Forensic Analysis of Binary Structures of Video Files

[1]Md Abir Hasan, [3]Nusrat Jahan,
Department of Electrical and Computer Engineering
University of Alaska Fairbanks
Fairbanks, Alaska.
Mhasan2@alaska.edu
njahan@alaska.edu

[2]Orion Lawlor
Department of Computer Science
University of Alaska Fairbanks
Fairbanks, Alaska.
lawlor@alaska.edu

Abstract— **As technology advances, multimedia files such as videos are susceptible to manipulation. This has led to serious concerns that images and videos are not trustworthy evidence as the files can be manipulated easily. As a result, forensic analysis of electronic multimedia files plays an important role in verifying the authenticity of video files. This paper provides comprehensive details of a binary file forensic analysis technique for different media file containers, mostly focused on AVI and MP4/MOV container format. We also provide a considerable number of details to identify a forgery among video files. We present pivotal parameters which need to be tested to authenticate a video file. By analyzing the binary data structures and metadata, we can detect the use of editing tools, verify the purported source of a video file, and identify the true acquisition device model.**

Keywords- ***Forensic Analysis, metadata, video container, data structures, authentication.***

I. Introduction

All electronic files are vulnerable to editing. With the advancement of technology nowadays, it is very easy to change media files, and edit, modify, and alter the original data. By changing these files, the authenticity of the files become vulnerable and it can alter the value and the meaning of critical evidence. With these manipulated files people might present false information in court, disseminate fake news, or perpetrate other serious forgeries. These modifications can be impossible to detect or very difficult to identify through visual inspection. As a result, the authenticity of media files needs to be verified. By analyzing file format metadata and binary data structures, we can extract an additional source of data on these files.

Among media files, digital videos and photographs are vulnerable to this manipulation, as their origin and editing history are not always trustworthy [1]. Furthermore, the identification of the original owner is very difficult to achieve so it is harder to detect copyright infringements and the validation of the legal propriety of the multimedia files. As a consequence, much recent research addresses forensic analysis of multimedia data. Most research in this field is devoted to still image analysis, as digital photographs are used extensively in medical, legal and other applications. As a result of intense research in digital image forensic analysis, nowadays it is possible to determine whether an image is authentic or not.

Although much significant research has taken place on digital image forensics, still digital video forensics is at an early stage of the research. A big challenge for the video forensics is that it has diverse characteristics in comparison to still images and it provides abundant opportunities for alterations of its content. In this paper, we will present the forensic analysis techniques for video file formats. For that, we will discuss background where past research will be described. Later we will describe video file formats and their data structures. Based on our test setup we will analyze different multimedia cameras' video file formats and demonstrate the differences among some example cameras. Beside different camera container's data structures, we will also examine files edited with video editor software. For a video edited file, four different software tools were used, and their data structures will be compared with the original file's data structures. Besides using various software tools, we also attempted to keep same metadata from the original video and after that, we studied the edited files to see the peculiarities. The paper closes with a discussion concluding remarks.

II. Background

Most prior work has been focused on digital still images. With the help of file format information and metadata, and image approaches such as sensor fingerprinting [2] forgeries on digital images can now be detected. In case of sensor fingerprinting forgeries, some intrinsic fingerprint traces such as the process of color filter array (CFA) [3], Camera Response Function (CRF) [4], sensor pattern noise [5] can be used to detect multimedia tampering by copy and paste [3], slicing [4], correlation [5][6], resampling[7].

Besides the forensic analysis of digital images, researchers also have focused on the video forensic techniques in recent years. Early research on video forensics was done by Milani et al. [8] where they provided an overview of video forensics techniques. They also provided all the possible alterations that can be applied to a single video file and the possibilities for identifying vulnerabilities present in forged files to find out important information about the original file. But the most comprehensive research on authentication of video techniques was done by Gloe et al [9]. This study provides a thorough analysis of specific video file formats, digital cameras, mobile phones, and video editing software. They used 19 digital camera models, 14 mobile phone models and 6 video editing software toolboxes to present a comprehensive analysis of AVI, mp4 and edited video file formats. In another article, Hall et al [10] showed authentication techniques for MP4 files. For his test, he used 66 video recordings including camera and mobile devices, and mainly showed the differences among different MP4 structures.

III. DIFFERENT CAMERA FORMATS

Different cameras use different media container formats. Among the most commonly used formats are- AVI container format [11], MPEG-4/MP-4[12], and MOV [13].

A. AVI Container Format

AVI stands for Audio Video Interleave. Microsoft introduced this format in 1992. This format can be seen in many first generation DSLR cameras from Nikon. This format can use different video and audio encoding techniques such as DivX and XviD. This format is based on the Resource Interchange File Format (RIFF) also used by WAV audio: the data is organized into "chunks" or blocks, each chunk is identified by a Four Character Code (FourCC), and LIST chunks can contain other chunks. An AVI file starts with a mandatory *AVI RIFF header*. Then it stores next mandatory list *hdrl* as shown in figure I. This list generally contains the information about the video width, height, and frame rate. Additionally, an optional sub-chunk might present after *hdrl*. The next mandatory sub-chunk is *movi*. This contains the actual audio/video data that forms the AVI movie. The last sub-chunk is idx1 chunk which indexes the data chunks and their location in the file to allow playback. Note that all of these structures are not necessarily present in every AVI file, but generally this structure is maintained in most AVI container files.

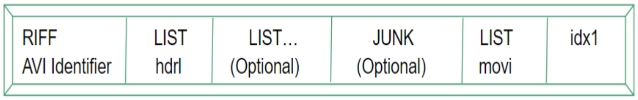

Figure I. General Data structures of AVI container

B. MOV Container

Apple first introduced this format in 1991. This format is also known as Apple QuickTime format. It can use several diverse codec techniques but most commonly the H.264 codec is used. Different cameras such as Nikon, Canon, Panasonic, and Sony mainly create this container form. This format is the basis of all the MP4-like formats. So, the structure of all the MP4-like formats is similar. This type of containers generally consists of individual data structures which are known as 'atoms'. These atoms are identified by unique 4-byte sequences. These video files generally start with a *ftyp* atom. These types refer the file type specifications and the compatibility. Then it has *moov* atom, which contains the metadata of the videos. Then *mdat* atom is used after *moov*. This atom is stored the actual data stream. There might be another atom *moof* for movie metadata, which is optional in this type of container. Figure II shows the binary data structures inside MPEG-4/MP-4/MOV containers.

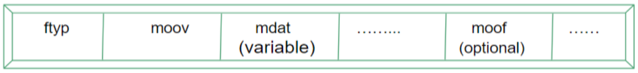

Figure II. General Data Structures of MPEG-4/MP-4/MOV containers

C. MP4 Container:

The MPEG-4 format is an industry standard format is used mostly on the web. It developed on the basis of MOV and introduced by Apple in 2003. In this format, H.264 is used as video compression technique and AAC compression is used as an audio compression technique. This format is similar to the MOV file format which also has 'atom' or 'box' structures. For the MP-4 containers individual data structures are known as 'box'.

IV. TEST SETUP

We used different cameras to collect different formats of video files. We compared the same formats on different camera models, different formats of different cameras, same formats of different camera models, and mobile phone models. We examined overall 11 cameras and 4 video editing programs. During video capturing all devices were subject to motion. Table I presents the device information we have used in our test setup.

We also edited the original video file using 4 video editing programs- Adobe Premiere, Corel Video Studio, VLC Media Player, and Windows Movie Maker (WMM). During the output of the video files, we tried to match the metadata with the original as much as possible.

During data collection, we measured whether the data structure of a collected file is altered during their transmission period from the respective device. To test this scenario file were tested by collecting the data in three different ways.

Table I. Test setup devices

Make	**Model**	**Container**	**Video Codec**	**Audio Stream**
Canon	PowerShot A520	AVI	MJPG	PCM
Casio	EX-FC100	AVI	MJPG	ADPCM
Fujifilm	FinePix2600Zoom	AVI	MJPG	PCM
Nikon	D5000	AVI	MJPG	PCM
Samsung	S730	AVI	MJPG	ADPCM
Canon	T3i	MOV	AVC	PCM
General Imaging Co.	E1680W	MOV	AVC	ADPCM
Nikon	Coolpix L620	MOV	AVC	PCM
Apple	iPhone 6S	MOV/MP4	AVC	PCM/AAC
Canon	T6i	MP4	AVC	AAC
Samsung	WB350F	MP4	AVC	AAC

a) Connecting removable memory devices directly in the data collection computer.

b) Connecting the camera device directly to the collection computer through a USB cable.

c) In the case of a mobile device, we tested the transfer process by sending the file via Gmail.

We collected the files these ways and tested them using different file hash techniques. The hash values remained the same for all the collection methods tested, which confirms that the structures of the files were not changed.

V. Analysis of Camera files

We analyzed 11 camera files as stated in our test setup. Most of the recent cameras store in MP4 or MOV container format and older cameras generally stores in AVI container format.

We used different tools to analyze the files. For metadata analysis, we used *MediaInfo* and *ExifTool*. For the data structures, we used file viewer plus and other open source hex file viewer tools. In this section we will compare among same device / different format files, different devices / same container format, and different device / different container formats.

Among different file containers, there are significant differences among the general structure of AVI and MP4/MOV containers. But there are fewer differences between MOV and MP4 containers. A comparison is shown in table II between MP4 and MOV files data structure. Note that Canon T3i stores files in MOV container format and Canon T6i stores files in MP4 format. By inspecting the data structures, it can be inferred that the data structures of MP4 and MOV file are similar. The major difference is in ftype file for MOV and MP4. The byte size of ftype is 28 for MP4 whereas for MOV the value is 24.

Table II. Two different MP4 files for two different devices.

Canon T6i (MP4)			Canon T3i (MOV)		
Atom Type		Size in Decimal	Box Type		Size in Decimal
ftype		28	ftype		24
moov		98276	moov		98280
	mvhd	108		mvhd	108
	trak	3493		trak	2311
	trak	2368		trak	498
	free	1978	🟥	🟥	🟥
	free	24549		free	29763
mdat		33437260	mdat		41449668

Table III. Two different MP4 files in two different devices

Canon T6i (MP4)			Samsung WB350F (MP4)		
Box Type		Size in Decimal	Box Type		Size in Decimal
ftype		28	ftype		28
moov		98276	moov		32394
	mvhd	108		mvhd	108
	trak	3493		trak	1269
	trak	2368		trak	1521
	free	1978		free	0
	free	24549		🟥	🟥
mdat		33437260	mdat		8783161

Afterwards, we also analyzed two different files from the same device. We analyzed two MP4 files from Canon T6i device. After analyzing the data structures carefully, we found the structures are similar to what we found in the previous analysis. For the same camera, many MP4 box types remain constant with respect to the files.

In table III, a comparison between two files for the MP-4 files is shown. For the two MP4 container files from different device shows significant difference than the same device. We found out the only ftype and mvhd were similar within the two files of the MP4 container from the different devices. Table 4 shows the comparison among two different MP4 file from two different devices.

AVI Files:

We tested different AVI file containers and found the files' binary data structures follow the general AVI structure, though different files represent data in different ways depending on the device. Table IV contains differences among three different AVI files from Samsung, Casio, and Nikon cameras. For the Samsung device, a variety of information does not show up in the data structures whereas Casio and Nikon have that information. For example, Samsung does not show the camera information and timestamps of the photo. But Nikon and Casio show the device information under the LIST info. Notice there are some differences between Casio and Nikon to represent the data--Casio stores the timestamp under the LIST 'hdrl' 'IDIT' (red color tab) chunk; while Nikon stores the timestamp under the LIST 'Info' chunk.

VI. Comparison of Edited Files with original

Four editing programs were used to change the original video file. During editing, we tried to insert different audio files or apply a different filter to the video files. Also, in one of the edited files, we inserted different audio files from the same device in the timeline of the video files. During the output of the edited files, we tried to match the metadata of the original file as much as possible. After finishing editing, we found most metadata were similar, but not all.

A comparison among some of the major edited video file metadata is shown in table V. In the table, green means the metadata matches with the original; whereas red means the metadata does not match. Notice that, among the measured metadata most of them matched with the original data. We found most metadata-conserving editing tool was Adobe Premiere, which kept the majority of the metadata information similar to the original file. After that, we compared the binary data structures among all the files and found many differences among them. We saw from the previous section that generally ftype of MP4 remains the same across the entire device. But we found for edited files they changed. Also, other box types were altered completely, so that one can easily find differences between the original file and the edited file. Additionally, in the movie data information for the original file, we find camera information, but none of the edited files contain that information.

Furthermore, we extracted some metadata information via ExifTool and found out that there are plenty of information about the camera were present in the original file whereas there is no information about the camera is found for the edited files. This includes: File Modification Date, Compressor Version, Orientation unit, Resolution Unit, Focal Length, Camera Temp., Time Zone, Model ID, Owner Name etc. Interestingly, for Adobe Premiere, ExifTool shows the output sequence of the edited files as 'adobe premiere tools' by which one can easily be confirmed that the files are manipulated by editing software. In table VI, the differences between the original and edited files are given. The red colors show that the data was absent for the respective files and dark orange represent the data has been changed from the original file

Table IV: Comparison of data structures among AVI file from three different devices

SAMSUNG				**CASIO**				**NIKON**				
RIFF	AVI			RIFF	AVI			RIFF	AVI			
LIST hdrl	avih			LIST hdrl	avih			LIST hdrl	avih			
	LIST strl	strh			LIST strl	strh				LIST strl	strh	
			vids				Vids					vids
			mjpg				Mjpg					mjpg
		strf				strf					strf	
			w:320				w:1280					w:640
			h:240				h:720					h:480
		strd	Zoran ME Corp			strd					strd	
	LIST strl				LIST strl					LIST strl		
		strh	auds			strh	Auds				strh	auds
		strf				strf					strf	
LIST movi						IDIT	3/20/2013 20:11					
idx1				LIST Info	CASIO EX-FC100			LIST Info	Nikon Corporation 2015:03:14 20:10:44			
				JUNK				JUNK				
				LIST movi				LIST movi				
				idx1				idx1				

Table V: Metadata comparison among original and edited files

Metadata	Original	Adobe	WMM	VLC	Corel
File Size	32 MB	32 MB	33 MB	32 MB	15 MB
File Type	MP4	MP4	MP4	MP4	MP4
File Permissions	rw-rw-rw-	rw-rw-rw-	rw-rw-rw-	rw-rw-rw-	rw-rw-rw-
MIMe type	video/mp4	video/mp4	video/mp4	video/mp4	video/mp4
Major brand	mp4 v2[ISO 14496-14]	mp4 v2[ISO 14496-14]	mp4 v2[ISO 14496-14]	MP4 Base Media v1 [IS0 14496-12:2003]	MP4 Base Media v1 [IS0 14496-12:2003]
Image Width/Height	1920/1080	1920/1080	1920/1080	1920/1080	1920/1080
Graphics Mode	srcCopy	srcCopy	srcCopy	srcCopy	srcCopy
X-Y Resolution	72	72	72	72	72
Bit Depth	24	24	24	24	24
Video Frame Rate	29.97	29.97	29.97	30.086	29.97

Video/Audio Format	AVC/mp4a	AVC/mp4a	AVC/mp4a	AVC/mp4a	AVC/mp4a
Media Time Scale	48000	48000	48000	1000000	48000
Audio BPS/channels	16/2	16/2	16/2	16/2	16/2
Audio Sample Rate	48000	48000	48000	48000	48000
Avg Bitrate	30.4	30.4	31.3	30.5	14.1 Mbps
MegaPixels	2.1	2.1	2.1	2.1	2.1
Bits/(Pixel*Frame)	0.484	0.484	0.481	0.482	0.223
Frame rate mode	Constant	Constant	Constant	Variable	Constant

Table VI: Difference Among original and edited files

Box Type		**Original**	**Adobe**	**Corel**	**WMM**	**VLC**
ftype		28	24	24	24	24
moov		98276	5543	5836	4423	8142
	mvhd	108	108	108	108	108
	trak	3493	2820	3343	2006	1969
	trak	2368	2543	2377	2233	6014
	free	1978	-	-	-	-
	free	24559	-	-	-	-
mdat		33437260	1	15518992	35066907	33005955

VII. Conclusion

This paper analyzed binary data structures in device video container formats from a forensic perspective. We analyzed AVI and MP4-like data structures and showed the differences among the data structures of the MP4 files among the same and different cameras. Also, we showed the main differences between the MP4 and MOV. Additionally, we analyzed different AVI files among different cameras and found out notable differences between devices, even though all the cameras stored files in the same general format. So, device files have a significant number of identifiable features in their binary data structures and metadata depending on their container format.

Another contribution of this paper is we presented a uniform technique to find out the differences between an original file and an edited file. There is a variety of information available inside the binary data structures and metadata of a file, but the most efficient way to find the differences is to analyze the parameters we mentioned in our results section. By analyzing the parameters listed in table VI one can find edited files easily. With the mentioned techniques one can find out the peculiarities in the edited video files very conveniently. Note that, although we did the analysis for one container format, similar parameters can be applied in any container format to find evidence of editing.

Although this paper tried to set a standard technique to analyze a file, much future work remains. In particular, we are planning to examine the distinctive video file format on a larger scale. We can examine more files to set a database for various types of camera model forensic analysis. Also, we did this analysis mostly manually. We are planning to implement an automatic tool which will check the new files parameters with the database files parameters to automatically test file authenticity. Additionally, our planned tool can be used to check major metadata information to identify the original camera details so that file authenticity of can be evaluated quickly.

References

[1] Farid H., “Exposing digital forgeries in scientific images”, Proc. 8th Workshop on Multimedia and Security (MM&Sec 2006), pp. 29–36, September 26–27, 2006.

[2] Li R., Li C.T., Guan Y., “Inference of a compact representation of sensor fingerprint for source camera identification”, Pattern Recognition, Volume 74, February 2018, pp 556-567.

[3] Sevinc B., H. Sencar, Nasir D. M., Ismail A., “Source camera identification based on CFA interpolation”, in Proc. IEEE Int. Conf. Image Processing, vol. 3, no., pp. III-69-72, 11-14, Sept. 2006.

[4] Yu-Feng H., Shih-Fu C., “Image Splicing Detection using Camera Response Function Consistency and Automatic Segmentation”, 2007 IEEE International Conference on Multimedia and Expo, Beijing, 2007, pp. 28-31.

[5] Jan Lukáš, Jessica Fridrich, and Miroslav Goljan "Detecting digital image forgeries using sensor pattern noise", Proc. SPIE 6072, Security, Steganography, and Watermarking of Multimedia Contents VIII, 60720Y (16 February 2006).

[6] Chih-Chung H., Tzu-Yi H., Chia-Wen L., Chiou-Ting H., “Video forgery detection using correlation of noise residue”, International Workshop on Multimedia Signal Processing, MMSP 2008, October 8-10, 2008.

[7] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," in IEEE Transactions on Signal Processing, vol. 53, no. 2, pp. 758-767, Feb. 2005.

[8] Fontani M., Milani S., Bestagini P., Barni M., Piva A., Tagliasacchi M, Tubaro S., “An overview on video forensics.” APSIPA Transactions Signal Information Process 2012.

[9] Gloe T., Fischer A., Kirchner M., “Forensic analysis of video formats.” Proc First Ann. DFRWS Eur, vol. 11, Supplement 1, no. 0, pp S68-S76, May 2014.

[10] Hall J.R., “Mpeg-4 video Authentication using file structure and Metadata,” MS Thesis, U. Colorado, 2015.

[11] Matrox Electronic Systems Ltd., OpenDML AVI File Format Extensions, Version 1.02.

[12] https://www.iso.org/standard/41828.html.

[13] https://developer.apple.com/standards/qtff-2001.pdf

Paper ID +organization	Full Name	Email address	Position	Research Interests	Personal website (if any)
P023 + UAF	Md Abir Hasan	abir.cuet08@gmail.com	others	Cybersecurity, Embedded Systems, IOT	
P023 + UAF	Dr. Orion Sky Lawlor	lawlor@alaska.edu	Assoc. Prof.	Computer graphics; parallel programming; robotics; 3D printing	https://www.cs.uaf.edu/~olawlor/
P023 +UAF	Nusrat Jahan	nusrat0802030@gmail.com	others	Cybersecurity, Embedded System, Power Systems	