infrastructures

MDPI

Article

Realising a Push Button Modality for Video-Based Forensics

Bako Zawali [1], Richard A. Ikuesan [2], Victor R. Kebande [3,4,*], Steven Furnell [5] and Arafat A-Dhaqm [6]

1 Cybersecurity Science Department, School of ICT, Federal University of Technology Minna, P.M.B 65 Minna, Nigeria; zawali.bako@st.futminna.edu.ng
2 Cyber and Network Security Department, Science and Technology Division, Community College of Qatar, Doha 00974, Qatar; richard.ikuesan@ccq.edu.qa
3 Department of Computer Science, Malmö University, SE-205 06 Malmö, Sweden
4 Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, 97187 Luleå, Sweden
5 School of Computer Science, University of Nottingham, Nottingham NG7 2RD, UK; steven.furnell@nottingham.ac.uk
6 School of Computer Science, Universiti Teknologi Malaysia, Johor Bahru 81310, Malaysia; mrarafat1@utm.my
* Correspondence: victor.kebande@mau.se or victor.kebande@ltu.se

Abstract: Complexity and sophistication among multimedia-based tools have made it easy for perpetrators to conduct digital crimes such as counterfeiting, modification, and alteration without being detected. It may not be easy to verify the integrity of video content that, for example, has been manipulated digitally. To address this perennial investigative challenge, this paper proposes the integration of a forensically sound push button forensic modality (PBFM) model for the investigation of the MP4 video file format as a step towards automated video forensic investigation. An open-source multimedia forensic tool was developed based on the proposed PBFM model. A comprehensive evaluation of the efficiency of the tool against file alteration showed that the tool was capable of identifying falsified files, which satisfied the underlying assertion of the PBFM model. Furthermore, the outcome can be used as a complementary process for enhancing the evidence admissibility of MP4 video for forensic investigation.

Keywords: multimedia forensics; push button forensics; file signature alteration technique

Citation: Zawali, B.; Ikuesan, R.A.; Kebande, V.R.; Furnell, S.; A-Dhaqm, A. Realising a Push Button Modality for Video-Based Forensics. *Infrastructures* **2021**, *6*, 54. https://doi.org/10.3390/infrastructures6040054

Academic Editor: Isam Shahrour

Received: 19 January 2021
Accepted: 28 March 2021
Published: 2 April 2021

1. Introduction

Information and communications technology (ICT) has taken over a substantial part of our lives and has brought about changes in our daily lives. Furthermore, the digital information that is stored in computers and multimedia devices is increasing, in particular multimedia content such as images, audio, and video. Video is one of the most significant groups of these multimedia data. However, as asserted in [1,2], the proliferation and ease of falsification of this class of multimedia data present a daunting challenge to society, thus further requiring the need for an advanced file fingerprinting mechanism [3,4]. Highlighting this notion, Reference [5] posited that the trustworthiness of a multimedia video is sacrosanct, the lack of a scientifically verifiable method notwithstanding. This challenge can be attributed to the complexity of editing software, which has also evolved to enable inexperienced users to manipulate the content of digital data (with little effort) with a high-quality output. As a consequence, questions regarding media authenticity are of growing significance, particularly in litigation where important decisions might be based on the reliability of the digital evidence [6]. A proper chain of custody, as well as a chain of evidence are also required to ensure the repeatability and possible expert presentation of a digital artefact [6–8].

The digital forensics discipline—the field saddled with the application of proven scientific methods to validate the reliability of digital artefacts [9,10]—has seen a steady growth in the number of professionals capable of extracting and verifying the authenticity

of multimedia data. Whilst this surge has been prevalent in developed countries where digital crimes are thoroughly investigated, the same cannot be said for developing nations. This is, however, conversely related to the reality of crime in the developing nations. Digital criminals tend to leverage the availability of state-of-the-art software and criminal networks to perpetuate seemingly sophisticated multimedia-related crimes. Therefore, a surge of fake multimedia content tends to dominate the cyber-ecosystem of most developing countries without a corresponding forensic/policing capability. Furthermore, the search for "better pay" in a seemingly "privileged" discipline has led to the migration of digital forensic experts from developing nations to advanced settings. Thus, the developing nations are left with a declining ratio of forensic experts to cyber criminals. A potential approach to this decreasing ratio is the integration of automation (a drive towards the bush button approach) in the forensic investigation process.

However, the notion of automation has been observed to further present diverse challenges, as asserted in [11], to include software and result verification challenges, the tendency of over-reliance on a tool (which could result in partial analysis), as well as the propensity to inhibit the soundness of the forensic process (given that the investigation process would be an art rather than a science, at best). These concerns can be summarized as the potential for a lack of reliability. Reliability in this regard involves both chain of custody and chain of evidence assurance. Attempts to address this automation challenge have been further asserted to provide a basis for departmental digital investigation workload reduction, to promote knowledge retention, and as a means towards forensic standardization and investigation coherence [11,12].

The remainder of this paper is organized as follows: In Section 2, the background of this study, as well as related studies are given. This is followed by the method used to realize the push button forensic concept, presented in Section 3. An implementation of the proposed approach is further presented, in Section 4 which contains the developed push button tool. Thereafter, a discussion is given in Section 5. The conclusion and future work of the study are given in Section 6.

2. Background and Related Literature

Multimedia Forensics (MF) is a digital forensics sub-domain that applies scientific techniques across a variety of digital content (audio, video, photo, etc.) for electronic discovery [13,14]. Like computer forensics [15], it involves the discovery of the source and/or location of multimedia data from their file metadata. Additionally, MF is tasked with the extraction of useful information for authentication and identification; for example, forgery detection, similarities between images, and the rate of accurate detection of multimedia facets. Image forensics plays a vital role in proving the authenticity and integrity of digital images by attempting to detect forgeries such as copy-move, copy-paste, region duplication, forged region, and region replacement within an image [16]. Audio forensic analysis, on the other hand, is the process of collecting, examining, and reviewing audio recordings to extract facts that are admissible during litigation by a court of competent jurisdiction [17]. Audio forensics has several applications that could be linked to the acoustic environment or location where the audio was recorded, the identification of speakers and audio improvements, or the actual device used to record the audio file. Similarly, video forensics aims to evaluate the authenticity and integrity of a moving image and integrated audio stream (video content) through the analysis of inherent device characteristics or processing artefacts in the video data [18]. Basically, MF focuses on source identification and forgery detection. Whilst source identification focuses on identifying or inferring knowledge about the source of digital information, forgery detection attempts to uncover traces of falsification by assessing the authenticity of the digital content [19]. MF is able to achieve this goal by relying on the extraction of facts and evidence to authenticate the integrity of digital data [20]. Videos are made by converting a camera's electrical impulses and saving the information as digital media. The number of still images per video time unit is referred to as the frame rate. Clips in digital videos use about 12–30 frame rates

per second, with 24 frames per second as the widely used frame rate (frame/s). The larger the number of frames, the smoother the video will appear. MP4 video, for instance, uses a sequence of pictures (discrete pixels) that can be continuously viewed to create the impression of motion, which manipulates the persistence of the perception of the human pictorial system [21].

These pixels can easily be represented by a number that uniquely identifies its overall value, which is easier for the computer to manipulate and store. Video falsification is a process of malicious modification of digital video content to obscure an entity or an event or change the meaning conveyed by the video; while video tampering detection aims to discover the traces of alteration and thereby evaluate the trustworthiness and integrity of the video file [22]. Insights into related studies on video forensics are further presented in the next paragraphs.

A large volume of research has proposed techniques and methods to confirm the trustworthiness and integrity of a digital video evidence. These techniques asserts that modifying the content of a video introduces specific artefacts that could be used for the alteration detection of a given video file. Detection techniques are classified as passive (blind) and active techniques [23]. According to [24], the availability of low-cost electronic multimedia devices and the high level of data processing capabilities have made video forensics increasingly important. Nevertheless, Reference [24] focused on discrepancies in video content using a human pictorial system through image resemblance measurement to find modifications in videos. This technique could readily detect alterations that are not noticeable to the human eye. The study in [25] reported that the accessibility of low-cost, portable, and highly usable digital multimedia devices has significantly increased the likelihood of location-less, network-related, or time-constrained digital multimedia. As a consequence, the authentication and verification of a given content have become increasingly difficult. The study further opined that this difficulty has several consequences when the digital content is used as a corroborating piece of evidence.

Similarly, the study in [26] proposed a video copy recognition system that is based on content fingerprinting that could also be used for the indexing and validation of video. The system uses a fingerprint extraction algorithm combined with a fast and approximate search algorithm to extract the compact content-based signatures from separate images of the video frames. Each of such images represents a short segment of the video and contains temporal, as well as spatial information about the video segment. The system extracts and pre-stores fingerprints of all the videos stored in the database. However, this approach only works for video with a very short length, thus making the approach inefficient for forensic investigation purposes. By limiting the investigation process to frame removals only, the study in [27] proposed a collection of automated frame removal or additional recognition techniques that considered changes in the P-frame prediction error of a video. This technique focuses on video codecs using a fixed-length group of pictures (GOPs) when compressing segmented frames in a video. Moreover, the result is only reliable if anti-forensics have not been applied to the video content. Leveraging the signal processing methodology, the studies in [25,28] inspected the effective approaches to reconstruct and authenticate the processing history of video data. The study asserted that most alterations are not revocable and leave some "footprints" in the reconstructed signals, which can be analysed to recognize the previous processing steps. However, empirical evaluation has shown that simple processing chains of a signal can be reconstructed with a negligible amount of modification to the signal, rendering the approach inefficient to check the footprint of the video content.

In an attempt to introduce an automation process (referred to as push button forensics), the study in [18] developed a system that explored the video stream of digital cameras and mobile phones in order to extract the file format structures. Upon successful extraction, the system then validated the structure with the original video file. Captured information included the origin of the file, recognizing the true device of the acquisition model, and the processing software that was used for the recording. Furthermore, it required an adapted

file parser(s) to read and extract all obtainable file formats and metadata from the videos in the database created. This approach is a passive technique of detecting alterations in videos. The tendency to store all models or vendor-specific peculiarities of digital devices used for creating video content was a major limitation of the study. A similar study in [22] established a method for perceiving suspicious areas using noise characteristics in static scene videos (surveillance).

A noise level function (NLF) describes the variance in image signals of the irradiance-dependent noise. The study used a probabilistic design approach, which regulated the noise characteristics at each pixel. Pixels in spliced areas were separated using the posterior maximum (MAP) estimation of the noise model where the NLFs were incompatible with the rest of the image. However, the study did not account for frame structures failure when the repeated frames were less than the calculated window size, especially when frame replication took place in a different order. Reference [29] also developed the VidentifierTM (VTM) Forensic system for automatically recognizing the modification of images and videos. VTM Forensic has two main features that are of interest to the multimedia community. First, it has a robust structure, precisely distinguishing difficult video alterations. Secondly, it is efficient, even on a very large scale. To recognize video modifications, VTM Forensic uses a mixture of a large-scale multidimensional NV-tree index and fine-grained local image descriptors. VTM Forensic is tolerant of many pictorial changes, including mirroring, camrips, compression, and subtitles. It, however, requires that the fingerprints of the authentic versions of the videos be stored in the database for assessment. The feasibility of creating a valid database for all original versions of video files cannot be ensured during a digital investigation. These studies attempted to develop viable alternatives for video forensics, albeit with inherent limitations. Furthermore, the forensic soundness of the push button forensic modality (PBFM) tools developed was ignored. To address these observed limitations, the current research proposed a forensically sound push button forensic (PBFM) tool for the investigation of the MP4 video file format. The file format selection hinged on a limited number of potential video file formats and the possibility of an exhaustive video file format integration.

3. Realising Push Button Forensics

As a step towards addressing this forensic reliability challenge, this study sought to promote the development of an automated video forensics process through a push button forensics modality (PBFM). The term PBFM is used to connote a forensically sound process implemented in a tool for conducting digital investigation. This process mainly includes corroborative evidence collection and pre-processing, as well as potential evidence analysis. A typical PBFM process defined for this study is further illustrated in Figure 1. Central to this illustration is the assurance of chain of custody and chain of evidence through a white-box testing approach. The decision to ascertain these attributes was considered essential for evidence admissibility and standardized forensic practice. Consequently, this process can potentially "reduce the case backlog while avoiding investigation biases and personal prejudice" [11]. Furthermore, the process considers the verification of the analysis methodology. In this regard, a formal approach that entails theoretical suppositions and logical reasoning can be used to substantiate the correctness of the analysis process.

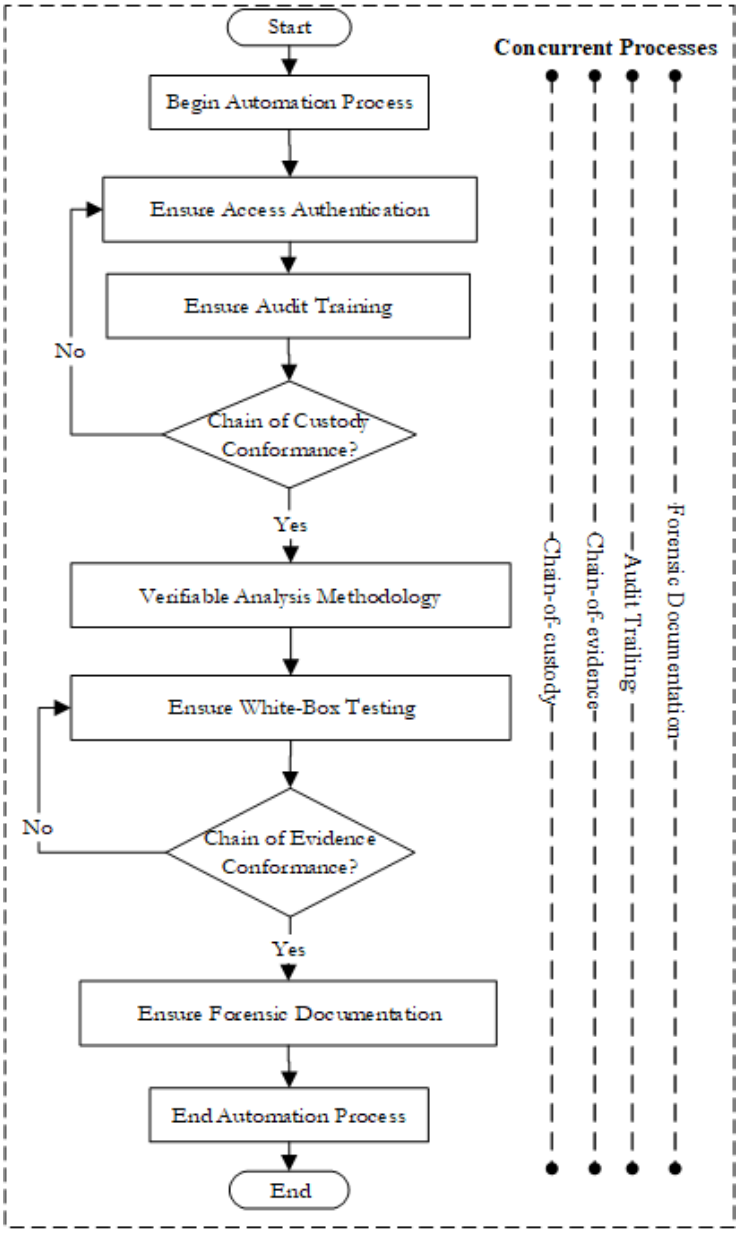

Figure 1. Push button forensics reliability assurance process.

3.1. Implementation Approaches

This section discusses the proposed PBFM approach, which is based on a file signature alteration technique that is capable of identifying potential modifications. A multi-tier architecture (n-tier), which is comprised of the presentation, logical, and data tiers, as shown in Figure 2, was adopted in this study. This architecture physically separates the application logic processing, data management roles, and presentation activities.

The presentation tier, as shown in Figure 2, which is the uppermost layer of the application, interconnects with the other two tiers of the application. It shows the data of the administered output content and the layer that users access through the graphical user interface (GUI). The logic tier (application tier, middle tier, or business logic) receives input from the presentation tier. It processes and controls the functionalities of the proposed application. On the other hand, the data storage tier is the data access layer that encapsulates the persistence mechanisms and exposes the data to the application tier. The storage mechanism allows for updates or changes without affecting the application tier clients.

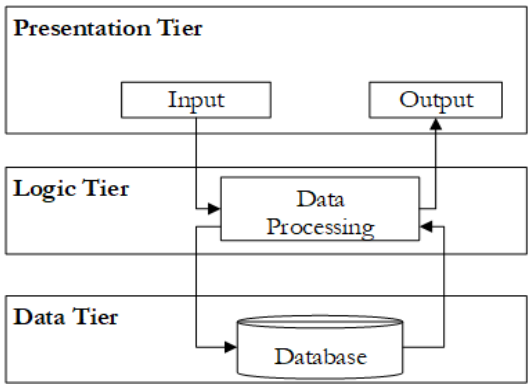

Figure 2. Multi-tier architecture.

3.2. Operational Framework

One core component of an automated forensics process, as highlighted in Figure 1, is the capability to ensure white-box testing. The combination of the tiered architecture and the process presented in Figure 3 was conceived of to address this focus. This further ensures that the software information domain and its component functions are fully understood, as are its behaviour, performance, and the interfaces required. An imputed Mp4 Video file is parsed for file signature identification and extraction. The extraction signature is then compared with a known signature. The report of this verification process is further hashed to ensure integrity verification. These are further explained in the following subsections.

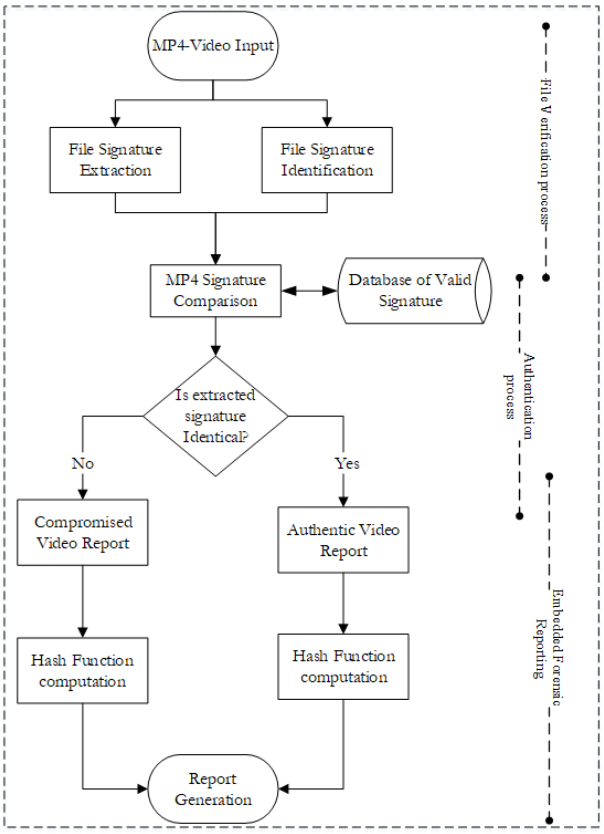

Figure 3. Operational framework.

3.2.1. Signature Extraction Mechanism

Tracing the forgery of a video file requires the identification of several parameters of the video file. Mostly, it becomes paramount for one to be able to prove that the file is a video file or whether there exists any enhancements or illegal processing using video processing tools. Figure 3 shows the flow of the video selection process. The file selection process identifies video-type media files, which is further verified to ascertain if the imputed sample is an accurate file format. The format in this context denotes the standard extension that identifies the file type. Upon successful identification of the desired file format, the data is parsed to the file signature extraction and comparison phase. Identification and extraction of the signature of a file comprise a forensic scheme that is performed at the preliminary stages, which allows one to match the content of the file against what may be residing in the database. A rather interesting concept in this approach is to identify the accidental or deliberate distortions or tampering. The file signature, in the context of this research, is identified as a key feature of video file. This is based on the supposition that each video file format has a specific frame content that corresponds to a unique hexadecimal value, otherwise known as its file signature. Known file signatures can, therefore, be stored in the repository. In the authors' in-depth analysis, distortions of the video signatures may arbitrarily hinder successful forensic discoveries. The theoretical basis and the corresponding algorithm for the signature extraction and comparison are presented in Section 3.2.2. Upon a successful match (or the converse), a corresponding forensic report is generated.

3.2.2. File Signature Identification and Extraction from MP4 Video

The MP4 video format (MPEG-4 Section 14, also known as MPEG-4 AVC, where AVC denotes Advanced Video Coding and MPEG refers to Motion Picture Expert Group) is one of the most common digital multimedia formats for storing video and audio. However, it can also be used to store other data such as subtitles and still images. The official file name extension for MPEG-4 Part 14 files is ".mp4", other extensions, most commonly ".m4a" and ".m4p", notwithstanding. MP4 is based on the ISO/IEC 14496-12:2004 standard, which in turn is based on the QuickTime file format. Its structure is similar to the QuickTime file format, with some additional features. An MP4 file has three sections: header (ftyp), video data (mdat), index information (moov), as shown in Figure 4. Furthermore, the MP4 format also consists of consecutive chunks. Each chunk of MP4 files includes an 8 byte header, a 4 byte chunk size (high byte first, big-endian), and a 4 byte chunk type. The hexadecimal composition of these chunks is further depicted in Figure 5. The first chunk of an MP4 file has a four byte chunk size at offset zero and a four byte chunk type.

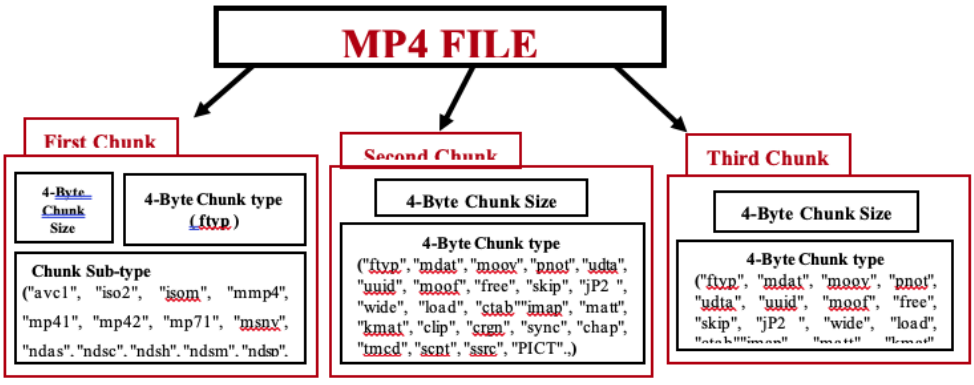

Figure 4. MP4 file structure.

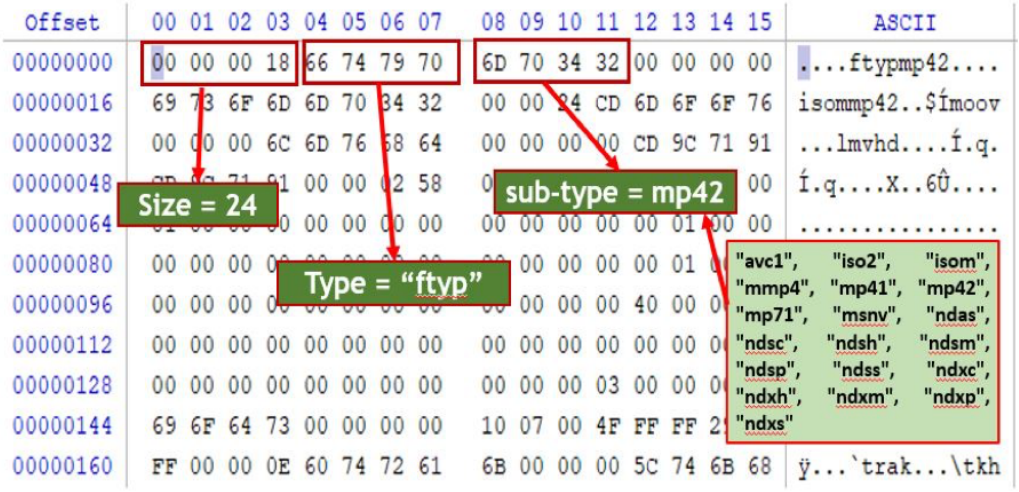

Figure 5. First chunk of a sample MP4 file.

From Figure 5, the offset locations 00 through 03 represent the size in a decimal value of the first chunk header. To extract the file signature, the hexadecimal values 00 00 00 18 are converted to decimal values, which correspond to 00 00 00 24. This is the size of the first chunk header in the sample MP4 file shown in Figure 4. The offset locations 04 through 07 represent the signature type (66 74 79 70) of the first chunk header of an MP4 file. These hexadecimal values are converted to ASCII values to obtain "ftyp". ftyp represents the first file signature type for every MP4 file, with details further depicted in Table 1.

Table 1. Signature type of MP4 files.

Hexadecimal	Decimal	ASCII
66 74 79 70	102 116 121 112	ftyp

Furthermore, the first chunk of every MP4 file also has a signature sub-type defined at offset locations 08 through 11. When these hexadecimal values are converted to their corresponding ASCII values, one of the following signature sub-types is obtained: "avc1", "iso2", "isom", "mmp4", "mp41", "mp42", "mp71", "msnv", "ndas", "ndsc", "ndsh", "ndsm", "ndsp", "ndss", "ndxc", "ndxh", "ndxm", "ndxp", "ndxs".

A summary of these signature sub-types is further presented in Table 2. The offset location of the second chunk starts at the size defined by the first chunk. It has a four byte chunk size and a four byte chunk type. A breakdown of the file signature subtype using the second chunk is also considered. This is further illustrated in Figure 6.

From Figure 6, offset locations 00 through 03 (00 00 00 18) represent the size (24 in decimals) of the first chunk header. Thus, offset location 24 marks the start point of the second chunk. From the sampled MP4 file shown in Figure 6, offset locations 24 through 27 (00 00 24 CD) represent the size (9421 in decimals) of the second chunk. The next four bytes after the size represent the file signature type of the second chunk. These file signature types could be any of the types shown in Table 3.

Table 2. Signature sub-type of MP4 files.

S/N	ASCII	Hexadecimal	Decimal
1	avc1	61 76 63 31	097 118 099 049
2	iso2	69 73 6F 32	105 115 111 050
3	isom	69 73 6F 6D	105 115 111 109
4	mmp4	6D 6D 70 34	109 109 112 052
5	mp41	6D 70 34 31	109 112 052 049
6	mp42	6D 70 34 32	109 112 052 050
7	mp71	6D 70 37 31	109 112 055 049
8	msnv	6D 73 6E 76	109 115 110 118
9	ndas	6E 64 61 73	110 100 097 115
10	ndsc	6E 64 73 63	110 100 115 099
11	ndsh	6E 64 73 68	110 100 115 104
12	ndsm	6E 64 73 6D	110 100 115 109
13	ndsp	6E 64 73 70	110 100 115 112
14	ndss	6E 64 73 73	110 100 115 115
15	ndxc	6E 64 78 63	110 100 120 099
16	ndxh	6E 64 78 68	110 100 120 104
17	ndxm	6E 64 78 6D	110 100 120 109
18	ndxp	6E 64 78 70	110 100 120 112
19	ndxs	6E 64 78 73	110 100 120 115

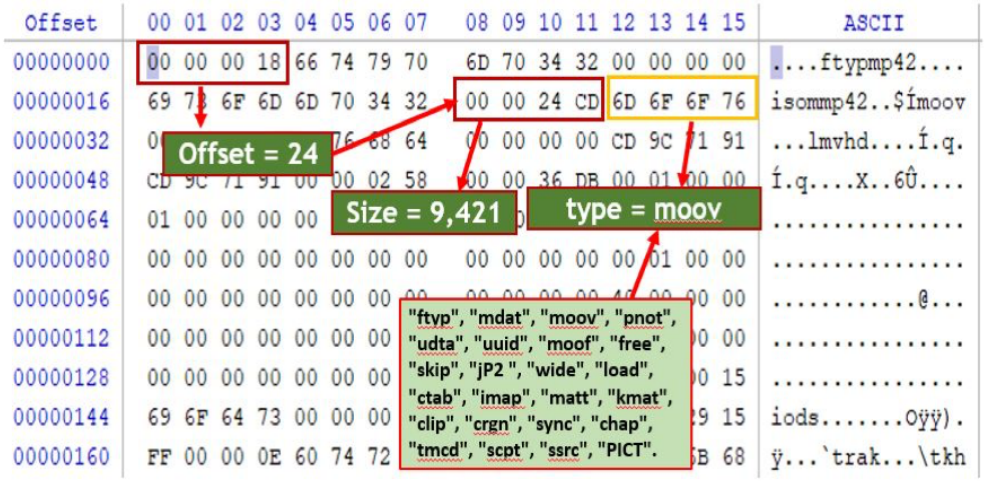

Figure 6. Second chunk of the sample MP4 file structure.

The offset locations 16 through 19 and 20 through 23 are also considered a file signature sub-type, which could be defined by any of the signatures in the MP4 signature sub-type shown in Table 3.

Table 3. MP4 signature types.

S/N	ASCII	Hexadecimal	Decimal
1	ftyp	66 74 79 70	102 116 121 112
2	mdat	6D 64 61 74	109 100 097 116
3	moov	6D 6F 6F 76	109 111 111 118
4	pnot	70 6E 6F 74	112 110 111 116
5	udta	75 64 74 61	117 100 116 097
6	uuid	75 75 69 64	117 117 105 100
7	moof	6D 6F 6F 66	109 111 111 102
8	free	66 72 65 65	102 114 101 101
9	skip	73 6B 69 70	115 107 105 112
10	jP2	6A 50 32	106 080 050
11	wide	77 69 64 65	119 105 100 101
12	load	6C 6F 61 64	108 111 097 100
13	ctab	63 74 61 62	099 116 097 098
14	imap	69 6D 61 70	105 109 097 112
15	matt	6D 61 74 74	109 097 116 116
16	kmat	6B 6D 61 74	107 109 097 116
17	clip	63 6C 69 70	099 108 105 112
18	crgn	63 72 67 6E	099 114 103 110
19	sync	73 79 6E 63	115 121 110 099
20	chap	63 68 61 70	099 104 097 112
21	tmcd	74 6D 63 64	116 109 099 100
22	scpt	73 63 70 74	115 099 112 116
23	ssrc	73 73 72 63	115 115 114 099
24	PICT	50 49 43 54	080 073 067 084

To determine the offset location of the third chunk for the MP4 sample file shown in Figures 5 and 6, the size of the first chunk is added to the size of the second chunk. In this case, the first and second chunk sizes are 24 and 9421 (both in decimals), respectively. Thus, the offset location of the third chunk is 24 + 9421 = 9445. Therefore, the offset locations 9445 through 9448 represent the size of the third chunk, while the offset locations 9449 through 9452 represent the file signature type of the chunk as defined in the file signature type table shown in Table 3. A summary depiction of the third chunk is shown in Figure 7.

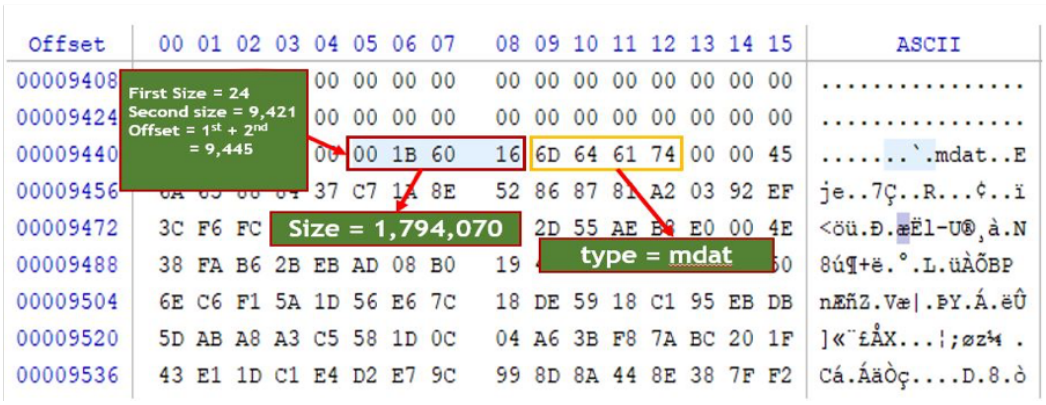

Figure 7. Third chunk of a sample MP4 file.

3.3. Comparison Lookup Table

To aid the comparison process, this study developed a lookup table, which was then used to compute the authentication process of the file format structure and file signature based on the alteration differences typically observed between an altered file and its original version. A synopsis of the lookup sequence is further presented in Table 4. The sequence is an integration of the defined file signature sub-types for chunks 1, 2, and 3.

Table 4. Signature comparison lookup table.

S/N	Chunk	Offset	Size	Type	Sub-type
1	Chunk 1	0	size-1	ftyp	Table 3
2	Offset-x	16–19		Table 2	NIL
3	Offset-y	20–23		Table 2	NIL
4	Chunk 2	size-1	size-2	Table 2	NIL
5	Chunk 3	size (1 + 2)	size-3	Table 2	NIL

The sequence synopsis given in Table 4 is further described as follows:

- Chunk: the chunk header information represents the order in which the chunk appears in the file, where chunk 1, chunk 2, and chunk 3 depict the first, second, and third chunks of the examined MP4 file, respectively.
- Offset: an offset depicts the distance from the beginning of the file (when viewed in Hex format) where "0" is the index position zero and "size-1" is the index position of the first size of the chunk observed in the examined MP4 file.
- Size: the size depicts the computed decimal value equivalent of the chunk's size, which consists of four octets.
- Type: this depicts a predefined signature used for the identification of each chunk.
- Sub-type: this depicts a predefined signature for identifying the sub-chunks of every chunk within the examined MP4 file.

4. Push Button Forensic Tool

A web-based video forensic tool, which provides the basic forensic functionality that is required for video forensic investigation, is presented in this section. Based on common practice in forensic examination, the developed system provides basic functionalities such as determining the video file format; identifying whether the video file has been manipulated or not; and identifying the manipulation/alteration techniques. The corresponding interface of the developed tool is shown in Figure 8 (the tool is available at https://github.com/mrzee498/Multimedia-Forensicator (accessed on 5 March 2021)). The tool uses a lightweight database as the storage location where the comparison mechanism gets a stored set of file signatures of the existing video format. Based on the various actions highlighted as the processes involving data input, different tables were designed to help store the information needed for such actions.

The user (forensic investigator) starts by choosing the type of multimedia file (in this case video) to be investigated. After a successful upload of the multimedia file, the user then performs the analysis by "pushing" the "Forensic Analysis" "button" shown in Figure 8. The result of the verification/authentication process is then displayed. The interpretation of the result is presented in Table 5.

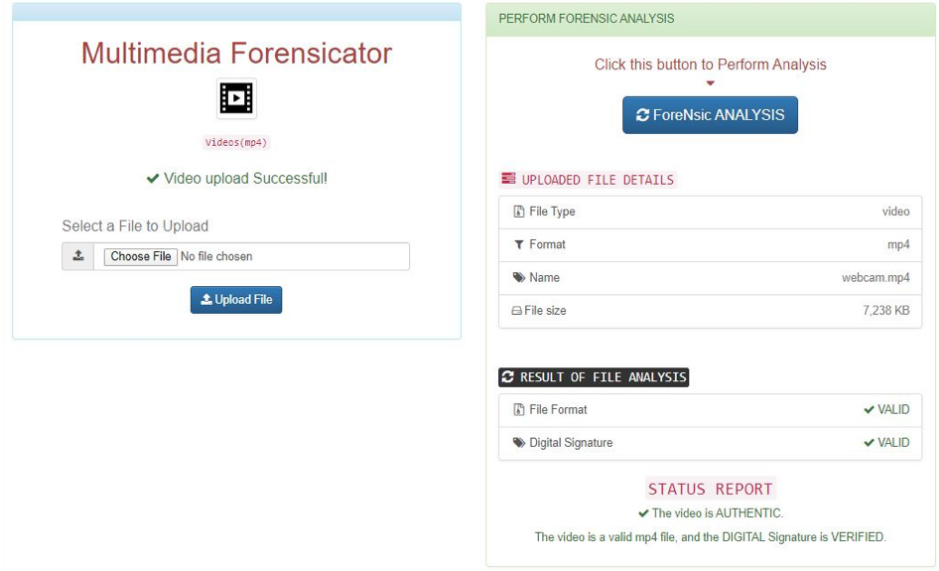

Figure 8. Result of analysis of MP4 file using the PBF tool.

Table 5. Description of the interpretation of the output of the tool.

S/N	Attribute	Description
1	File type	specifies the type of multimedia file that is being investigated; in our case, video
2	Format	specifies the file format of the multimedia file that was selected from the file type.
3	Name	shows the name of the file as stored in the source location
4	Size	this is the size of the uploaded mp4video file in kilobytes
5	Status Report	shows whether its integrity has been compromised or not

5. Discussion

The increasing rate of multimedia devices and the demand for digital data call for a scientific and forensically proven approach to verify the authenticity of digital evidence presented in video content. This is necessary because forensic analysis plays an important role in criminal investigations and civil litigation cases when administered in a court of law. Forensic practitioners have researched several techniques and methods to verify the trustworthiness and integrity of the content of a digital video [23]. Some of the tools developed use different alteration detection techniques to authenticate digital video content, and examples of such tools include: the VidentifierTM (VTM) Forensic system for automatically identifying modification in images and videos, developed by Asmundsson and Lejsek [29]. The method used by VTM is a fingerprint-based extraction unit associated with the database server where the geometric and time-based properties of the extracted fingerprints are stored. The system requires that the fingerprints of the authentic videos be obtained and stored in the database to be used for assessing videos under investigation. However, a priori knowledge of digital content may not be available for forensic investigation. Therefore, this proves a limitation in the VTM forensic system given that obtaining the authentic version of all available video files may prove challenging.

Secondly, the study in [25] developed an approach to reconstruct and authenticate the processing history of video data. The study assumed that alterations are not revocable and that they leave some evidence in the reconstructed signals, which can be analysed to

recognize the previous processing steps. However, according to [28], simple processing chains of a signal can be reconstructed without adding an excessive amount of modification to the signal, thus rendering the approach inefficient to check the integrity of the video content. The study in [27] proposed a collection of automated frame removals. This system achieved video authentication with a mathematical model of video frame removal and accumulation discovery techniques aimed at video codecs using a fixed length group of pictures (GOP) when compressing segmented video frames. The limitation of the outcome is that the method is only reliable if anti-forensic techniques have not been applied to the video content. Similarly, the study in [18] developed a passive technique of alteration detection that explores video streams and extracts file format structures of the videos from multimedia devices. The approach is based on recognizing the true device of the acquisition model or the processing software that was used for the recording and required adapted file parser(s) to read and extract all obtainable file formats' data and metadata from videos. Detecting video file structure information based on camera and mobile phone model specifics may not be effective in the future, because determining all models or vendor-specific peculiarities of digital devices used for creating video content is challenging.

Lastly, the study in [30] proposed an algorithm for detecting frame deletion in HEVC-coded videos that were classified by machine learning classifiers. The research employed the passive alteration detection technique of multimedia forensic methods. Results from the study revealed that learning-based classifiers were more efficient than model-based ones. Furthermore, the system had a limitation in forgery detection capabilities when the number of deleted frames doubled the number of groups of pictures (GOPs) in digital videos. This implies that videos falsified by experienced anti-forensic individuals will lure the system to a false negative report. A descriptive summary of these tools is further presented in Table 6.

While several tools have been developed with different alteration detection techniques, as reflected in the comparative analysis in Table 6, the file signature method of the active authentication technique as considered in this study presents a white-box paradigm. Authentication is used loosely, in this regard, to refer to the process of identifying and validating the file signature of the given MP4 file using the baseline Hex structure as the signature. This technique is considered the most effective of all the approaches examined [22]. It involves the process of extracting the unique digital structure of the file signature embedded at the point when video files are created. Moreover, altering the file in any manner deteriorates the embedded signature [21]. Furthermore, the white-box paradigm ensures that the reliability of the forensic process can be evaluated. This study did not presume any prior knowledge of the authentic versions of the video files under investigation, unlike other studies that attempted to extract fingerprints from the original versions of the videos. Furthermore, this study did not rely on the architectural structures of the file formats only or the acquisition devices, because anti-forensic techniques can falsify the structure and source of digital content, as proven in previous studies.

These studies applied diverse video alteration techniques, which can be summarized as frame insertion, frame deletion, frame and header substitution, metadata alteration, and header information alteration. Furthermore, video editing tools such as Hex editor, Adobe Premiere Pro (for timeline-based video editing), Freemaker video converter, Windows movie maker, EZGif, movie maker, Corel VideoStudio Ultimate, Magix Movie Edit Pro, OpenShot (available at https://www.openshot.org/ (accessed on 5 March 2021)), Atube catcher, Camtasia studio, and Adobe Spark are potential tools that can be explored to falsify video content. Recent advances that explore the deep learning approach in image alteration are also applicable. To further evaluate the effectiveness of the developed tool, Adobe Premiere Pro, Atube catcher, and Windows movie maker were used to perform MP4 file alteration. Three MP4 files were altered and then verified using the developed tool. The result, as shown in Table 7, supports the theoretical supposition presented in this study.

Table 6. Comparative analysis MP4 file forensic analysis.

Video Forensic Tool	Approach	Results	Limitation	Analysis
Proposed	Active detection technique using the file signature method.	Efficient and works accurately for both large and small size videos in MP4 format. Detects forgeries irrespective of the alteration technique.	It works only for MP4 video files	Although MP4 is a widely used format for presenting videos, it is not the only format available. The system is accurate only when the source devices use the MP4 format at the point of creation.
[29]	Finger-print-based extraction.	Efficient at a very large scale and accurately detects difficult video transformations.	The system requires prior knowledge of the original version of the digital content under investigation.	Obtaining the original versions of digital videos for investigation is quite challenging. An effective forensic tool should confirm the integrity of digital content without any prior knowledge of the original file.
[25]	Employed both forensics and video surveillance techniques to extract from the multi-camera system the trajectories of moving people to create a video forensic authentication tool.	Effective in retrieving people snapshot and trajectory information that could be of interest during the investigative process.	The research reconstructed processing chains of signals with the assumption that no excessive amount of modifications was made.	Reconstructing processing chains of signals leave some forensically detectable traces in the reconstructed signals, which can be analysed to render the data forensically contaminated.
[27]	Frame forgery detection of the multimedia passive authentication technique.	The developed system can automatically detect video forgeries with a high degree of accuracy if anti-forensics is not used.	The system only detects forgeries that involve frame deletion and addition techniques.	There are other numerous anti-forensic techniques (such as region shuffling, frame duplication, camera source forgery, etc.) that can be applied to digital videos aside frame deletion and addition to fool a forensic investigation.
[21]	Inter-frame forgery detection techniques using tamper traces from spatio-temporal and compressed domains.	Successful at forgery detection in frame shuffling, frame insertion, frame deletion, and frame duplication.	The system employs only the passive method of multimedia forensic analysis.	The system fails to authenticate digital videos when other anti-forensic techniques are applied.
[30]	Algorithm for detecting frame deletion in HEVC-coded videos, which are classified by machine learning classifiers.	Employed the passive alteration detection technique of multimedia forensic methods.	Findings from the research illustrate that learning-based classifiers are more robust and reliable than model-based classifiers.	The system remains robust only for in-frame deletion scenarios involving static scenes and effortlessly manipulated videos. While the system is only limited to frame deletion situations, anti-forensic methods could fool the system and render the authentication process abortive if other alteration techniques are employed.

The verification process presented in Table 7 shows that signature mismatch can be used to distinguish altered files irrespective of the alteration techniques applied. This study therefore presented the background for a reliable approach towards a PBFM platform. Such a platform is essential to address the growing deficit of skill shortage in developing nations. It is needless to highlight that the exodus of forensic experts from most developing nations, as well as the corresponding lack of competent forensic examiners could pose a consequential challenge to the global forensics community. The proposed tool, however, provides a fundamental basis for the admissibility and reliability of forensic artefacts, more specifically, complying with the reliability assurance process stated in Figure 1. Furthermore, there is a constant need to incorporate cost-savings mechanisms (forensic readiness) when it comes to digital forensics, which in the context of this study may be useful to an organization. This basically allows incidental planning as a solution of getting evidence when needed in order to reconstruct an event [31,32]. Additionally, the automation process of the tool ensures that every action taken by the user while using the tool is logged, and the resultant output of the analysis is carefully documented with a corresponding hash digest for both the logs and the analysis result. Through this, the

result of the automation process can be verified by another examiner, when required, as asserted in [11,33].

Table 7. Tool validation using off-the-shelf file alteration tools.

S/N	Alteration Tool	Applied Technique	Obtained Result
1	Windows Movie Maker	Frame deletion	• **File Format**: VALID • **Digital Signature**: NOT VALID • **STATUS Report**: Some frames were removed from the original copy
2	Windows Movie Maker	Frame Replacement	• **File Format**: VALID • **Digital Signature**: NOT VALID • **STATUS Report**: The video is a valid mp4 file, but the DIGITALsignature is not verified
3	Adobe Premier Pro	Copy Paste Frames	• **File Format**: VALID • **Digital Signature**: NOT VALID • **STATUS Report**: The video is a valid mp4 file, but the DIGITAL signature is not verified
4	Adobe Premier Pro	Copy Paste Frames	• **File Format**: VALID • **Digital Signature**: NOT VALID • **STATUS Report**: The video is a valid mp4 file, but the DIGITAL signature is not verified
5	Atube Catcher	Frame Conversion	• **File Format**: VALID • **Digital Signature**: NOT VALID • **STATUS Report**: The frames are converted from another format
6	Atube Catcher	Frame Compression	• **File Format**: VALID • **Digital Signature**: NOT VALID • **STATUS Report**: The video is a valid mp4 file, but the DIGITAL signature is not verified

6. Conclusions and Future Works

This study presented a technique for verifying MP4 video data integrity by authenticating the embedded digital signature. It also showed that the authentication of digital data is not strictly based on complex mathematics and algorithms. A video file can be authenticated by understanding the file structures and decoding the embedded digital signature at the point of creation. This research work presented a method for authenticating MP4 videos by creating a lookup table for the architectural structure and composition of the content. The developed system is a useful tool for digital investigations that will provide a simple user interface for multimedia forensics investigators. While this study did not provide an exhaustive lookup table of all possible video file formats, further study is currently under way to include all potentially available video file formats, as well as other multimedia file types in the developed tool. The tools was further conceptualized to provide a baseline for the development of push button forensics capable of enhancing forensics investigation in developing nations. Future work will also include other video file formats and the use of combined alteration detection techniques to make the tool more robust and sophisticated to span across various anti-forensic techniques.

Author Contributions: Conceptualization, R.A.I.; methodology, R.A.I., B.Z., and V.R.K.; software, B.Z.; validation, S.F. and V.R.K.; formal analysis, B.Z., R.A.I., V.R.K., S.F., and A.A.-D.; investigation, B.Z., R.A.I., V.R.K., S.F., and A.A.-D.; resources, B.Z.; data curation, B.Z., R.A.I., V.R.K., S.F., and A.A.-D.; writing—original draft preparation, B.Z., R.A.I., V.R.K., S.F., and A.A.-D.; writing—review and editing, B.Z., R.A.I., V.R.K., S.F., and A.A.-D.; visualization, B.Z., R.A.I., V.R.K., S.F., and A.A.-D.; supervision, S.F.; project administration, R.A.I.; funding acquisition, V.R.K. All authors read and agreed to the published version of the manuscript.

Funding: The APC was funded by Malmö University, Sweden.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used are available upon request; however, the developed tool is available at https://github.com/mrzee498/Multimedia-Forensicator (accessed on 5 March 2021).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; nor in the decision to publish the results.

References

1. Rocha, A.; Scheirer, W.; Boult, T.; Goldenstein, S. Vision of the unseen: Current trends and challenges in digital image and video forensics. *ACM Comput. Surv. (CSUR)* **2011**, *43*, 1–42. [CrossRef]
2. Azhan, N. Analysis of DQT and DHT in JPEG Files. *Int. J. Inf. Technol. Comput. Sci. (IJITCS)* **2013**, *10*, 1–11.
3. Adeyemi, I.R.; Abd Razak, S.; Azhan, N.A.N. A review of current research in network forensic analysis. *Int. J. Digit. Crime Forensics (IJDCF)* **2013**, *5*, 1–26. [CrossRef]
4. Kebande, V.R.; Mudau, P.; Ikuesan, R.A.; Venter, H.; Choo, K.K.R. Holistic Digital Forensic Readiness Framework for IoT-Enabled Organizations. *Forensic Sci. Int. Rep.* **2020**, 100117. [CrossRef]
5. Al-Sanjary, O.I.; Ahmed, A.A.; Sulong, G. Development of a video tampering dataset for forensic investigation. *Forensic Sci. Int.* **2016**, *266*, 565–572. [CrossRef]
6. Ikuesan, A.R.; Venter, H.S. Digital behavioural-fingerprint for user attribution in digital forensics: Are we there yet? *Digit. Investig.* **2019**, *30*, 73–89. [CrossRef]
7. Kebande, V.R.; Ikuesan, R.A.; Karie, N.M.; Alawadi, S.; Choo, K.K.R.; Al-Dhaqm, A. Quantifying the need for Supervised Machine Learning in Conducting Live Forensic Analysis of Emergent Configurations (ECO) in IoT Environments. *Forensic Sci. Int. Rep.* **2020**, *2*, 100122. [CrossRef]
8. Baror, S.O.; Venter, H.S.; Adeyemi, R. A natural human language framework for digital forensic readiness in the public cloud. *Aust. J. Forensic Sci.* **2020**, 1–26. [CrossRef]
9. Makura, S.M.; Venter, H.; Ikuesan, R.A.; Kebande, V.R.; Karie, N.M. Proactive Forensics: Keystroke Logging from the Cloud as Potential Digital Evidence for Forensic Readiness Purposes. In Proceedings of the 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), Doha, Qatar, 2–5 February 2020; pp. 200–205.
10. Baror, S.O.; Ikuesan, R.A.; Venter, H.S. A Defined Digital Forensic Criteria for Cybercrime Reporting. In *International Conference on Cyber Warfare and Security*; Academic Conferences International Limited: Reading South Oxfordshire, UK, 2020; p. 617-XVIII.
11. James, J.I.; Gladyshev, P. Challenges with automation in digital forensic investigations. *arXiv* **2013**, arXiv:1303.4498.
12. Ikuesan, R.A.; Ganiyu, S.O.; Majigi, M.U.; Opaluwa, Y.D.; Venter, H.S. Practical Approach to Urban Crime Prevention in Developing Nations. In Proceedings of the 3rd International Conference on Networking, Information Systems & Security, New York, NY, USA, 30 March–2 April 2020; Association for Computing Machinery: New York, NY, USA, 2020. [CrossRef]
13. Battiato, S.; Giudice, O.; Paratore, A. Multimedia forensics: Discovering the history of multimedia contents. In Proceedings of the 17th International Conference on Computer Systems and Technologies 2016, Palermo, Italy, 23–24 June 2016; pp. 5–16.
14. Khan, M.K.; Zakariah, M.; Malik, H.; Choo, K.K.R. A novel audio forensic data-set for digital multimedia forensics. *Aust. J. Forensic Sci.* **2018**, *50*, 525–542. [CrossRef]
15. Karie, N.M.; Kebande, V.R.; Venter, H. Diverging deep learning cognitive computing techniques into cyber forensics. *Forensic Sci. Int. Synerg.* **2019**, *1*, 61–67. [CrossRef]
16. Malviya, A.V.; Ladhake, S.A. Pixel based image forensic technique for copy-move forgery detection using auto color correlogram. *Procedia Comput. Sci.* **2016**, *79*, 383–390. [CrossRef]
17. Zakariah, M.; Khan, M.K.; Malik, H. Digital multimedia audio forensics: Past, present and future. *Multimed. Tools Appl.* **2018**, *77*, 1009–1040. [CrossRef]
18. Gloe, T.; Fischer, A.; Kirchner, M. Forensic analysis of video file formats. *Digit. Investig.* **2014**, *11*, S68–S76. [CrossRef]
19. Talmale, G.; Talhan, A.; Dharaskar, R.V. Analysis of multimedia forensic technique. In Proceedings of the 2011 International Conference on Communication, Computing & Security, Rourkela, Odisha, India, 12–14 February 2011; pp. 289–294.

20. Warbhe, A.D.; Dharaskar, R.; Thakare, V. Computationally efficient digital image forensic method for image authentication. *Procedia Comput. Sci.* **2016**, *78*, 464–470. [CrossRef]
21. Sitara, K.; Mehtre, B. Detection of inter-frame forgeries in digital videos. *Forensic Sci. Int.* **2018**, *289*, 186–206.
22. Sitara, K.; Mehtre, B.M. Digital video tampering detection: An overview of passive techniques. *Digit. Investig.* **2016**, *18*, 8–22. [CrossRef]
23. Wang, W.; Farid, H. Exposing digital forgeries in video by detecting double quantization. In Proceedings of the 11th ACM Workshop on Multimedia and Security, Princeton, NJ, USA, 7–8 September 2009; pp. 39–48.
24. Wan, Q.; Panetta, K.; Agaian, S. A video forensic technique for detecting frame integrity using human visual system-inspired measure. In Proceedings of the 2017 IEEE International Symposium on Technologies for Homeland Security (HST), Waltham, MA, USA, 25–26 April 2017; pp. 1–6.
25. Calderara, S.; Prati, A.; Cucchiara, R. Video surveillance and multimedia forensics: An application to trajectory analysis. In Proceedings of the First ACM Workshop on Multimedia in Forensics, Beijing, China, 23 October 2009; pp. 13–18.
26. Thomas, J.M. A Robust And Fast Video Copy Detection System Using Spatio-Temporal Features. *IEEE Trans. Inf. Forensics Secur.* **2011**, *2*, 27–33.
27. Stamm, M.C.; Lin, W.S.; Liu, K.R. Temporal forensics and anti-forensics for motion compensated video. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 1315–1329. [CrossRef]
28. Milani, S.; Fontani, M.; Bestagini, P.; Barni, M.; Piva, A.; Tagliasacchi, M.; Tubaro, S. An overview on video forensics. *APSIPA Trans. Signal Inf. Process.* **2012**, *1*. [CrossRef]
29. Ásmundsson, F.H.; Lejsek, H.; Daðason, K.; Jónsson, B.Þ.; Amsaleg, L. Videntifier™ forensic: robust and efficient detection of illegal multimedia. In Proceedings of the 17th ACM International Conference on Multimedia, Vancouver, BC, Canada, 19–24 October 2009; pp. 999–1000.
30. Hong, J.H.; Yang, Y.; Oh, B.T. Detection of frame deletion in HEVC-Coded video in the compressed domain. *Digit. Investig.* **2019**, *30*, 23–31. [CrossRef]
31. Raghavan, S. Digital forensic research: Current state of the art. *CSI Trans. ICT* **2013**, *1*, 91–114. [CrossRef]
32. Kebande, V.R.; Venter, H.S. Adding event reconstruction to a Cloud Forensic Readiness model. In Proceedings of the 2015 Information Security for South Africa (ISSA), Johannesburg, South Africa, 12–13 August 2015; pp. 1–9.
33. Singh, A.; Venter, H.S.; Ikuesan, A.R. Windows registry harnesser for incident response and digital forensic analysis. *Aust. J. Forensic Sci.* **2020**, *52*, 337–353. [CrossRef]