If You Know What I've Seen

An Audio Landmark Detection based approach for Video Fingerprinting

Manu Ramesh
Department of ECE
Purdue University

Praneet Singh
Department of ECE
Purdue University

***Abstract*—Video fingerprinting is a task that involves matching video segments to the original video source in order to perform tasks like source video retrieval and plagiarism detection. In today's day and age, Deep Learning Based techniques have been extensively applied to most fields but their reliance on large amount of training data and computational resources can serve as a huge obstacles. When it comes to Video Fingerprinting, it would be next to impossible to train a neural network on every video that exists today to achieve accurate fingerprinting. In this paper, we leverage the useful work done in the field of Audio Fingerprinting and show that it can seamlessly be used on Videos. We use a simple yet powerful landmark based audio fingerprinting technique and show that its performance on videos is comparable to that of the current learning based methods. We also test the effect of common distortions on our fingerprinting scheme and propose mitigation strategies to improve fingerprinting in the presence of these distortions.**

***Index Terms*—Video fingerprinting, plagiarism, Deep Learning, landmark, distortions**

I. Introduction

Have you ever caught a glimpse of a TV show segment and wondered which episode it is from to watch the whole of it? Have you ever watched portion of a movie on someone else's screen, couldn't recognize the actors but wanted to know which one it was? Ever wanted to find out the source of a viral meme video? Video to video search in all these contexts would be the tool to go for. One could hold up their smartphone cameras, record the video for a few seconds and then use this tool to get the original video. A quick internet search showed us that while there exist tools to search for audio with audio fragments, images with images, there is no simple tool handy to search for videos with videos.

One obvious way to search for original videos using segments is to perform a cross correlation between the video segment and all the available videos in the reference database. This brute force approach would be highly inefficient. There could also be distortions due to the properties of video capturing and rendering device. Illumination, scale and many other factors hinder the efficient functioning of this tool. On the lines of 'Audio Fingerprinting', we go on to define 'Video Fingerprinting' as a method to search for full, original video tracks given just a fragment of it, despite differences in color, scale, encoding, noise etc. In this paper we discuss methods that we developed to achieve this task.

Since audio fingerprinting has achieved significant success, with Music Information Retrieval services such as Shazam being the quintessential contributor, we decided to try a similar approach on videos. An obvious starting point would be to somehow make use of the existing audio retrieval algorithms directly. To achieve this, we devise ways to convert video signals to pseudo audio signals and then use the audio fingerprinting algorithms directly. Figure 1 depicts this strategy.

Similar to the use cases of audio fingerprinting algorithm, the video fingerprinting tool could potentially be used for detecting plagiarism and for monitoring media as well.

This paper is divided into five sections. Section II lists the related works, Section III describes our methodology in detail. The experiments we conducted to gauge the performance of our method and the results of these experiments are detailed in Section IV and the potential mitigation strategies are discussed in Section V. Section VI has the concluding remarks.

II. Related Work

Tasks like Audio Fingerprinting, Image & Video based retrieval & searching techniques are very similar to Video Fingerprinting. In most of these tasks, the state-of-the-art techniques use learning based approaches. Our aim in this paper is to show that suitable fingerprinting can be achieved without any learning involved.

Several popular image retrieval & search algorithms are reviewed in [1], [2], [3] & [4]. Most of these algorithms are however constructed to retrieve related images. Although, these techniques could be used effectively for image fingerprinting, they directly cannot be employed for video due to the simple fact that the same image could be present in multiple videos thereby rendering the systems ineffective.

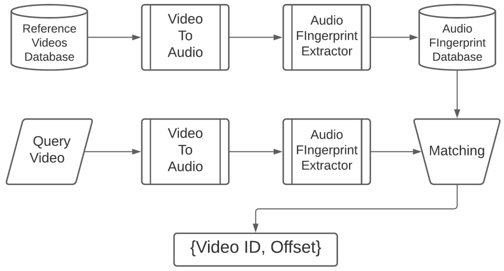
Reference Videos Database
Video To Audio
Audio FIngerprint Extractor
Audio FIngerprint Database
Query Video
Video To Audio
Audio FIngerprint Extractor
Matching
{Video ID, Offset}

Fig. 1: Video Fingerprinting Strategy

Video based techniques for retrieval have been reviewed in [5] & [6]. Most of these papers deal with learning-based techniques that are very powerful but are also are computationally expensive. These techniques have been built to search for closely related videos instead of the exact video to which a particular input video segment belongs.

In terms of Video Fingerprinting, learning based approaches have been proposed in [3], [7] & [8]. These techniques rely on 3D Convolutional Neural Networks which are extremely difficult to train due to time and GPU constraints. In our paper, we are attempting to eliminate the need of these 3D Networks while maintaining suitable video fingerprinting accuracy. A similar approach to ours has been tried in [9], where multiple videos of an event are matched using the video's audio . However, in our case we make sure we only work with the video.

Techniques that work extremely well across domains are very popular today. Taking note of the efficacy of audio fingerprinting techniques we thought of using them directly on videos instead of audio. We went in search of an audio fingerprinting algorithm that works and works well within our constraints. [10] compares three different audio fingerprinting techniques based on various criteria. Since it is obvious video files are much larger than audio files to begin with, we decided to go with the algorithm that generates the reference database of the smallest size i.e [11].

III. Methodology

Steering away from a learning based technique to achieve accurate Video Fingerprinting, we decided to use a similar approach to the one proposed in [11].

This landmark based fingerprinting technique, works by first computing the spectrogram of the audio signal. It then searches for distinctive local maxima/peaks within this spectrogram, while suppressing other peaks within a predefined neighbourhood from the detected peaks. Each peak is defined by its position in the spectrogram, by the tuple (f, t), which defines its frequency and time coordinates. These peaks by themselves aren't sufficient to serve as hashes. So, peaks within a certain distance of each other are chosen in pairs and their relative positions are recorded as a landmark triplet - $(t_2 - t_1, f_1, f_2 - f_1)$. This triplet is used to compute the hash value. The frequencies and the frequency differences are quantized using an $L = 256$ quantizer while the time difference is quantized using an $L = 64$ level quantizer. This means that there are $256 \times 256 \times 64$ available hash values; which is sufficient. These hash values are used to create an inverted index map $[hash \rightarrow \{trackID, offset\}]$; i.e., a hash table with the hashes as the key and the values being a list of all locations in all tracks where the hashed landmark has occurred is generated and used as the reference database. When a query audio sample is passed, the landmark hashes are computed and matched with the reference database. Temporal alignment is then applied to the detected matches to find the one true match.

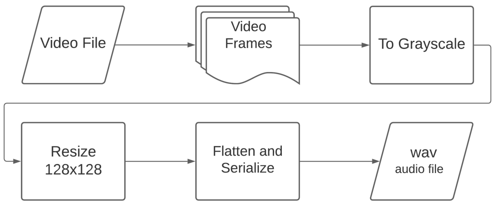

Fig. 2: Video To Audio Conversion Procedure

Our approach is described in Fig 2. We use a video as input, read one frame at a time. Each frame that is read is converted to a grayscale, to make the algorithm robust to color shifts. Next, the grayscale frame is downscaled to a 128×128 image. The size of the downscaled image is a hyper-parameter. This downsampling is to ensure that the algorithm is invariant to scale shifts in the query image. The down-sampled frame is then flatted or reshaped into a $1 - D$ array which is then written as a single channel (mono) *wav* audio file, sample by sample. Each pixel value in the frame is written as a single sample. The audio sample rate should be equal to frame height$(128) \times$ frame width$(128) \times FPS$. This sample or frame rate acts as a scaling factor, enabling the fingerprint matching algorithm to compute the exact time of occurrence of the video segment. The resulting *wav* files are then ingested by the audio fingerprinting software, which outputs a list of landmark hashes. This procedure is thus used as a wrapper around the audio fingerprinting software to process the reference videos and query videos. Thus, the reference videos are used to generate a reference database of audio landmark hashes and the query videos are converted to audio landmark hashes before matching is performed. This approach is so much simpler for fingerprinting than having to incrementally train a neural network every time new videos get created.

IV. Experiments & Results

In this section, we will first review the dataset being used for our experiments. Next, we evaluate the baseline performance of our method based on the number of frames that are required to achieve suitable video fingerprinting. We then understand how common distortions like watermarking, compression, flip & rotation and resolution scaling affect the performance of our video fingerprinting technique. We follow a systematic approach of evaluation as proposed in [8] so that we can compare our results to the learning based method to some extent.

A. Dataset

In order to evaluate our method, we used a subset of the UCF-101 Video Database [12]. The UCF-101 Dataset is an Activity Recognition Dataset with videos of varying lengths and resolutions that are divided into 101 classes.

In our case, we used all the videos from the UCF-101 Dataset that lasted more than 15 seconds. As a result, our matching database was constructed using a total of 200 videos each having a resolution of 320x240. We generated test segments of length $\leq 10s$ from the same sequences to ensure suitable evaluation of our fingerprinting scheme.

B. Baseline Performance

From [8], we see that their 3D-ResNet models achieved a testing accuracy of 93% when the first 100 frames (4 seconds) of each video from the UCF-101 set was used as a query.

We wanted to see how well our dataset could perform based on the number of frames available per video segment for fingerprinting i.e with the first 50, 100, 150, 200, & 250 frames available. The results are as seen in Table 1

Method	Number of Frames	Duration (s)	Accuracy %
Xinwei Li et al.	100	4	93
Ours	50	2	11.44
Ours	100	4	29.35
Ours	150	6	38.805
Ours	200	8	44.2786
Ours	250	10	94.527

TABLE 1: Query Length vs Fingerprinting Accuracy

We can clearly see that our method performs better as the number of frames increases. Although, we do require more frames to achieve a suitable fingerprinting accuracy as seen in [8], it is acceptable as our technique has no training & GPU requirement. For all the other experiments present in this section, we use segments of 250 frames each for fingerprinting.

C. Effect of Distortions

Video fingerprinting systems can break due to some commonly distortions seen in Videos. We will now test our method on some of these distortions.

1) Resolution: In this case we wanted to evaluate how different input resolutions would affect our method's accuracy. Thus, we re-scaled our test segments while maintaining the original aspect ratio. The accuracy of our approach on different resolution inputs can be seen in Table 2

Resolution	Accuracy %
320x240	94.5
400x300	74.626
240x180	60.696

TABLE 2: Fingerprinting Accuracy based on Query Resolution

As we see, variations in resolution of the input reduces the effectiveness of our approach. The algorithm still does a good enough job due to the rescaling step applied during the video to audio conversion.

2) Frame Rotation & Flip: Similar to our previous experiment, another simple strategy to fool our fingerprinting technique is rotation. We tested our matching algorithm on $90\deg$ and $180\deg$ rotated versions of queries & also tested it on vertical and horizontal flipped queries. Results are seen in Table 3.

Our technique clearly is not rotation invariant. Surprisingly, it works better on $180\deg$ rotated versions of the test segments than the original segments. This needs to be explored further. One possible approach to make our algorithm invariant to rotation and flip is explained in Section V.

Rotation Angle	Accuracy %	Flip	Accuracy %
None	94.5	None	94.5
$90\deg$	0.45	Horizontal Flip	7.46
$180\deg$	97.512	Vertical Flip	5.4

TABLE 3: Rotation / Flip vs Fingerprinting Accuracy

3) Watermarking: Adding a custom watermark to the videos is equivalent to adding distortion to pixels at fixed positions. We watermarked every test segment at four different locations. The fingerprinting accuracy was then computed and the values are reported in Table 4

Watermark Location	Accuracy %
No Watermark	94.5
Bottom Left	45.77
Bottom Right	49.25
Top Left	56.716
Top Right	57.21

TABLE 4: Watermark Location vs Fingerprinting Accuracy

We can clearly see that watermarking reduces the accuracy of our method but finding methods to combat this could be significantly difficult.

4) Compression: Most videos available online are compressed using popularly video codecs like x265 & x264. In this experiment, we test how compression affects our fingerprinting algorithm.

We compress all our test segments using x265 in both the All-Intra and Inter frame (P-frames only) encoding modes at five different Quantization parameters (QPs) i.e 10, 20, 30, 40, & 50. The effects of compression become more prominent as the QP value increases.

QP	All-Intra	Inter
10	81.59	79.6
20	76.11	77.6
30	72.636	69.15
40	63.6	57.2
50	34.32	34.32

TABLE 5: x265 Encoding vs Fingerprinting Accuracy

Looking at the results of encoding in Table 5, we can see that the effectiveness of our approach decreases as the QP increases i.e as the compression increases.

V. Potential mitigation Strategies

A. Multi-scale approach

In this approach, we sequentially search for the query videos across multiple scales to achieve two goals: 1) To reduce the number of query frames required for fingerprinting & 2) To help improve fingerprinting accuracy.

As seen in Figure 3, this approach involves a sequential search across different scale sizes to match queries with the database videos. If the query is not found in the 128x128 scale, we move on to the next scale. We do this till we reach a scale where the video is either found or we have iterated through the entire pyramid but no match has been found.

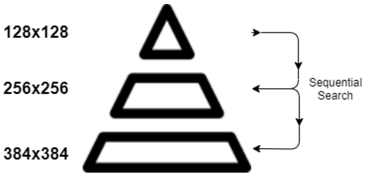

Fig. 3: Scale Pyramid

Scale	Accuracy %
128	46.67
256	53.3
384	73.3
128 → 256	66.67
128 → 256 → 384	86.67

TABLE 6: Using the Scale Pyramid to improve Fingerprinting Accuracy

In order to evaluate this, we compiled a smaller dataset of 15 YouTube videos but of much higher resolution than the UCF dataset i.e 720p. Also, the YouTube videos have an average length of 2 minutes but we have restricted the queries to 10 seconds. The results of using different scales and a multi-scale approach is seen in the Table 6.

Clearly, using a multi-scale approach helps us improve our fingerprinting accuracy. We also believe that this method will help us get better results with smaller queries on the UCF dataset.

B. Rotation and Flip Invariance

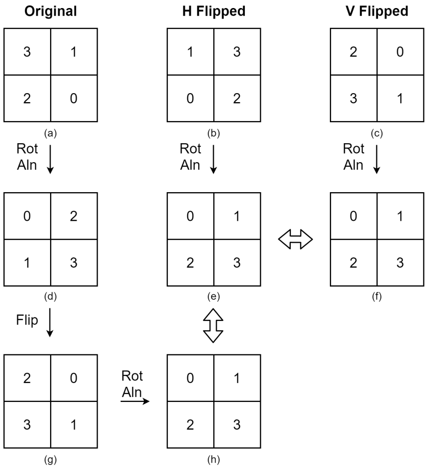

Fig. 4: Rotation and Flip alignment

The 128×128 grayscale frame is divided into four quadrants and each quadrant is ranked in increasing order of their means; as seen in Fig 4 (a). We then 'Rotate Align' the frame by forcing the 0^{th} ranked quadrant to occur at the top left position. We rotate the frame till this is achieved (Fig 4 (d)). Fig 4 (b) & (c) show horizontally and vertically flipped versions of the original frame. The rotation aligned versions of these frames are exactly the same (see Fig 4 (e) & (f)), meaning that overcoming one of those two is sufficient. A quick glance at the rotation aligned original and the flipped frames will tell us that they only differ in the secondary diagonal ranks. We now force the lower ranked quadrant of the two to be on the top right. This is done by flipping the frame and rotation aligning it. We force every frame of the database and query videos to undergo this procedure so that they come out with the same orientation with respect to rotation and flip, thereby achieving invariance.

C. Video2Audio++

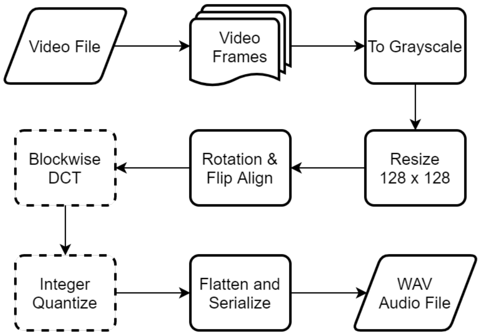

Fig. 5: Video2Audio++

We tried two more video to audio methods. The first one, Video2Audio+, incorporates the rotation and flip alignment to the existing method shown in Fig 2, while the second one, Video2Audio++ adds blockwise DCT and integer quantization. We retained only the top 48 coefficients of each DCT block. The addition of Integer Quantized DCT was to increase robustness to compression and high frequency noise. A comparison of these techniques can be found in Table 7.

Distortion	Video2Audio	Video2Audio+	Video2Audio++
Rotated 90	0.45	63.18	68.02
Horizontal Flip	7.46	63.18	68.02
Vertical Flip	5.4	63.18	68.02

TABLE 7: Performance comparison on Rotated and Flipped queries

VI. Conclusion

The technique proposed in this paper leverages a simple Audio fingerprinting approach based on Landmark detection for Videos. We have shown that a simple non-learning technique can be used to do effective video fingerprinting.

We have discussed the effects of several commonly seen distortions on our fingerprinting approach. We also provide solutions to negate the effects of few of the distortions. However, efforts are being made towards negating the effects of watermarking and compression.

References

[1] W. Chen, Y. Liu, W. Wang, E. Bakker, T. Georgiou, P. Fieguth, L. Liu, and M. S. Lew, "Deep image retrieval: A survey," *arXiv preprint arXiv:2101.11282*, 2021.

[2] M. T. S. A. Singh, J. Boaddh, and J. Samar, "A survey on digital image retrieval technique and visual features," 2021.

[3] X. Li, J. Yang, and J. Ma, "Recent developments of content-based image retrieval (cbir)," *Neurocomputing*, 2021.

[4] B. Jena, G. K. Nayak, and S. Saxena, "Survey and analysis of content-based image retrieval systems," in *Control Applications in Modern Power System*. Springer, 2021, pp. 427–433.

[5] D. A. Phalke and S. Jahirabadkar, "A survey on near duplicate video retrieval using deep learning techniques and framework," in *2020 IEEE Pune Section International Conference (PuneCon)*. IEEE, 2020, pp. 124–128.

[6] A. Oerlemans, Y. Guo, M. S. Lew, and T.-S. Chua, "Special issue on deep learning in image and video retrieval," 2020.

[7] L. Xinwei, X. Lianghao, and Y. Yi, "Compact video fingerprinting via an improved capsule net," *Systems Science & Control Engineering*, pp. 1–9, 2020.

[8] X. Li, C. Guo, Y. Yang, and L. Xu, "Video fingerprinting based on quadruplet convolutional neural network," *Systems Science & Control Engineering*, vol. 9, no. sup1, pp. 131–141, 2021.

[9] C. V. Cotton and D. P. Ellis, "Audio fingerprinting to identify multiple videos of an event," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2010, pp. 2386–2389.

[10] V. Chandrasekhar, M. Sharifi, and D. Ross, "Survey and evaluation of audio fingerprinting schemes for mobile query-by-example applications," in *12th International Society for Music Information Retrieval Conference (ISMIR)*, 2011.

[11] A. Wang, "An industrial strength audio search algorithm." 01 2003.

[12] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.