Video Tampering Detection Using Difference-Hashing Algorithm

To cite this article: G. Sujatha *et al* 2021 *J. Phys.: Conf. Ser.* **1804** 012145

ECS The Electrochemical Society
Advancing solid state & electrochemical science & technology
ECS UNITED
247th ECS Meeting
Montréal, Canada
May 18-22, 2025
Palais des Congrès de Montréal
Abstracts due December 6th
Showcase your science!

Video Tampering Detection Using Difference-Hashing Algorithm

G. Sujatha[1], Dr. D. Hemavathi[2], K. Sornalakshmi[3], S. Sindhu[4]
SRM Institute of Science and Technology, Chennai, Tamil Nadu, India.

[1]*sujathag@srmist.edu.in,* [2]*hemavatd@srmist.edu.in,* [3] *sornalak@srmist.edu.in,* [4]*sindhus2@srmist.edu.in*

Abstract: Videos plays an important role in the judgement of many criminal cases. It can be considered as an evidence in every step in the criminal justice process. Since it is involved in the judgement, the integrity of the video is very much important. If anything gets tampered in the video, it leads to misconception in the process. Video tampering detection is a technique of finding whether the video content is modified or not. Alteration of video contents is rising speedily because of increasing video procurement gadgets and enormous programming devices to alter the video content. This integrity check can be done by either active or passive techniques. Along these lines our zone of worry in our research work is to display our perspectives on various passive video tampering detection techniques to check its correctness. Passive techniques are assembled into ensuing three arrangements relying upon the kind of counterfeiting. There are different methods to perform video tampering detection but passive methods prove more effective in terms of maintaining the originality of the video. Even in passive methods, hashing has been the sort after approach because of its accuracy and so we have adopted Difference hashing algorithm (D-Hash) to perform tampering detection.

Keywords: Tampering detection, Video forensics, integrity check, difference hashing.

1. Introduction

Data modification is the alteration of Video content intentionally by the attacker. The modification can be either altering or wrecking of content or even by applying double compression [1,2]. The data can be in either transit state or in rest. In both occasions, data could be blocked and altered with. In many criminal cases, video evidence plays a vital role. This increases the importance of Video Content integrity in turn raise the application of video tampering detection. Video tampering detection has to identify the traces of change in the video content and check the correctness and integrity of the video content. These can be further classified as follows:

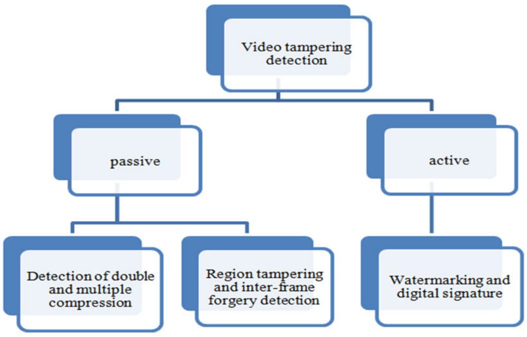

Video tampering detection
passive
active
Detection of double and multiple compression
Region tampering and inter-frame forgery detection
Watermarking and digital signature

Figure.1. Classification of tampering detection methods

In our research work, we are concentrating only the passive tampering detection techniques. The passive video tampering identification can be further categories as follows

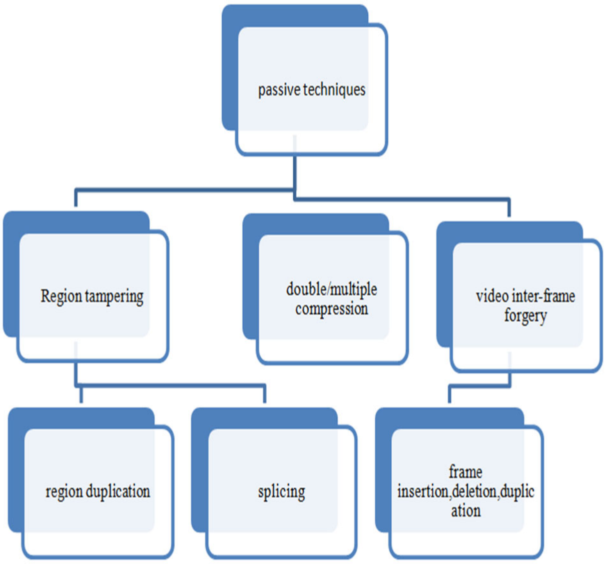

passive techniques
Region tampering
double/multiple compression
video inter-frame forgery
region duplication
splicing
frame insertion,deletion,duplication

Figure 2: Types of passive methods.

1.1 Literature survey:

Computerized multimedia forensic investigation is gaining most extreme significance to confine exploitative utilization of such effly accessible tampering systems. Nowadays, it is really easy for individuals to capture videos even with their advanced mobile phones. There is also a sudden development in the utilization of observation cameras to capture the public events. Videos captured making use of these gadgets typically contains enormous proof of some occasion incidence and in this manner most helpless to inter-frame forgery which may be effortlessly achieved by way of addition/expulsion/replication of frame(s). StaffyKingra [3] proposed this approach, which delivered the method for automatic identification and localization of modification by means of using prediction residual gradient and optical flow gradient. It mechanically detects forged video by means of actually making use of the spikes count irrespective of any quantity of frames in the video.

Videos are frequently widespread to give greater grounded scientific evidence than nonetheless photos, e.g., while applied in claims. Anyways, an extensive association of extremely good and easy to-utilize video altering equipment are available to everybody. In this way, it's far plausible for an intruder to maliciously forge a video e.g., through evacuating or embedding an object in a scene. These types of control can be done with numerous techniques. For instance, a segment of the first video might be supplanted by either a still picture rehashed in time or, in more mind boggling cases, by a video sequence. Besides, the assailant can also use as source facts either a spatial-transient location of a comparable video, or a place taken from an outside association. In the work proposed by Paolo Bestagini[4] they presented the research of the traces left when tampering with a video collection, and endorse an identity calculation that permits a scientific investigator to find video forgeries and restriction them in the spatio-temporary region. This technique is highly accurate in the real global surroundings however it's time consuming.

The correctness and integrity of video is very important.Jie Xu and YuyanLiang[5] proposed a novel approach to detect different types of forgeries of videos that have a still backfround. That is conjoint frames will have very stronginter-relation with each other. If a video is altered then such frames are disturbed and thus their degree of inter-relation is disturbed. Therefore in this method pixel values are obtained by cutting off the sequence of the frames either in the horizontal or vertical angle. Four contiguous pixels constitute a pixel belt. Then with the assist of the histogram intersection approach, the inter-relation between those belts are calculated. The simulations display that if the video tampered, there can be outliers exist within the correlation coefficients. Simulation outcomes show that this paper also can perceive the place of the forgery.

2. Video Tampering Detection

The main purpose of our research work is to identify videos that have been tampered especially [6,7,8] those that can be useful in judicial cases. From time to time many research works has been taken place to find the tampered location [9,10,11,12]. Various methods such as active and passive methods can be used but passive approach [13] have proven to be more effective than the active methods as they do not interfere with the video in any way. These techniques are used to find many tampering types like double compression, frame duplication [14], frame deletion [15,16], and multiple compression [17]. Videos can be an important evidence in related judicial cases and with the help of video forensics [18] these videos are handled in proper way. We use hashing method to check the integrity of the frames and detect if the videos have been tampered or not. Hashing techniques such as MD5 or SHA proved effective for texts but when it comes to images/videos they do not prove to be ideal. So better algorithms have to be introduced to detect tampering in videos. The Difference-hashing or the D-Hash algorithm is used in our project as it proves to be more effective in terms of accuracy and time.

3. Proposed methodology

In our proposed work we are taking videos as input and verifying whether it has been tampered or not. The video taken as input is then split into frames for analysis. To perform this tampering detection, we use D-hash algorithm which is otherwise called as Difference-hashing algorithm which makes use of the difference between the intensities of the pixels to decide the binary value of each of the frames. Now the binary hash value should be converted to decimal for encryption and then it will append the hash value with the source video and send it to the recipient. The recipient then performs the same set of functions and then verification takes place. If the received hash value matches with the calculated value then the receiver can conclude that there is no tampering otherwise it is decided that the video content has been tampered.

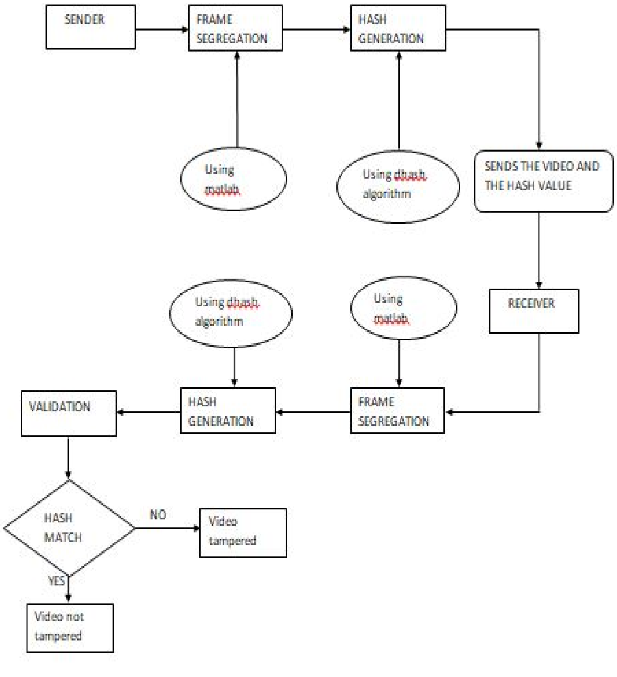


Figure 3: Architecture diagram of the proposed system

4. Design of proposed system

In the proposed system the user takes an input video that is to be validated. Now the video has to be verified using the software that has been developed. We use Matlab for video tampering detection. The videos are segregated into frames and then D-Hash algorithm has to be used to generate the hash and get transmitted with the video to the recipient. This hash value thus generated will avoid hash collision and it should also ensure that the hash values should change even for the smallest of change in the video. This effect is called avalanche effect. This is the main advantage of using D-Hash algorithm as other hash methods do not render to these minute changes.

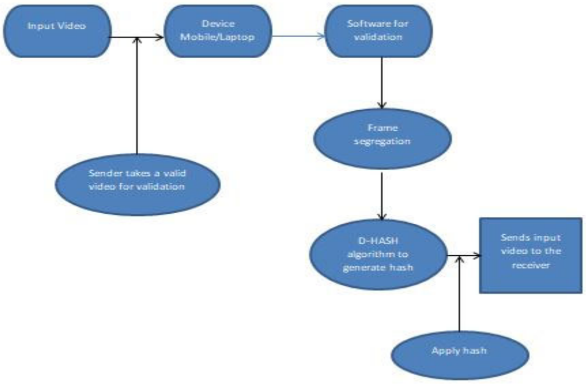

Input Video
Device Mobile/Laptop
Software for validation
Sender takes a valid video for validation
Frame segregation
D-HASH algorithm to generate hash
Sends input video to the receiver
Apply hash

Figure 4: Design of the Proposed System

5. Implementation of Algorithm

D-Hash algorithm used in our work has a few pre-requisites such as conversion of video to frames and then conversion of rgb image to gray scale image. Once these are done, we move onto the actual process where a frame is taken and its pixel values i.e. the intensity values are considered. If the left pixel value is greater than the right pixel value then the bit value is taken as 1 else 0. Similarly, 64 bit value is generated for each frame. This binary value may be tedious to transmit so we convert it to decimal value and apply the hash and send it to the receiver. The same process is repeated at the receiver. Once this is done the final process of verification is done. If the calculated and received hash values match with each other then it is concluded that the content of the video is not get tampered otherwise it is been tampered. D-Hash algorithm used in our research work have a few pre-requisites such as conversion of video to frames and then conversion of rgb image to gray scale image. Once these are done, we move onto the actual process where a frame is taken and its pixel values i.e. the intensity values are considered. If the pixel value in the left side is greater than the pixel value in the right side, then the bit value is taken as 1 else 0. Similarly, 64-bit value is generated for each frame. This binary value may be tedious to transmit so we convert it to decimal value and apply the hash and send it to the receiver. The same process is repeated at the receiver. Once this is done the final process of verification is done. If the calculated and received hash values match with each other then it is concluded that the content of the video is not get tampered otherwise it is been tampered.

6. Working of the algorithm

The steps involved in D-Hash algorithm as follows

A. Grayscale converstion
B. Resizing the frame
C. Computing the difference of pixels
D. Building the hash matrix

6.1. GRAYSCALE CONVERSION:

There are various methods available for the grayscale conversion. They are

i. Single color: Here the gray value is taken as
gray=Red (or)
gray=Green (or)
gray=Blue

ii. The mean method: Here the average value of Red, Green, and Blue components is computed as gray value. It is given by (R+G+B) / 3.

iii.The weighted-average method: This is almost similar to the mean method where the average of RGB components is calculated. Along with that this method also estimate a weightage for the average value. Based on sensitivity, green component is considered the most since it is comparitively more responsive to the naked eye. The possible calculations are

gray= (0.21 R + 0.72 G + 0.07 B)
gray=(0.3 + 0.59 G + 0.11 B)
gray=(0.29 R + 0.58 G + 0.114 B)

6.2. Resize

In order to calculate the hash, we have to resize the frame to 9x8 pixels. This can be achieved by using python package.

6.3. Compute the Difference

The difference algorithm works with the principle of computing the difference between the subsequent pixels. Since the number of pixels in the input image is nine for each row, we use it to determine the difference between subsequent pixels and therefore land at eight values. There are 8 rows with 8 values which leads to a total of 64 values that is a 64-bit hash value.

6.4: Build the Hash:

The final step in the process is building the hash using the input image and the pixel values and we call this the binary test. The test can be carried out as follows

P[a] >P[a + 1] = 1 else 0.

Where a represents the pixel number,

If the intensity of the smaller pixel number in the comparison is greater than the larger pixel number the corresponding output value is set to 1 otherwise it is set to 0. Thus a 64 bit hash value can be generated for each frame.

7. Results

Thus a video whose integrity is to be tested is taken as an input and hash values are generated for the frames. This video is sent along with the hash value to the receiver end. Validation is done at the receiver end to detect tampering and the outcome can be as follows:

i. If the hash values match then the video isn’t tampered and the integrity is 100 percent.

ii.Whereas if the calculated and received hash values are not same, then the video is tampered somewhere and adequate action has to be taken.

Sample Output:

7.1 Video to Frame conversion

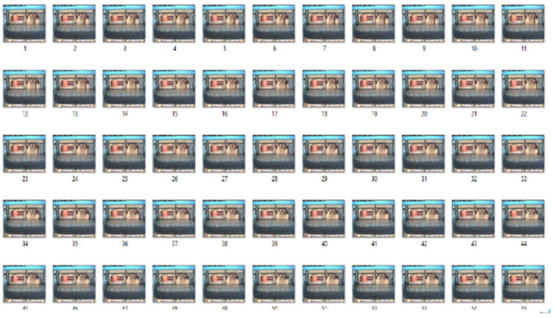
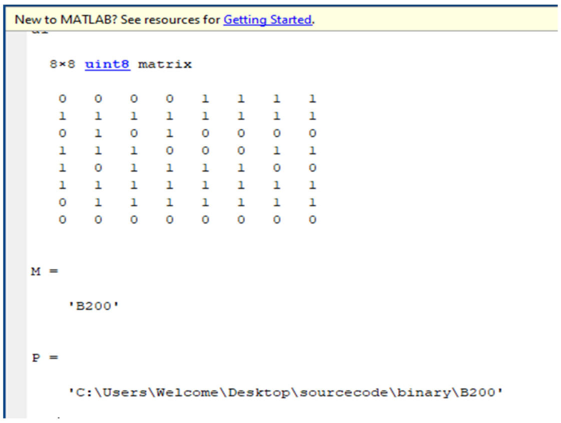


Figure 5: Video to Frame Conversion

7.2 Building hash:

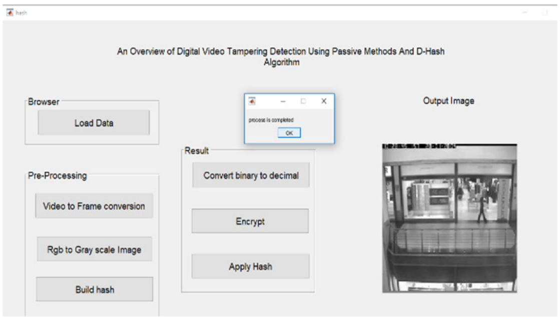


New to MATLAB? See resources for Getting Started.

```
  8×8 uint8 matrix

    0   0   0   0   1   1   1   1
    1   1   1   1   1   1   1   1
    0   1   0   1   0   0   0   0
    1   1   1   0   0   0   1   1
    1   0   1   1   1   1   0   0
    1   1   1   1   1   1   1   1
    0   1   1   1   1   1   1   1
    0   0   0   0   0   0   0   0


M =

    'B200'


P =

    'C:\Users\Welcome\Desktop\sourcecode\binary\B200'
```

Figure 6: Building Hash

7.3 Sending Video

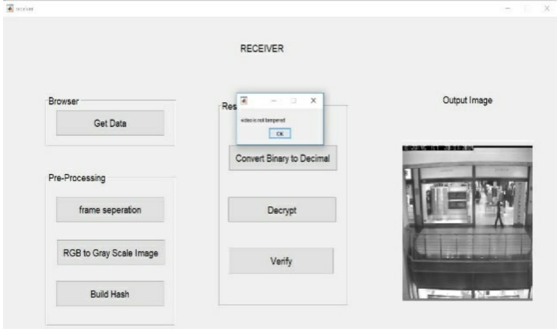


An Overview of Digital Video Tampering Detection Using Passive Methods And D-Hash Algorithm

process is completed

Figure 7: Sending Video

7.4 Verification of Received video



RECEIVER

video is not tampered

Figure 8: Receiving Video and Verification

8. Conclusion

A video content can be tampered by many ways. There is a requirement for verifying the validity of the videos for a variety of judicial cases. In this research work, we applied D-hashing algorithm to ensure the integrity of the video. The availability of video tampering detection tools and efficient software are very less. The result of our work is to simply identifying whether the video content got tampered or not. Further research work can be extended to the location where exactly the modification happened. It is a fruitful research area.

References:

[1]. Jiang, X., He, P., Sun, T., Xie, F., Wang, S. Detection of double compression with the same coding parameters based on quality degradation mechanism analysis.

[2]. Zhensheng Huang1, Fangjun Huang1, Jiwu Huang, Detection of double compression with the same bit rate in MPEG2 videos

[3]. StaffyKingra, Naveen Aggarwal, Raahat Devender Singh, Video inter-frame forgery detection approach for surveillance and mobile recorded videos

[4]. Paolo Bestagini; Simone Milani; Marco Tagliasacchi; Stefano Tubaro, Local tampering detection in video sequences

[5]. Jie Xu; Yuyan Liang; Xingfa Tian; AiyunXie, A Novel video inter-frame forgery detection method based on histogram intersection

[6]. Bestagini, P., Battaglia, S., Milani, S., Tagliasacchi, M., Tubaro, S., 2013a. Detection of temporal interpolation in video sequences. In: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, pp. 3033–3037.

[7]. Bestagini, P., Milani, S., Tagliasacchi, M., Tubaro, S., 2013b. Local tampering detection in video sequences. In: Multimedia Signal Processing (MMSP), 2013 IEEE 15th International Workshop on. IEEE, pp. 488–493.

[8]. Bidokhti, A., Ghaemmaghami, S., March 2015. Detection of regional copy/move forgery in MPEG videos using optical flow. In: Artificial Intelligence and Signal Processing (AISP), 2015 International Symposium on. pp. 13–17.

[9]. Chetty, G., Biswas, M., Singh, R., 2010. Digital video tamper detection based on multimodal fusion of residue features. In: Network and System Security (NSS), 2010 4th International Conference on. IEEE, pp. 606–613.

[10]. Cozzolino, D., Poggi, G., Verdoliva, L., Oct 2014. Copy-move forgery detection based on patchmatch. In: 2014 IEEE International Conference on Image Processing (ICIP). pp. 5312–5316.

[11]. Hsu, C.-C., Hung, T.-Y., Lin, C.-W., Hsu, C.-T., 2008. Video forgery detection using correlation of noise residue. In: Multimedia Signal Processing, 2008 IEEE 10th Workshop on. IEEE, pp. 170-174

[12]. Joshi, V., Jain, S., March 2015. Tampering detection in digital video - a review of temporal fingerprints-based techniques. In: Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on. pp. 1121–1124.

[13]. Lin, C.-S., Tsay, J.-J., 2014. A passive approach for effective detection and localization of region-level video forgery with spatio-temporal coherence analysis. Digital Investigation 11 (2), 120–140.

[14]. Lin, G.-S., Chang, J.-F., Chuang, C.-H., 2011. Detecting frame duplication based on spatial and temporal analyses. In: Computer Science & Education (ICCSE), 2011 6th International Conference on. IEEE, pp. 1396–1399.

[15]. Liu, H., Li, S., Bian, S., 2014. Detecting frame deletion in H.264 video. In: Information Security Practice and Experience. Springer, pp. 262-270

[16]. Shanableh T., 2013. Detection of frame deletion for digital video forensics. Digital Investigation 10 (4), 350–360.

[17]. Milani, S., Bestagini, P., Tagliasacchi, M., Tubaro, S., 2012a. Multiple compression detection for video sequences. In: Multimedia Signal Processing (MMSP), 2012 IEEE 14th International Workshop on. IEEE, pp. 112–117.

[18]. Milani, S., Fontani, M., Bestagini, P., Barni, M., Piva, A., Tagliasacchi, M., Tubaro, S., 2012b. An overview on video forensics. APSIPA Transactions on Signal and Information Processing