

스마트폰 JPEG 파일의 출처 식별을 위한 DQT 정보 데이터베이스 구축*

김민식,[†] 정두원, 이상진[‡]
고려대학교 정보보호대학원

Building a Database of DQT Information to Identify a Source of the SmartPhone JPEG Image File*

MinSik Kim,[†] Doowon Jung, Sang-jin Lee[‡]
Graduate School of Information Security, Korea University

요 약

스마트폰을 이용한 사진 촬영이 보편화됨에 따라 이미지 조작, 기밀 유출로 인한 사건이 빈번하게 발생되고 있어 이미지 파일의 위변조 여부와 원본 증명에 대한 수요가 꾸준히 증가하고 있다. 일반적으로 스마트폰은 사진 파일을 JPEG 형식의 이미지 파일로 저장하는데 JPEG 이미지의 Header부분에는 이미지의 압축률을 결정하는데 사용되는 DQT가 저장되어 있다. 또한 JPEG 내부 Thumbnail 이미지에도 DQT가 존재한다. 기존의 연구에서는 DQT만을 이용해 이미지를 촬영한 장치를 판별하였다. 하지만 이 연구는 장치를 판별하기에는 정확도가 매우 낮다. 이에 본 논문에서는 DQT 정보뿐만 아니라 Thumbnail 이미지의 DQT 정보에 대한 실험을 통해 사진을 촬영한 스마트폰 기기와 사진 파일을 편집하고 저장한 애플리케이션을 유추할 수 있도록 JPEG 파일의 DQT 정보 및 Thumbnail 이미지의 DQT 정보 데이터베이스를 구축하여 보다 정확한 이미지의 출처 판별을 돕는다.

ABSTRACT

As taking pictures by using smartphones has become more common in society, there are many incidents which are unexpected manipulation of images and leak of confidential information. Because of those incidents, demands that identify forgery/alteration of image file and proves of the original copy is constantly increasing. In general, smartphone saves image file as JPEG form and it has DQT which determines a compression rate of image in a header part of image. There is also DQT in Thumbnail image which inside of JPEG. In previous research, it identified a smartphone which take image by only using DQT, However, the research has low accuracy to identify the devices. There are two main purposes in this research. First, this research will analogize a smartphone and an application that takes a picture, edits and save an image file by testing not only about a DQT information but also a information of Thumbnail image. Second, the research will build a database of DQT and Thumbnail information in JPEG file to find more accurate image file's origin.

Keywords: JPEG, DQT, Thumbnail, SmartPhone, Image

Received(07. 03. 2015), Modified(11. 02. 2015),
Accepted(02. 25. 2016)

* 이 논문은 2015년도 정부(미래창조과학부)의 재원으로 한국
연구재단-공공복지안전사업의 지원을 받아 수행된 연구임(2

012M3A2A1051106)

[†] 주저자, aminsik2@gmail.com

[‡] 교신저자, sangjin@korea.ac.kr(Corresponding author)

I. 서론

스마트폰의 사용이 급증하면서 시간과 장소에 구애받지 않고 누구든지 쉽게 카메라 촬영을 할 수 있게 되었다. 또한, 스마트폰의 성능 향상으로 굳이 촬영한 사진을 PC로 옮기지 않더라도 스마트폰에서 곧바로 이미지 파일을 손쉽게 편집할 수 있게 되었다. 이에 따라 최근 이미지 조작 및 기밀 유출로 인한 분쟁 사건이 빈번하게 발생되고 있어 스마트폰에서 발견된 이미지 파일의 위·변조 여부와 원본 증명에 대한 수요가 꾸준히 증가하고 있다.

현재 대부분의 스마트폰과 디지털 카메라는 사진 파일을 JPEG 형식의 이미지 파일로 저장한다. JPEG 이미지 파일의 Header에는 이미지에 대한 다양한 정보가 포함되어 있어 이미지 파일의 출처를 유추할 수 있으므로 디지털 포렌식 관점에서 매우 중요하다. 현재 JPEG 포맷에 대한 연구와 Header 정보를 해석하는 연구는 많이 진행된 상태이나 이를 포렌식 관점에서 사진 파일의 출처를 유추하는 연구는 부족한 실정이다.

이에 본 논문에서는 JPEG 이미지의 DQT와 Thumbnail을 함께 활용하여 사진을 촬영한 스마트폰 기기와 편집 여부를 확인하고 사진 파일을 편집한 애플리케이션을 유추할 수 있도록 데이터베이스를 구축하여 보다 정확한 이미지의 출처 판별을 돕는다.

II. 관련 연구

최근 출시되는 스마트폰은 모두 이미지파일을 저장할 때 JPEG 포맷을 사용하므로 EXIF 정보를 이미지 파일 내부에 저장하고 있다. JPEG과 JPEG 내부 파일에 존재하는 EXIF(Exchangable Image File format) 포맷에 대한 연구는 다수 진행되었다[1]. JPEG 이미지 Header에 존재하는 EXIF 정보에는 날짜, 시간정보, 카메라 설정, 저작권 정보 등 JPEG 이미지에 대한 정보들이 저장되어 있어 이미지를 촬영한 기기에 대해 파악할 수 있다. 하지만 EXIF는 위·변조가 매우 쉽고 이미지의 저장 방식에 따라 정보가 남아 있지 않는 경우도 있어 EXIF 분석만으로 이미지의 출처를 유추하기에는 한계가 있다.

JD Kornblum은 DQT를 통해 이미지를 작업한 소프트웨어를 구분할 수 있음을 설명하였다[2]. DQT를 통해 디지털 카메라나 스마트폰의 이미지

촬영 소프트웨어를 구분할 수가 있다. 하지만 단순 JPEG 이미지의 DQT만을 통해 장치를 판단하는 방식은 동일한 DQT가 존재할 가능성이 매우 높다. 본 논문의 실험 결과 동일한 DQT 형태를 가지는 스마트폰 모델이 많은 것으로 확인되었다. 또한 디지털 카메라와 같이 사용자 애플리케이션이 없는 경우에는 동일한 DQT가 존재할 가능성이 많이 높진 않지만 사용자 애플리케이션이 존재하는 스마트폰의 경우에는 애플리케이션에 따라 생성되는 DQT 형태가 매우 다양하기 때문에 동일한 DQT 형태가 존재할 가능성이 매우 높다. 따라서 DQT만으로 이미지의 출처를 판단하기에는 한계가 있다. 본 논문에서는 해당 문제점을 실험을 통해 밝혀내고 문제점을 보완하여 더 정확한 방법으로 이미지의 출처를 판단하는 방법을 제시한다.

또 다른 이미지 출처 유추 기술과 관련한 기술로는 카메라 센서 패턴 노이즈를 분석하여 이미지 저장 장치를 판별하는 기술이 존재한다. 사진 촬영 및 저장 과정에서 카메라 센서에 의해 의도치 않게 발생하여 카메라마다 고유의 센서 패턴 노이즈를 갖게 된다. 이를 이용하여 카메라 노이즈의 데이터베이스를 구축하고 이미지 저장 장치를 판별하는 연구가 진행되고 있다[3-6].

하지만 이러한 카메라 센서 패턴 노이즈를 이용한 방법은 여러 이미지 저장 장치를 모두 정확히 판별하기는 힘들다. 같은 모델의 스마트폰이라도 카메라의 상태에 따라 차이를 보인다. 만약 카메라에 흠집이나 기타 이물질로 촬영에 방해가 된다면 노이즈를 식별하는데 어려움을 겪으며 어두운 곳에서 촬영한 이미지는 해당 방법으로는 분석이 어려워 이미지 저장 장치를 판별하기 힘들다. 또한 유사도 분석을 통해 결과를 도출하는 방식으로 기준점에 의해 오차가 생길 가능성이 크다. 이러한 여러 문제점으로 인해 센서 패턴 노이즈를 이용한 방법으로 이미지 저장 장치를 정확히 판별하기에는 한계가 있다.

이에 본 논문에서는 스마트폰에서 저장한 JPEG 이미지의 DQT와 Thumbnail 정보들을 데이터베이스로 구축하여 이미지의 출처를 유추하는데 기존의 문제점들을 보완하고 보다 정확한 결과를 도출할 수 있게 한다.

III. DQT(Define Quantization Table)

DQT란 JPEG 이미지 안에 포함된 양자화 테이블

DQT	
DQT ID	0
DQT Length	67
DQT, Row #0	2 2 1 2 3 6 7 8
DQT, Row #1	2 2 2 3 4 8 8 8
DQT, Row #2	2 2 2 3 6 8 10 8
DQT, Row #3	2 2 3 4 7 12 11 9
DQT, Row #4	2 3 5 8 9 15 14 11
DQT, Row #5	3 5 8 9 11 14 16 13
DQT, Row #6	7 9 11 12 14 17 17 14
DQT, Row #7	10 13 13 14 15 14 14 14
DQT ID	1
DQT Length	67
DQT, Row #0	2 2 3 6 14 14 14 14
DQT, Row #1	2 3 4 9 14 14 14 14
DQT, Row #2	3 4 8 14 14 14 14 14
DQT, Row #3	6 9 14 14 14 14 14 14
DQT, Row #4	14 14 14 14 14 14 14 14
DQT, Row #5	14 14 14 14 14 14 14 14
DQT, Row #6	14 14 14 14 14 14 14 14
DQT, Row #7	14 14 14 14 14 14 14 14

Fig. 1. DQT (define quantization table)

블로써 이미지의 압축률을 결정하는데 사용된다. Fig.1은 DQT의 형태를 보여주고 있다. DQT는 이미지의 저장 방식이나 이미지의 화질에 따라 형태가 달라진다. 다시 말해 사진을 촬영하거나 편집했을 때 촬영 기기나 편집 도구에 따라 서로 다른 DQT 형태를 가지게 된다. 또한 JPEG 이미지는 Thumbnail 이미지를 포함하고 있어 이에 해당하는 DQT도 가지고 있다. DQT가 촬영 기기나 편집 도구마다 전부 고유한 값으로 존재하지는 않는다. 동일한 DQT 형태가 나타날 확률이 매우 높다. 하지만

JPEG 이미지의 DQT와 Thumbnail 이미지의 DQT를 동시에 비교한다면 두 가지 모두 동일할 확률은 매우 낮아진다. 만약 두 가지 모두 동일한 도구가 존재하더라도 이미지의 출처의 범위를 줄여주는 것만으로도 포렌식적 관점에서 의미 있는 일일 것이다.

IV. 실험

본 절에서는 이전 논문의 연구에 대한 실험과 본 논문에서 새로 제시하는 방식에 대한 실험 내용을 보여준다. 각각의 스마트폰의 기본 카메라와 써드 파티 애플리케이션으로 촬영 및 편집한 이미지의 DQT 정보와 Thumbnail DQT 정보를 확인하였다.

4.1 DQT 정보만을 이용한 촬영 스마트폰 유추

이전 논문의 연구에서는 JPEG 이미지의 DQT 정보만을 가지고 촬영한 장치를 판별하였다. 하지만 촬영 장치가 스마트폰인 경우 해당 연구는 크게 유용하지 않다.

Table 1은 여러 스마트폰 모델에 탑재된 기본카메라로 촬영한 사진의 DQT 형태를 보여준다. 총

Table 1. DQT form of image which has taken by the default camera application

Model	Software Version	DQT
SPH-W4700	BC27	Type 1
SHV-E160K	E160KKKJNB3	Type 2
SHV-E160K	E160KKKJMC3	Type 2
SHW-M250K	M250KK	Type 2
SHW-M340S	M340S.FH27.2302	Type 2
SHV-E210K	E210KKKJMK1	Type 3
SHV-E210K	E210KKKUGNA3	Type 4
SHV-E250S	E250SKSUENA1	Type 4
SHV-E300S	E300SKSUENA2	Type 5
SHV-E330S	E330SKSUAMFO	Type 6
SHW-M240S	M240S.TG18.1108	Type 7
LG-F240s	F240s10z	Type 9
LH8000	LH800909	Type 10
iPhone 5	iOS 7.1	Type 11
iPhone 5s	iOS 7.1	Type 11
iPhone 5	iOS 8.1	Type 12
iPhone 5s	iOS 8.1	Type 12
iPhone 6	iOS 8.1	Type 12
IM-A830S	S0214130	Type 13
IM-A840S	S0215216	Type 14

20개의 스마트폰 기기를 대상으로 기본카메라에서 촬영한 DQT를 확인하였다.

실험 결과 고유한 DQT 형태를 가지는 모델이 다수 존재하는 것을 확인할 수 있었다. 또한 같은 모델이지만 Software Version이 다른 경우 DQT가 달라지는 것을 확인할 수 있다. 하지만 동일한 DQT 형태를 가지는 스마트폰이 다수 존재하기 때문에 기본 카메라로 촬영한 이미지의 DQT만으로 이미지를 촬영한 스마트폰을 찾기엔 한계가 있다.

4.2 Thumbnail DQT를 함께 이용한 촬영 스마트폰 유추

JPEG 이미지에 해당하는 DQT 정보 뿐만 아니라 JPEG 이미지 내부의 Thumbnail 이미지에 해당하는 DQT 정보를 함께 이용하여 실험을 진행하였다. 또한 기존에 연구하지 않았던 애플리케이션에 대한 연구도 함께 진행하였다.

4.2.1 기본 카메라 애플리케이션

Table 2는 기본 카메라에서 촬영한 이미지들의

DQT 정보와 Thumbnail DQT 정보를 모두 비교한 결과이다.

두 가지 정보를 모두 비교한 결과 동일한 Type의 DQT 형태를 갖는 스마트폰이 매우 적음을 확인할 수 있다. iPhone의 경우에만 동일한 Type의 DQT 형태가 나타났으며 iPhone 또한 iOS version에 따라 형태가 달라지는 것을 확인하였다. iPhone을 제외한 모든 스마트폰은 고유한 정보를 가지고 있어 동일한 Type을 가지는 경우가 거의 없음을 확인할 수 있었다. 따라서 기존 연구의 DQT만으로 촬영 스마트폰을 판단한 결과보다 훨씬 높은 정확도로 기기를 유추해낼 수 있다. 이를 이용한다면 어떤 스마트폰에서 촬영하였는지 그 범위를 매우 좁힐 수 있으므로 기본카메라에 대한 DQT 데이터와 Thumbnail DQT 데이터를 추적하여 정확도를 더 높일 필요가 있다.

4.2.2 써드 파티 애플리케이션

스마트폰에는 무수히 많은 애플리케이션이 존재한다. 또한 각각의 애플리케이션은 스마트폰 모델에 따라 혹은 애플리케이션의 버전에 따라 조금씩 달라진

Table 2. Thumbnail DQT and DQT form of image which has taken by the default camera application

Model	Software Version	DQT and Thumbnail DQT	Note
SPH-W4700	BC27	Type 1	Unique
SHV-E160K	E160KKKJNB3	Type 2	Unique
SHV-E160K	E160KKKJMC3	Type 3	Unique
SHW-M250K	M250KK	Type 4	Unique
SHW-M340S	M340S.FH27.2302	Type 5	Unique
SHV-E210K	E210KKKJMK1	Type 6	Unique
SHV-E210K	E210KKKUGNA3	Type 7	Unique
SHV-E250S	E250SKSUENA1	Type 8	Unique
SHV-E300S	E300SKSUENA2	Type 9	Unique
SHV-E330S	E330SKSUAMFO	Type 10	Unique
SHW-M240S	M240S.TG18.1108	Type 11	Unique
LG-F240s	F240s10z	Type 12	Unique
LH8000	LH800909	Type 13	Unique
iPhone 5	iOS 7.1	Type 14	Non Unique
iPhone 5s	iOS 7.1	Type 14	Non Unique
iPhone 5	iOS 8.1	Type 15	Non Unique
iPhone 5s	iOS 8.1	Type 15	Non Unique
iPhone 6	iOS 8.1	Type 15	Non Unique
IM-A830S	S0214130	Type 16	Unique
IM-A840S	S0215216	Type 17	Unique

다. 본 절에서는 이미지를 촬영하거나 편집할 수 있는 애플리케이션으로 저장한 이미지의 DQT 정보와 Thumbnail DQT 정보를 확인하였다.

Table 3은 여러 애플리케이션으로 저장한 이미지의 DQT를 비교한 결과이다.

실험 결과 같은 스마트폰 모델인 경우 애플리케이션에 따라 DQT 형태가 달라지는 것을 확인할 수 있었다. 또한 운영체제의 종류, 애플리케이션의 버전에 따라 DQT 형태가 달라지는 것을 확인할 수 있었다. 이를 이용한다면 특정 스마트폰에서 어떤 애플리케이션을 사용하여 저장되었는지 그 범위를 매우 좁힐 수 있다. 하지만 DQT만을 이용한다면 어떤 스마트폰에서 저장된 이미지인지 유추가 힘들다. 따라서 Thumbnail DQT를 함께 이용하여 출처를 유

추하였다.

Table 4는 여러 애플리케이션에서 저장한 이미지의 Thumbnail DQT 형태를 비교한 결과이다.

실험 결과 동일한 스마트폰인 경우에는 애플리케이션의 종류와 상관없이 모두 동일한 형태의 Thumbnail DQT를 가진다. 이는 기본 카메라에서 촬영한 이미지도 해당한다. 다시 말해 같은 스마트폰에서 어떠한 작업을 거쳐 저장된 이미지는 모두 동일한 Thumbnail DQT를 가진다. 따라서 Thumbnail DQT가 다른 이미지라면 해당 스마트폰에서 촬영 및 편집이 이루어진 이미지가 아니며 두 가지 DQT를 함께 이용한다면 어떤 스마트폰의 어떤 애플리케이션에서 저장된 이미지인지 유추가 가능하다.

Table 3. DQT form of images that has taken and saved from the third party application

Model	Facebook	KakaoTalk	Naver Line	Pudding Camera	Photo Wonder
SHV-E250S	Type 1	Type 3	Type 5	Type 7	Type 9
SHV-E300S	Type 1	Type 3	Type 5	Type 7	Type 9
SHV-E330S	Type 1	Type 3	Type 5	Type 7	Type 9
SHW-M250K	Type 1	Type 3	Type 5	Type 7	Type 9
SHW-M340S	Type 1	Type 3	Type 5	Type 7	Type 9
LG-F240s	Type 1	Type 3	Type 5	Type 7	Type 9
LH8000	Type 1	Type 3	Type 5	Type 7	Type 9
iPhone 5	Type 2	Type 4	Type 6	Type 8	Type 10
iPhone 6	Type 2	Type 4	Type 6	Type 8	Type 10
IM-A830S	Type 1	Type 3	Type 5	Type 7	Type 9
IM-A840S	Type 1	Type 3	Type 5	Type 7	Type 9

Table 4. Thumbnail DQT DQT form of images that has taken and saved from the third party application

Model	Facebook	KakaoTalk	Naver Line	Pudding Camera	Photo Wonder
SHV-E250S	Type 1	Type 1	Type 1	Type 1	Type 1
SHV-E300S	Type 2	Type 2	Type 2	Type 2	Type 2
SHV-E330S	Type 3	Type 3	Type 3	Type 3	Type 3
SHW-M250K	Type 4	Type 4	Type 4	Type 4	Type 4
SHW-M340S	Type 5	Type 5	Type 5	Type 5	Type 5
LG-F240s	Type 6	Type 6	Type 6	Type 6	Type 6
LH8000	Type 7	Type 7	Type 7	Type 7	Type 7
iPhone 5	Type 8	Type 8	Type 8	Type 8	Type 8
iPhone 6	Type 8	Type 8	Type 8	Type 8	Type 8
IM-A830S	Type 9	Type 9	Type 9	Type 9	Type 9
IM-A840S	Type 9	Type 9	Type 9	Type 9	Type 9

V. 데이터베이스

본 절에서는 JPEG 이미지의 출처를 유추하는 절차를 제시하고 절차에 필요한 데이터베이스를 구축하고 DQT 정보를 수집하여 보다 더 정확하게 이미지의 출처를 유추할 수 있게 한다.

Fig.2는 JPEG 이미지의 출처를 유추하는 절차이다. 가장 먼저 Thumbnail 이미지의 DQT를 비교하여 해당 스마트폰에서 촬영이나 편집을 하였는지 확인한다. 두 번째로 해당 스마트폰의 기본카메라에 해당하는 DQT를 비교한다. 만약 일치한다면 해당 스마트폰에서 촬영한 이미지로 분류된다. 세 번째로 해당 스마트폰의 애플리케이션들에 해당하는 DQT와 비교 한다. 만약 형태가 일치한다면 해당 애플리케이션에서 촬영이나 편집을 통해 저장된 이미지로 분류된다. 하지만 Thumbnail DQT가 일치함에도 불구하고 동일한 DQT를 갖는 애플리케이션이 없다면 이미지의 조작 여부를 고려해야한다.

절차를 수행하기 위해서는 데이터베이스를 구축하

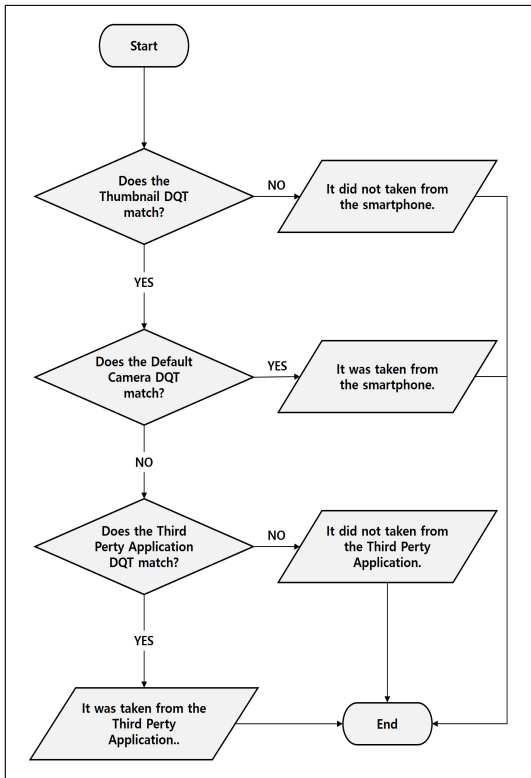


Fig. 2. Flowchart to identify a source of JPEG Image

여 DQT정보를 수집하여야한다. 절차에 따라 수집이 필요한 데이터는 Thumbnail DQT, 기본 카메라에서 촬영한 이미지의 DQT, 애플리케이션을 통해 저장한 이미지의 DQT 등 총 세 가지이다.

5.1 Thumbnail 이미지의 DQT

기본 카메라에서 촬영한 이미지의 DQT 정보와 함께 이미지 출처의 범위를 매우 좁힐 수 있으며 해당 스마트폰에서 작업한 이미지인지 그 여부도 판단할 수 있어 데이터 수집이 필요하다.

해당 데이터 수집을 위해 설계한 데이터베이스의 Primary Key는 (Smart Phone Model, OS Version) 이며 데이터베이스의 화면은 Fig.3과 같다.

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI
Model	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
OS_Verl	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Thumbnail_DQT	BINARY(500)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Date	DATETIME	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fig. 3. Thumbnail DQT Database table

5.2 기본 카메라로 촬영한 이미지의 DQT

Thumbnail DQT 정보와 함께 이미지 출처의 범위를 매우 좁힐 수 있어 데이터 수집이 필요하다.

해당 데이터 수집을 위해 설계한 데이터베이스의 Primary Key는 (Smart Phone Model, OS Version) 이며 데이터베이스는 Fig.4와 같다.

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI
Model	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
OS_Verl	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Default_Camera_DQT	BINARY(500)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Date	DATETIME	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fig. 4. DQT Database table of Default Camera Application

5.3 애플리케이션을 통해 저장한 이미지의 DQT

Thumbnail DQT 정보와 함께 이미지 출처의 범위를 매우 좁힐 수 있어 데이터 수집이 필요하다.

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI
Model	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
OS_Verl	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
App_Package_Name	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
App_Ver	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
App_DQT	BINARY(500)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Date	DATETIME	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fig. 5. DQT Database table of Third Party Application

해당 데이터 수집을 위해 설계한 데이터베이스의 Primary Key는 (Smart Phone Model, OS Version, Application Package Name, Application Version) 이며 데이터베이스는 Fig.5와 같다.

VI. 도구 개발

본 논문에서는 이미지 출처 판별을 위해 절차를 제시하고 절차를 포함하는 도구를 개발 하였다. Fig.6.은 도구의 메인 화면이다. 중요 기능으로는 JPEG 이미지의 EXIF 정보와 DQT 정보를 확인할 수 있으며 Thumbnail 이미지 확인도 가능하다. 또한 DB서버와의 통신을 통해 해당 JPEG 이미지가 어떤 스마트폰에서 촬영되었는지 알 수 있다. 본 논문에서 수집하고 있는 DQT와 Thumbnail 이미지의 DQT 정보를 이용하여 해당 이미지의 촬영 장치 판별을 돕는다. 데이터베이스의 수집은 계속 진행

중이며 수집이 진행될수록 출처를 파악할 수 있는 이미지 또한 늘어나고 있다.

VII. 결론

스마트폰의 사용량이 늘어남에 따라 포렌식 관점에서 JPEG 이미지의 출처를 알아내는 것은 매우 중요하다. 단순히 이미지 파일만을 가지고 어떤 스마트폰에서 촬영되었으며 어떤 애플리케이션을 통해 저장되었는지 유추할 수 있다면 포렌식 관점에서 매우 의미 있는 일이 될 것이다.

현재까지 JPEG 이미지만으로 출처를 유추하는 연구로는 카메라의 노이즈를 이용한 방법과 JPEG 이미지의 DQT만을 이용한 방법이 있다. 하지만 해당 방법들은 환경에 따라 변수가 많아 정확한 판단이 힘들다. 특히 DQT만을 이용하는 방법은 같은 형태의 DQT를 가지는 스마트폰이 다수 존재하기 때문에 출처를 식별하는데 한계가 있다.

따라서 이를 보완하기 위해 본 논문에서는 JPEG 이미지 Header 부분의 DQT 정보와 Thumbnail 이미지의 DQT 정보를 종합하여 분석하고 식별 절차를 세웠다. 또한 절차에 필요한 데이터베이스를 구축 하고 데이터를 수집하였다. 이를 이용하여 JPEG 이미지 파일의 촬영 스마트폰, 저장에 사용한 애플리케이션 등 이미지의 출처를 보다 정확하게 유추함으로써 포렌식 조사를 진행함에 있어 많은 도움이 될 것이다.

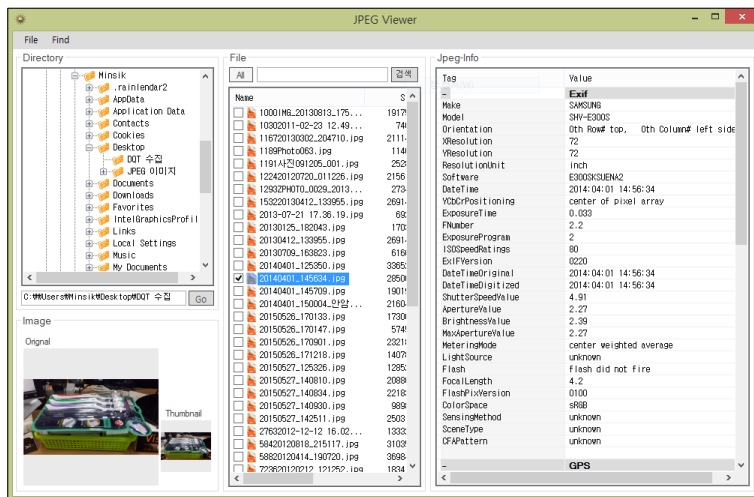
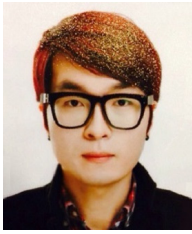


Fig. 6. Tool to identify a source of JPEG Image

References

- [1] Orozco, Ana Lucila Sandoval, et al, "Analysis of errors in exif metadata on mobile devices," *Multimedia Tools and Applications*, vol. 74, no. 13, pp. 4735-4763, July 2015
- [2] Kornblum, Jesse D, "Using JPEG quantization tables to identify imagery processed by software," *Digital Investigation* 5, pp. S21-S25, Sep 2008
- [3] Lukas, Jan, Jessica Fridrich, and Miroslav Goljan, "Digital camera identification from sensor pattern noise," *Information Forensics and Security, IEEE Transactions on*, vol. 1, no. 2, pp. 205-214, June 2006
- [4] Li, Chang-Tsun, "Source camera identification using enhanced sensor pattern noise," *Information Forensics and Security, IEEE Transactions on*, vol. 5, no. 2, pp. 280-287, June 2010
- [5] Goljan, Miroslav, Mo Chen, and Jessica Fridrich, "Identifying common source digital camera from image pairs," *ICIP 2007*, vol. 6, pp. 125-128, Sep 2007
- [6] Chen, Mo, et al, "Source digital camcorder identification using sensor photo response non-uniformity," *Electronic Imaging 2007*, pp. 65051G-65051G, Feb 2007
- [7] Bayram, Sevinc, et al, "Source camera identification based on CFA interpolation," *ICIP 2005*, vol. 3, pp. 69-72, Sep 2005
- [8] Soobhany, A-R, et al, "Mobile Camera Source Identification with SVD," *Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering*, vol. 313, pp. 123-131, 2015
- [9] S Hamdy, "Quantization Table Estimation in JPEG Images," *International Journal of Advanced Computer Science and Applications*, vol. 1, no. 6, pp. 17-23, Dec 2010
- [10] Thai, Thanh Hai, Florent Reiraint, and Rémi Cogramne, "Camera model identification based on the generalized noise model in natural images," *Digital Signal Processing*, vol. 48, pp. 285-297, Jan 2016
- [11] Lee, Sang-Hyeong, et al, "Digital Camera Identification Using Sensor Pattern Noise," *The Second International Conference on Information Security and Digital Forensics 2015*, pp. 30, 2015

〈저자소개〉



김 민 식 (KyuChul Yeon) 학생회원
 2014년 2월: 광운대학교 컴퓨터소프트웨어학과 공학사
 2014년 3월~현재: 고려대학교 정보보호대학원 석사과정
 <관심분야> 디지털 포렌식, 모바일 포렌식, 사이버 범죄 수사



정 두 원 (Doo-won Jeong) 학생회원
 2011년 8월: 고려대학교 공과대학 산업경영공학과 공학사
 2011년 9월~현재: 고려대학교 정보보호대학원 정보보호학과 석·박사통합과정
 <관심분야> 디지털 포렌식, 데이터 마이닝, 이미지 포렌식



이 상 진 (Sang-jin Lee) 종신회원
 1987년 2월: 고려대학교 수학과 학사
 1989년 2월: 고려대학교 수학과 석사
 1994년 8월: 고려대학교 수학과 박사
 1989년 10월~1999년 2월: ETRI 선임 연구원
 1999년 3월~2001년 8월: 고려대학교 자연과학대학 조교수
 2001년 9월~현재: 고려대학교 정보보호대학원 교수
 2008년 3월~현재: 고려대학교 디지털포렌식연구센터 센터장
 2015년 1월~현재: 고려대학교 정보보호대학원 부원장
 <관심분야> 디지털 포렌식, 심층 암호, 해쉬 함수