Droppix: Towards More Realistic Video Fingerprinting*

Przemysław Błaśkiewicz[1,2], Marek Klonowski[1,2] and Piotr Syga[1,2]
[1] *Wrocław University of Science and Technology, Poland*
[2] *Vestigit sp.z o.o., Poland*

Keywords: Watermarking, Video, Copyright Management, Intellectual Property Protection.

Abstract: We present preliminary results for a way of video fingerprinting that is different from typical methods based on paradigms from watermarking of still images. Our approach is based on modifying some fragments of the clip in a carefully chosen manner. We show the robustness of our approach against a number of typical of attacks. The marks introduced by our modifications are almost imperceptible to the viewer and their impact can be adjusted. Finally, our protocol is computationally light and can be combined with others schemes as an extra security layer.

1 INTRODUCTION

With a rapid growth of multimedia market, the need for preventing piracy is stronger than ever before. An illegal redistribution of copyrighted content causes significant losses to its producers and providers. This is the main motivation for constructing more and more advanced audio and video fingerprinting techniques, that introduce individual marks into each copy of a clip. Such mark (*fingerprint*) can be then linked with a legitimate receiver. If a video is illegally circulated, e.g. retransmitted, the copy can attributed to its associated legitimate user.

One of the fingerprinting techniques for multimedia is watermarking which leverages the fact that bandwidth of digital video/image signals is much higher that the amount of information available to the human eye. The idea is to embed in each copy of a movie or picture marks that are invisible to the viewer, however can be easily detected by the copyright owner.

Designing methods of fingerprinting one needs to take into account that the watermarked content can be subject to various attacks aimed at removing them without significant degradation of quality. That is, an attacker aims at destroying the embedded message, yet preserving the commercial value of the original data. There are many transformations that could be viewed as an attack on the watermark, including rotations, flipping along the vertical axis or using various filters (see e.g. (Asikuzzaman and Pickering, 2018) for a list of possible attacks). In principle, an adequate fingerprinting method needs to provide at least two features The first is **transparency** – quality of the image with embedded fingerprinting information cannot be significantly worse than that of the original. Ideally, the watermarked content is indistinguishable from the original one. Second fundamental issue is **robustness** – it is expected that the fingerprinting method should be immune to all attacks that preserve reasonable quality of the output. That is, the adversary could be capable of removing the characteristic mark only if the video or image is changed to such an extent that it becomes useless for commercial purposes.

While fingerprinting methods designed for static pictures seem to be very mature, there is still many problems with securing video content. Many papers treat video material as a sequence of static pictures and a typical attempt to fingerprint a video is to replace one such picture with any efficient "static" watermark. Even though such approach is theoretically efficient, it usually does not work in practice. The majority of today's video content is represented as a GoP structure (ISO/IEC 23009-1, 2012; ISO/IEC 14496-12, 2011), with carefully chosen key frames (IFrames) and vast majority of finally displayed images is represented as difference between the frames (PFrames or BFrames). While it allows dropping redundant data, an efficient embedding a picture into a

*This Work Is Partially Supported by the National Centre for Research and Development (Ncbir) Project POIR.01.01.01-00-1032/18 and Polish National Science Centre – Project UMO-2018/29/B/ST6/02969.

Błaśkiewicz, P., Klonowski, M. and Syga, P.
Droppix: Towards More Realistic Video Fingerprinting.
DOI: 10.5220/0009876104680476
In *Proceedings of the 17th International Joint Conference on e-Business and Telecommunications (ICETE 2020) - SECRYPT*, pages 468-476
ISBN: 978-989-758-446-6

video can be impossible, since fragile changes can be immediately removed by the encoder before fingerprinting marks are detected.

Our Contribution. In this paper we test an approach to video fingerprinting that is more suitable for real-life video content. The basic idea is to carefully modify some fragments of the video in a way that is characteristic for each copy of a given film. Deletion is however not realised by removing frames but rather fading out some short fragments. This approach preserves the quality of the original video in a much better way and is almost imperceptible to the viewers if implemented properly. In our method the copy of a film can be detected even by cam-coding a film displayed on a wide classes of devices. Our protocol does not require any heavy computations for embedding or detection, unlike some new and very promising watermarking protocols based on machine learning (rather for still images than video material).

To the best of our knowledge, the only similar approach to video fingerprinting is presented in (Lee et al., 2006) wherein the authors present a nice idea of removing single frames from a movie. Note that in contrast to the paper, we discuss the impracticality of removing frames (cf. Sect. 3) and instead we modify their luminance. Moreover, we investigate a wider spectrum of attacks than Lee et al. (cf. Sect 4).

2 PREVIOUS AND RELATED WORK

Methods for Static Pictures. Methods of robust fingerprinting of digital content have been studied for a long time. Majority of commonly used techniques can be found in survey (Potdar et al., 2005). The earliest methods of utilising transparent watermarks relied on negligible modifications made to the spatial domain of the image (e.g. modifying least significant bits (LSB) of the colour intensity (Fridrich, 1998) followed by (Tjokorda Agung B.W et al., 2012) that enables RLE compression or statistical testing of the mean intensity of the marked pixels (Nikolaidis and Pitas, 1998)). Due to the robustness requirement, coping with the effects of compression was of particular interest (Sebé et al., 2000). Often, when the intensity modification was made in the spatial-domain, the authors resort to converting colour space (e.g., to YCbCr (Patvardhan et al., 2018)), that allows more significant (i.e., more robust) modifications without significant visible changes. Note that transparency is not always required; the authors in (Ping Wah Wong and Memon, 2001) allowed a parameter that is responsible for visibility or the transparency of the watermark, depending on the embedder's intentions. In order to allow more robust and transparent modifications, many researchers turned to frequency-domain alterations. Due to the nature of popular compression algorithms, Discrete Cosine Transform was a primarily investigated method (Hwang et al., 2003; Chang and Tsan, 2005). In (Kingsbury, 2000), the author introduced a concept of utilising Dual-tree complex wavelet transform. This approach was followed extensively by (Mabtoul et al., 2007; Alkhathami et al., 2013; Bhatnagar and Wu, 2013; Zebbiche et al., 2018) among others. Another widely used technique is Singular Value Decomposition (SVD) (Gupta and Raval, 2012; Makbol and Khoo, 2013; Makbol and Khoo, 2014). Hybrid approaches, mixing the basic space-domain methods like LSB with frequency-domain approaches were presented e.g., in (Sheth and Nath, 2016), watermarking in frequency-domain with utilisation of SVD (Gaur and Srivastava, 2017) or utilising Hadamard transform with Schur decomposition (Li et al., 2018). A new approach to watermarking was introduced with the advent of neural networks and deep learning. The authors of (Tsai and Liu, 2011) combine the idea of wavelet transforms with utilising neural networks. In (Zhu et al., 2018) a novel framework for end-to-end approach utilising an encoder and decoder with nosier layers was introduced. The authors of (Wen and Aydore, 2019) presented the notion of adversarial training, whereas (Zhong and Shih, 2019) proposed a fully automated system for images captured directly from the camera. An additional neural network was used in (Luo et al., 2020) so that distortions improving robustness were added.

Methods for Motion Pictures. Majority of algorithms for marking video material are based on a direct application of methods constructed for still pictures. The main flaw of direct mapping of the static image watermarking algorithms to motion pictures is that the time-domain compression results, quite often, in deletion or distortion of the embedded watermark. Nevertheless, the authors of many watermarks designed for videos followed the way paved by static images, e.g. DCT (Sun et al.,), DT-CWT (Coria et al., 2008), 3D Discrete Fourier Transform (Deguillaume et al., 1999). In (Noorkami and Mersereau, 2005) main focus was watermarking in H.264 compressed domain and the authors of (Xu et al., 2011) use DCT in order to gain robustness against H.264/AVC compression. Somewhat different, and quite similar to the one described in this paper (cf. 3) approach was presented by (Lee et al., 2006), where certain frames from a movie are removed so that the pattern of missing frames allows unique user identification. An object-based watermarking was introduced in (Swanson et al., 1997) and extended to scene in (Swanson et al., 1998), whereas in (Lee and Seo,) an adaptive modulation is used. A robust, visual watermarking has been proposed in (Zhang et al., 2007). More information on the techniques used in video watermarking may be found in (Doerr and Dugelay, 2003; Bhattacharya et al., 2006; Chang et al., 2011; Asikuzzaman and Pickering, 2018).

3 DESCRIPTION OF THE ALGORITHM

The general intent while working on the algorithm was to embed a unique sequence into the movie, so that the numbers retrieved from the sequence uniquely identify the user leaking the movie. Such approach is fundamentally different to the approach of adapting static image watermarking to embed entire information in a single frame. A natural idea to embed such watermark is to add some mark on selected frames and the number of the frames create a unique sequence. In order to cope with compression (both of a single frame and time-domain) and provide some level of robustness against attacks, one can go to the extreme and remove certain frames (Lee et al., 2006). Such approach has two major downsides: it is difficult to implement in a compressed movie (e.g. AVC or HEVC) and often frame removal is simulated by replacing it with a black one, which allows easy localisation by the adversary and "filling up the hole" if they have access to another copy of the movie. Alternatively, the attacker can black-out few more frames in order to mislead the decoder. To fix it we can ensure that each copy not only has a sequence of frames that are "missing" (either blacked-out or removed) but also there is a frame that **is not** removed in only this single copy. However, the second issue arises with the fix – such idea does not scale well with the number of users. An average feature movie lasts around 100 minutes, which results in roughly 144000 frames. If there needs to be a single frame that is unique to a given user, the watermarking for only 100000 possible users requires removing the majority of the movie for all the users. Due to the problems described above and specific challenges presented by AVC or HEVC compression when trying to remove a frame (especially problematic are the cases when two consecutive frames in a single GOP are removed), instead of removing the frames, replacing them with a black one or leaving some visual macroartifact, we decided to test if modification of the luminance of a frame would provide adequate marking capabilities.

Droppix. We start our algorithm with generating $\mathcal{S}$, a sequence of the frames to be watermarked. It uniquely identifies a user and is generated as a codeword of their unique number. In the final version of the algorithm Reed-Solomon codes or Levenshtein distance may be used to extend S to provide sufficient correcting power for cases when original frames from S have not all been detected by the decoder or additional ones have been falsely identified. Next, we decode the movie in order to obtain a sequence of frames. For each frame $s \in \mathcal{S}$ we perform the same operations (Alg. 8). We start by converting the frame into CIELAB colour space. Next, we multiply the L component (lightness) of each pixel by a predefined constant α. Our test showed that in order for a robust, yet transparent watermark, the values between 0.92 and 0.97 can be used. We proceed with converting the frame back to BGR colour space and replacing the resulting frame instead of s. When all frames in $\mathcal{S}$ were modified, we code the movie back to the original container using compression algorithm.

```
Data: $M$ – movie to be watermarked, $k$ – user
      key (e.g. device ID), $\alpha$ – watermark
      strength parameter
Result: $M$ – watermarked movie
$\mathcal{S} \leftarrow$ SequenceGen($k$);
$\langle f_1, f_2, \ldots, f_n \rangle \leftarrow$ Decode($M$);
for $i \in \{1, 2, \ldots, n\}$ do
    if $i \in \mathcal{S}$ then
        $\hat{f} \leftarrow$ BGR2LAB($f_i$);
        $\hat{f}_L \leftarrow \hat{f}_L \cdot \alpha$;
        $f_i \leftarrow$ LAB2BGR($\hat{f}$);
$M \leftarrow$ Encode($\langle f_1, f_2, \ldots, f_n \rangle$);
```

Algorithm 1: Droppix: watermark embedding.

In order to decode the watermark, we need also the original movie and the set of parameters used by the embedding algorithm. We start by decoding both movies to obtain respective sequences of frames. We proceed by comparing pairs of corresponding frames using structural similarity (Zhou Wang et al., 2004) of their grayscale versions. Next, we calculate the mean square error of the similarity distances for each frame-pair in an array mse. In order to retrieve the information on the frames that were watermarked we establish a threshold of considered values and localising the peaks of the normalised errors. The indexes of the peaks are then subject to the reverse $\mathcal{S}$ generating algorithm in order to correct possible mistakes in retrieving the information and to recover the user ID of the leaker. The idea of the decoding procedure is shown in Algorithm 2.

```
Data: $M$ – original movie, $\hat{M}$ – copy of the
      movie with potential watermark, $\mathcal{K}$ –
      set of user keys, $t$ – acceptance
      threshold
Result: $k$ – user ID
SEtab←[];
$\langle f_1, f_2, \ldots, f_n \rangle$ ←Decode($M$);
$\langle \hat{f}_1, \hat{f}_2, \ldots, \hat{f}_n \rangle$ ←Decode($\hat{M}$);
for $i \in \{1, 2, \ldots, n\}$ do
    $g$ ←RGB2GRAY($f_i$);
    $\hat{g}$ ←RGB2GRAY($\hat{f}_i$);
    if StructuralSimilarity($g$,$\hat{g}$)<$t$
     then
        mse ←MSE($g$, $\hat{g}$);
        SEtab.append(mse);
median←median(SEtab);
Δ ←SEtab−median;
P←findPeakIndexes(Δ);
k←SequenceGen$^{-1}$(P);
```

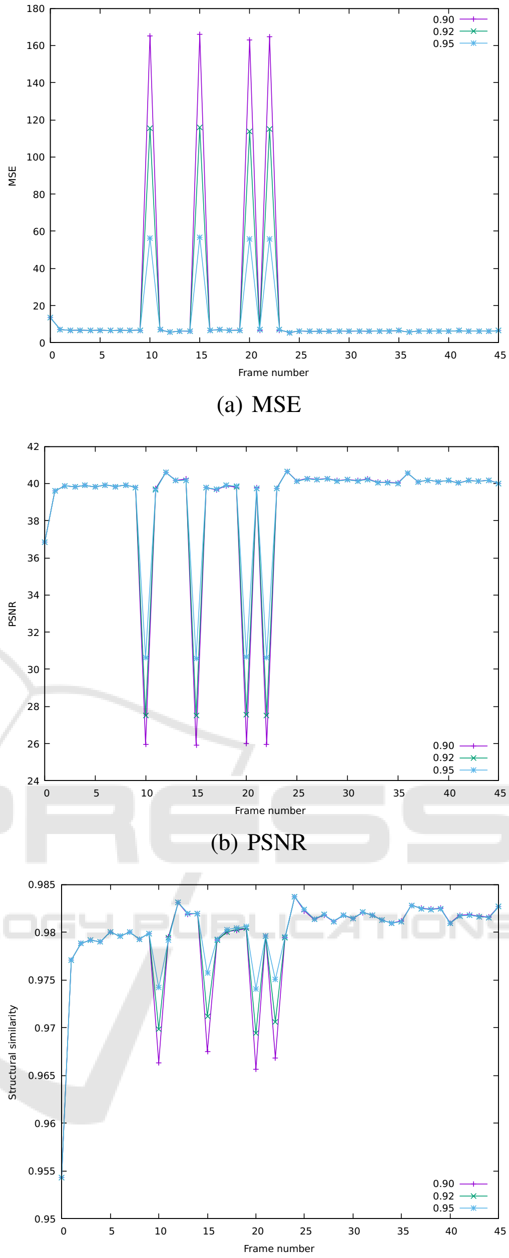
Algorithm 2: Droppix: watermark decoding.

4 EXPERIMENTAL RESULTS

Corpus. In order to test our solution we gathered a corpus of clips from various commercial television broadcasts to be our testing body. They all were H.264 encoded HD quality 120 seconds long fragments. In each of those samples a watermark was embedded using our method, and for each such marking the numbers of frames that we used for watermarking were stored.

Watermarking. Next, we checked how our watermarking procedure changes values of mean square error (MSE), structural similarity (SS) and peak signal-to-noise ratio (PSNR) calculated with respect to non-watermarked clips. Fig. 1, shows the influence of droppix parameter α on those indicators for an exemplary clip. One can tell that as α approaches 1, there is smaller influence of the watermark on the parameters, and hence detection success rate can be expected to drop. More importantly, the watermarking leaves a clear peak in all three values, a property we want to maintain when the clip undergoes a modification attack.

Visual inspection of the influence of α resulted in the following observations. For value 0.9 the modified frame can be slightly noted as flicker, with 0.95 a (subjective) perception is marginal and depends heavily on the particular scene the frame belongs to. On the other hand, as discussed below, for values above 0.95 the success rate of detection deteriorates rapidly. Example original and watermarked frames (for $\alpha = 0.92$) and their difference are shown in Fig. 2. Note that difference between "clean" and watermarked frame is a uniquely almost-black rectan-

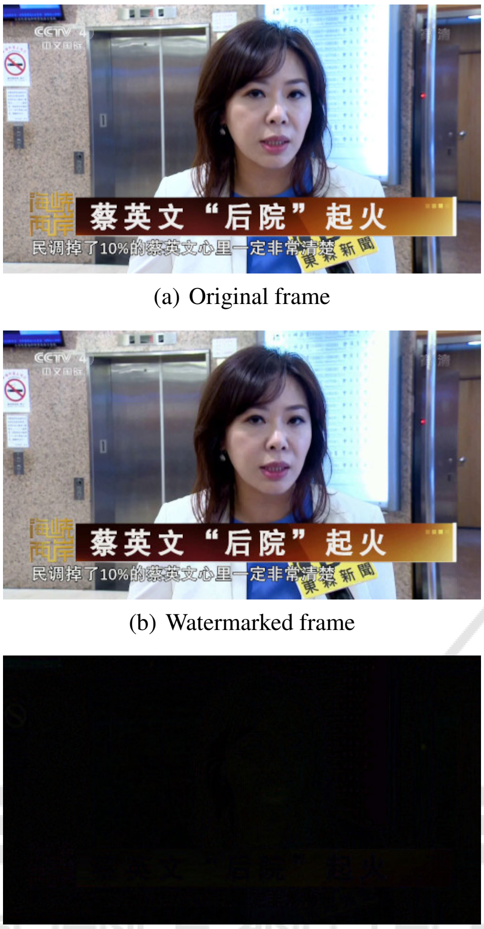


(a) MSE

(b) PSNR

(c) SS

Figure 1: Influence of watermarking on MSE, PSNR and SS values for different α.



(a) Original frame

(b) Watermarked frame

(c) Difference – uniformly almost black

Figure 2: Example frame from testing clip.

gle, since the luminance of all pixels has been reduced by the same (small) amount.

Attacks. The aim of the attack in our scenario is to change PSNR, MSE and SS characteristics of the clip, so that finding the frames used for watermarking becomes impossible.

From many possible attacks possible we chose six to give a general idea about the performance of our scheme. These attacks are easy to perform and as such are more available to less experienced attackers, and there might be more leaked content modified by such simple means. The attacks we chose are in the list below, and for each we provide a short rationale for choosing it. All attacks kept other characteristics (resolution, framerate, etc.) of the movie unchanged.

1. Transcoding from H.264 to MPEG-2 format and downsampling to SD. Both these formats are popular and can provide good quality video. Due to different methods of compression, transcoding can effectively erase pixel-level modifications used in other watermarking techniques. Since our approach modifies lightness (also a visual property), we checked how different compression algorithm influences detection.
2. Vertical flip. This method is frequently applied to counter watermarking techniques relying on spatial information within a frame. Importantly, vertical flip can be visually acceptable for the viewer as little content is lost and there are many examples of such content on the internet.
3. Globally increasing brightness. Since lightness of a frame is our embedding technique, such attack might limit our detection capability.
4. Slight rotation (1 degree). Similar to the vertical flip above, this attack yields visually acceptable results to the viewer, yet is able to mis-lead watermarking schemes utilising spacial information of a frame. Because we utilise structural similarity between frames as one of the decision factors, we wanted to verify how such slight change might affect detection.
5. Blurring. This is one of the common attacks against watermarks embedded in pixel structure of the frame. To test our approach with this attack, we selected (subjectively) maximal parameters for blurring both chrominance and luminance channels so that the result was still acceptable to the viewer as HD content.
6. Frame averaging. This is another popular technique of attack against watermarking. Essentially, it is a low-pass filter applied to the movie so that fast/sharp movement, which the human eye is not capable of identifying is smoothed out. In our tests we transformed each frame to be the average of two consecutive frames, yielding still acceptable results to the viewer.

All attacks were performed using *ffmpeg* in version 4.2.2 and the implementation of the watermarking and detection algorithms was done in Python with OpenCV ver. 4.2.0 bindings and SciKit ver. 0.16.2

Testing Framework. Each clip from the corpus was first watermarked. Next, the content was modified according to the attack scenario. Finally, both the original clip and the watermarked and modified content were passed to watermark detection. In that process, the PSNR, structural similarity and mean-square error (MSE) for each pair of frames was calculated. Next, frame numbers suspected as ones encoding the watermark were selected. Finally, the chosen frame numbers were matched against those stored for particular clip at the moment of watermark insertion. If all frames have been found and no other – the trial was deemed success. Any additional frames *except* those determined by watermarking yielded a false-positive. And finally, when not all watermarked frames were found, the trial was considered failed.

Results. For the sake of presentation, the results are shown for clips where watermark was embedded using frames number 10, 15, 20 and 22, which allows visual inspection of obtained data. There were 28 such clips. Data presented in Fig. 1, 3, 4 and 5 correspond to the same clip, so that relative change between non-attacked and attacked/modified video can be traced.

Overall results are shown in Tab. 1. Columns "success" and "FP" show the number of clips where the watermark was identified, and where *additional* frames were also selected, respectively. The column "extra" presents the mean number of extra frames detected and the column "σ" shows its standard deviation.

It can be seen that the watermark embedding is little influenced by brightness modification, and much less still by transcoding and down-sampling. Rotation by a small degree makes the scheme more prone to false-positives as well as limits the success rate. Vertical flip attack seems to be most effective against our method, however, on closer inspection it was revealed that MSE values calculated in this case were *smaller* then average for frames used for watermarking than for regular ones. This is exactly the opposite from what has been observed for other attacks and consequently our decoding algorithm was not able to detect those negative peaks. This is illustrated in Fig. 3.

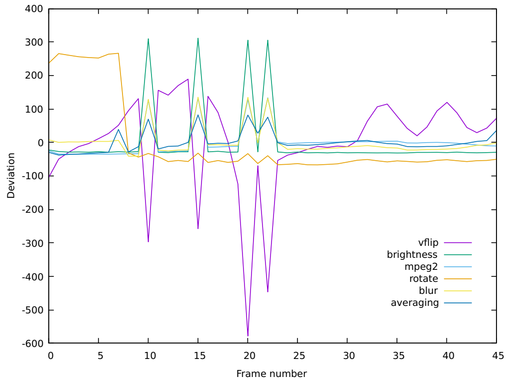


Figure 3: MSE deviation from average per frame for different attacks on the same movie. Watermark is inserted in frames 10, 15, 20 and 22.

In general, the flipping attack has the same impact on PSNR and structural similarity indicators in that it produces *opposite* orientation of peaks as compared to other studied attacks (Figs. 4 and 5). Therefore, for this particular attacks our detection algorithm must be adequately adjusted.

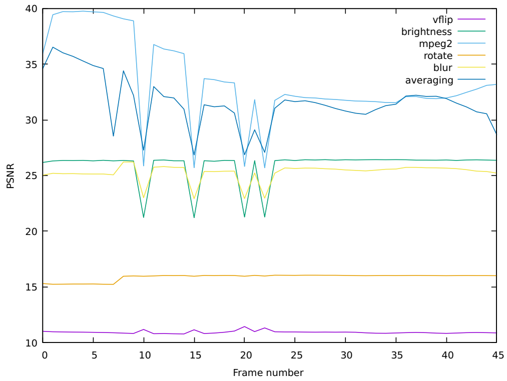


Figure 4: PSNR of frames for different attacks on the same movie. Watermark is inserted in frames 10, 15, 20 and 22.

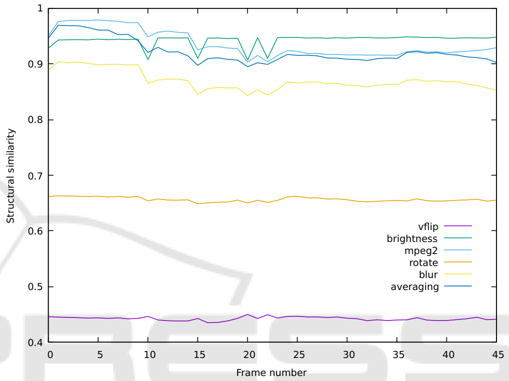


Figure 5: Structural similarity of attacked frames to originals for different attacks on the same movie. Watermark is inserted in frames 10, 15, 20 and 22.

Interpretation of Results. Our preliminary results on the proposed method for watermarking suggest that it is immune against some popular and common attacks on video content. While some aspects of the detection method require a more evolved approach, the tests support our claim about potential of this method, particularly as a one-of-many methods used concurrently on the same video content.

Building a fully functional detector requires more samples and wider range of attack cases, or employing ANN as a decision maker based on a sequence of tuples of the form (PSNR, MSE, SS) for each frame. Different clips have different thresholds by which to decide whether a given frame pertains to the watermark or not. As a consequence, the detector should determine acceptance/rejection parameters on the fly and for particular clip, with only some initial startup.

Because our research was about testing the idea, our testing scenario did not use redundancy codes

Table 1: General detection results for 28 clips.

α	0.9				0.92				0.95			
Attack	success	FP	extra	σ	success	FP	extra	σ	success	FP	extra	σ
brightness	28	0	–		28	0	–		28	9	5.88	1.36
vflip	0	27	3.22	2.79	0	25	3.25	2.81	0	27	3.28	2.71
mpeg2 + SD	28	3	1	0	28	3	1	0	28	2	1	0
rotate	23	28	4.14	2.42	20	27	3.96	2.8	16	27	4.07	2.22
blur	27	8	8	5.07	26	12	8.58	4.31	24	18	8.78	4.37
averaging	26	5	2.4	1.36	25	7	4.14	3	23	11	6.36	3.08

nor did we use any correlation between numbers of frames used for embedding. The use of Reed-Solomon codes and the like would allow rising $\alpha = 0.94$ even higher (since 0.98 produced more failed attempts due to not all frames being detected). And because in our tests not detecting *all* frames used for embedding was considered failure, some standard method of recovering missing information would further rise the success rate.

As we mentioned above, a broader framework can be devised where there is *another* scheme, that co-exists in the movie and is resistant to other attacks, for which our scheme is susceptible.

5 CONCLUSION

In our paper we presented an approach to fingerprinting of video materials that diverges from a usual paradigm. The proposed protocol provides immunity against some common attacks and is adequate for contemporary multimedia format. Moreover, the transparency of our scheme is acceptable enough to be commercially acceptable and can be traded off for better detection rates. The presented protocol is somehow "orthogonal" to and independent from many other methods of typical fingerprinting. Therefore we believe that our approach can also be used as an extra security layer together with other, classical protocols. The synergy of such combination can lead to a very strong immunity against various attacks. One path of experimentation is that of immunizing the scheme against attack using a certain number of copies of the movie.

Additionally, as a future work we leave out enhancing the method so that it allows blind-identification. The only reason the original footage is required is for estimation of MSE between the frames, however as one of every few IFrames is watermarked by increasing its luminance channel, it will be an outlier among frames describing the same scene. Localisation of the outliers in a sliding window would allow identification of the marked frames without a reference clip, and due to its significant robustness against averaging it is feasible to adjust the parameters so that blind detection is possible without significant loss in accuracy.

REFERENCES

Alkhathami, M., Han, F., and Van Schyndel, R. (2013). Fingerprint image watermarking approach using dtcwt without corrupting minutiae. In *2013 6th International Congress on Image and Signal Processing (CISP)*, volume 03, pages 1717–1723.

Asikuzzaman, M. and Pickering, M. R. (2018). An overview of digital video watermarking. *IEEE Transactions on Circuits and Systems for Video Technology*.

Bhatnagar, G. and Wu, Q. J. (2013). Biometrics inspired watermarking based on a fractional dual tree complex wavelet transform. *Future Generation Computer Systems*, 29(1):182 – 195. Including Special section: AIRCC-NetCoM 2009 and Special section: Clouds and Service-Oriented Architectures.

Bhattacharya, S., Chattopadhyay, T., and Pal, A. (2006). A survey on different video watermarking techniques and comparative analysis with reference to h.264/avc. In *2006 IEEE Int. Symposium on Consumer Electronics*.

Chang, H. T. and Tsan, C. L. (2005). Image watermarking by use of digital holography embedded in the discrete-cosine-transform domain. *Appl. Opt.*, 44.

Chang, X., Wang, W., Zhao, J., and Zhang, L. (2011). A survey of digital video watermarking. In *2011 Seventh International Conference on Natural Computation*, volume 1, pages 61–65.

Coria, L. E., Pickering, M. R., Nasiopoulos, P., and Ward, R. K. (2008). A video watermarking scheme based on the dual-tree complex wavelet transform. *IEEE Transactions on Information Forensics and Security*, 3.

Deguillaume, F., Csurka, G., O'Ruanaidh, J. J., and Pun, T. (1999). Robust 3D DFT video watermarking. In *Security and Watermarking of Multimedia Contents*, volume 3657. Int. Society for Optics and Photonics.

Doerr, G. and Dugelay, J.-L. (2003). A guide tour of video watermarking. *Signal Processing: Image Communication*, 18(4):263 – 282. Special Issue on Technologies for Image Security.

Fridrich, J. (1998). Image watermarking for tamper detection. In *Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No.98CB36269)*, volume 2, pages 404–408 vol.2.

Gaur, S. and Srivastava, V. K. (2017). A hybrid rdwt-dct and svd based digital image watermarking scheme using arnold transform. In *2017 4th Int. Conf. on Signal Processing and Integrated Networks (SPIN)*.

Gupta, A. K. and Raval, M. S. (2012). A robust and secure watermarking scheme based on singular values replacement. *Sadhana*, 37(4):425–440.

Hwang, D.-C., Bae, K.-H., Lee, M.-H., and Kim, E.-S. (2003). Real-time stereo image watermarking using discrete cosine transform and adaptive disparity maps. In Tescher, A. G., Vasudev, B., Jr., V. M. B., and Divakaran, A., editors, *Multimedia Systems and Applications VI*, volume 5241, pages 233 – 242. International Society for Optics and Photonics, SPIE.

ISO/IEC 14496-12 (2011). Information technology – Coding of audio-visual objects. Standard, International Organization for Standardization, Geneva, CH.

ISO/IEC 23009-1 (2012). Information technology – Dynamic adaptive streaming over HTTP (DASH). Standard, International Organization for Standardization, Geneva, CH.

Kingsbury, N. (2000). A dual-tree complex wavelet transform with improved orthogonality and symmetry properties. In *Proceedings 2000 Int. Conf. on Image Processing*, volume 2.

Lee, S.-W. and Seo, D.-I. Novel robust video watermarking algorithm based on adaptive modulation.

Lee, Y., Kim, C., and Lee, S. (2006). Video fingerprinting based on frame skipping. In *Proceedings of the International Conference on Image Processing, ICIP 2006, October 8-11, Atlanta, Georgia, USA*. IEEE.

Li, J., Yu, C., Gupta, B. B., and Ren, X. (2018). Color image watermarking scheme based on quaternion hadamard transform and schur decomposition. *Multimedia Tools and Applications*, 77(4):4545–4561.

Luo, X., Zhan, R., Chang, H., Yang, F., and Milanfar, P. (2020). Distortion Agnostic Deep Watermarking. *arXiv e-prints*, page arXiv:2001.04580.

Mabtoul, S., Hassan, E., Elhaj, I., and Aboutajdine, D. (2007). Robust color image watermarking based on singular value decomposition and dual tree complex wavelet transform. In *2007 14th IEEE International Conference on Electronics, Circuits and Systems*.

Makbol, N. M. and Khoo, B. E. (2013). Robust blind image watermarking scheme based on redundant discrete wavelet transform and singular value decomposition. *AEU-International Journal of Electronics and Communications*, 67(2):102–112.

Makbol, N. M. and Khoo, B. E. (2014). A new robust and secure digital image watermarking scheme based on the integer wavelet transform and singular value decomposition. *Digital Signal Processing*, 33:134–147.

Nikolaidis, N. and Pitas, I. (1998). Robust image watermarking in the spatial domain. *Signal Processing*, 66.

Noorkami, M. and Mersereau, R. M. (2005). Compressed-domain video watermarking for h.264. In *IEEE Int. Conf. on Image Processing 2005*.

Patvardhan, C., Kumar, P., and Vasantha Lakshmi, C. (2018). Effective color image watermarking scheme using ycbcr color space and qr code. *Multimedia Tools and Applications*, 77(10):12655–12677.

Ping Wah Wong and Memon, N. (2001). Secret and public key image watermarking schemes for image authentication and ownership verification. *IEEE Transactions on Image Processing*, 10(10):1593–1601.

Potdar, V. M., Han, S., and Chang, E. (2005). A survey of digital image watermarking techniques. In *INDIN '05. 2005 3rd IEEE International Conference on Industrial Informatics, 2005.*, pages 709–716.

Sebé, F., Domingo-Ferrer, J., and Herrera, J. (2000). Spatial-domain image watermarking robust against compression, filtering, cropping, and scaling. In Goos, G., Hartmanis, J., van Leeuwen, J., Pieprzyk, J., Seberry, J., and Okamoto, E., editors, *Information Security*, pages 44–53. Springer Berlin Heidelberg.

Sheth, R. K. and Nath, V. V. (2016). Secured digital image watermarking with discrete cosine transform and discrete wavelet transform method. In *2016 International Conference on Advances in Computing, Communication, Automation (ICACCA) (Spring)*, pages 1–5.

Sun, J., Yang, N., Liu, J., Yang, X., Li, X., and Zhang, L. Video watermarking scheme based on spatial relationship of dct coefficients.

Swanson, M. D., Bin Zhu, and Tewfik, A. H. (1998). Multiresolution scene-based video watermarking using perceptual models. *IEEE Journal on Selected Areas in Communications*, 16(4):540–550.

Swanson, M. D., Zhu, B., Chau, B., and Tewfik, A. H. (1997). Object-based transparent video watermarking. In *Proceedings of First Signal Processing Society Workshop on Multimedia Signal Processing*.

Tjokorda Agung B.W, Adiwijaya, and Permana, F. P. (2012). Medical image watermarking with tamper detection and recovery using reversible watermarking with lsb modification and run length encoding (rle) compression. In *2012 IEEE Int. Conf. on Communication, Networks and Satellite (ComNetSat)*.

Tsai, H.-H. and Liu, C.-C. (2011). Wavelet-based image watermarking with visibility range estimation based on hvs and neural networks. *Pattern Recognition*, 44.

Wen, B. and Aydore, S. (2019). ROMark: A Robust Watermarking System Using Adversarial Training. *arXiv e-prints*, page arXiv:1910.01221.

Xu, D., Wang, R., and Wang, J. (2011). A novel watermarking scheme for h.264/avc video authentication. *Signal Processing: Image Communication*, 26(6):267 – 279.

Zebbiche, K., Khelifi, F., and Loukhaoukha, K. (2018). Robust additive watermarking in the dtcwt domain based on perceptual masking. *Multimedia Tools and Applications*, 77(16):21281–21304.

Zhang, J., Ho, A. T. S., Qiu, G., and Marziliano, P. (2007). Robust video watermarking of h.264/avc. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 54(2):205–209.

Zhong, X. and Shih, F. Y. (2019). A Robust Image Watermarking System Based on Deep Neural Networks. *arXiv e-prints*, page arXiv:1908.11331.

Zhou Wang, Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612.

Zhu, J., Kaplan, R., Johnson, J., and Fei-Fei, L. (2018). HiDDeN: Hiding Data with Deep Networks. In *The European Conference on Computer Vision (ECCV)*.