성 출력

```
bbb.png: PNG image data, 2658 x 1220, 8-bit/color RGBA, non-interlaced
jungmo_k@JungMo-K-MacBookPro Desktop %
```

[대칭, 회전, DRM]

<input> - 명령어 command, 편집 대상이 되는 이미지

```
jungmo_k@JungMo-K-MacBookPro ~ % rotate -r 90 ./aaa.jpg
```

<output> - 성공 여부 출력, 별도의 창을 띄워 변환된 이미지 출력

```
success
jungmo_k@JungMo-K-MacBookPro Desktop %
```

3. Alternative solution

- Image processing library : pillow, OpenCV, matplotlib
- 기존에 존재하는 SW

1) 알씨 : 알툴즈에서 지원하는 이미지 편집 GUI software.

장점 : 압축 파일 이미지 보기 및 다양한 포맷 간의 변환 지원

단점 : 딱히 없는 것 같음

2) 그림판 : 윈도우에서 기본적으로 지원하는 이미지 편집 GUI software.

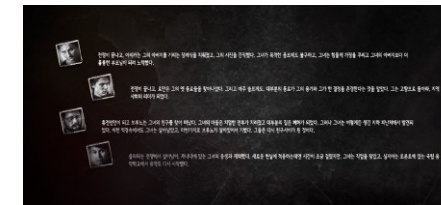
장점 : 심플하고 가벼운 소프트웨어, 다양한 색상 선택 가능

단점 : 주로 그리기 기능 위주, 이미지 편집 기능은 기본적인 것들만 존재

3) image magic : 이미지 편집 기능을 제공하는 opensource software.

장점 : 이미지 합성, 암호화 및 복호화, openCL 알고리즘 지원, 패턴 인식 등 가장 많은 기능 지원

단점 : 2016년 이후 몇 차례 보안 취약점 문제 발생



대상 이미지



창으로 출력되는
변환된 이미지

4. Project justification

- 이미 비슷한 SW가 다수 존재하지만, 멀티미디어공학개론에서 배운 이미지 이론을 실제로 구현해보는 의의
- 그림판의 기본적인 이미지 편집 기능 + 알씨/image magic의 포맷 변환 기능
- 부가적으로 Steganography 기법을 활용한 디지털 워터마크 기술 사용으로 차별성 부여

5. Risk analysis & reduction plan

1) 포맷 변환 및 이미지 출력

- Jpg 및 Png의 인코딩 및 디코딩이 복잡함.
 - 시간 부족에 따른 risk 발생 가능.
 - ⇒ 다른 기능 축소 or 일부 API 사용
 - ⇒ 핵심 기능과, 기능의 중요도에 따른 구현 우선순위 부여
 - ⇒ 우선순위가 낮은 기능의 경우 부득이하게 API로 구현
 - 1) 포맷 변환(핵심) 및 Steganography 기반의 DRM 기술(차별성)
 - 2) 회전
 - 3) 대칭

2) 효율적인 steganography 기법의 구현

- 대부분 하위 bit에 data를 넣으나 압축과정에서 손실 발생
 - ⇒ 효율적인 steganography 기법에 대한 논문을 참고해야 함

※ References

- 멀티미디어공학개론 강의 자료 (bmp, jpg)
- 위키피디아 (png, steganography, DRM)
- File format wiki (png, jpg, bmp)
- Python 표준 library site
- Pillow document site (image library 사용)

3) 다양한 option 및 mode

- Bmp 및 jpg의 경우 header에서 다양한 mode를 지원함
 - ⇒ default mode에 대해서 먼저 개발

6. Success criteria

1) 완성도에 대한 기준

핵심 기능 : 포맷 변환, Steganography 디지털 워터마크 DRM 기술 구현

(전체 개발 기간의 80% 예측 - 80% 구현의 기준)

부가 기능 : 대칭, 회전 기능 구현

(전체 개발 기간의 20% 예측 - 20% 구현의 기준)

2) 각 기능의 구현 성공에 대한 기준

- 포맷 변환 : 파일 속성 output message로 확인

[제대로 성공한 경우, data format이 바르게 표시]

```
bbb.png: PNG image data, 2658 x 1220, 8-bit/color RGBA, non-interlaced
jungmo_k@JungMo-K-MacBookPro Desktop %
```

[단순히 파일 확장자 명만 바뀐 경우 data format에서 드러남]

```
aaa.jpg: PNG image data, 2658 x 1220, 8-bit/color RGBA, non-interlaced
jungmo_k@JungMo-K-MacBookPro Desktop %
```

- 대칭, 회전 : 변환 후 이미지를 별도의 창으로 띄워 실제로 확인

⇒ 입력한 조건과 동일하게 변했는지 직접 확인 가능

- Steganography 디지털 워터마크 DRM

⇒ Steganography의 경우 정보 은닉이 목적이므로 존재 여부는 따로 알려주지 않음.

※ 확인 절차

1) Steganography 기법으로 삽입할 image와

워터마크가 삽입될 image를 준비함

2) Steganography 기법으로 디지털 워터마크를 삽입

3) 다시 추출 했을 때 육안으로 동일하다고 식별 가능한 디지털 워터마크 image가 나와야 함