International Journal of Innovative Technologies
WWW.IJITECH.ORG
ISSN 2321-8665
Vol.05,Issue.01,
January-2017,
Pages:0033-0042

Cloud-Based Multimedia Content Protection System

SIRISHA. A[1], RAMESH BABU VARUGU[2]
[1]PG Scholar, Dept of CSE, Annamacharya Institute of Technology and Sciences, Hyderabad, TS, India,
E-mail: siri04vinny@gmail.com.
[2]Associate Professor & HOD, Dept of CSE, Annamacharya Institute of Technology and Sciences, Hyderabad, TS, India,
E-mail: ramesh.vnl@@gmail.com.

Abstract: We propose another outline for vast scale interactive media content assurance frameworks. Our outline influences cloud foundations to give cost productivity, quick arrangement, versatility, and flexibility to suit differing workloads. The proposed framework can be utilized to secure distinctive media content sorts, including 2-D recordings, 3-D recordings, pictures, sound clasps, melodies, and music cuts. The framework can be sent on private and/or open mists. Our framework has two novel segments: (i) strategy to make marks of 3-D recordings, and (ii) conveyed coordinating motor for sight and sound items. The mark strategy makes hearty and agent marks of 3-D recordings that catch the profundity signals in these recordings and it is computationally productive to process and think about and also it requires little stockpiling. The circulated coordinating motor accomplishes high adaptability and it is intended to bolster distinctive sight and sound articles. We executed the proposed framework and sent it on two mists: Amazon cloud and our private cloud. Our investigations with more than 11,000 3-D recordings and 1 million pictures demonstrate the high exactness and versatility of the proposed framework. Furthermore, we contrasted our framework with the assurance framework utilized by YouTube and our outcomes demonstrate that the YouTube insurance framework neglects to distinguish most duplicates of 3-D recordings, while our framework recognizes more than 98% of them. This correlation demonstrates the requirement for the proposed 3-D signature technique, since the cutting edge business framework was not ready to handle 3-D recordings.

Keywords:3-D Video, Cloud Applications, Depth Signatures, Video Copy Detection, Video Fingerprinting.

I. INTRODUCTION

Propels in handling and recording hardware of media substance and additionally the accessibility of free web facilitating locales have made it generally simple to copy copyrighted materials, for example, recordings, pictures, and music cuts. Illicitly redistributing sight and sound substance over the Internet can bring about critical loss of incomes for substance makers. Finding illicitly made duplicates over the Internet is a complex and computationally costly operation, as a result of the sheer volume of the accessible mixed media content over the Internet and the multifaceted nature of contrasting substance with recognize duplicates. We show a novel framework for sight and sound substance security on cloud bases. The framework can be utilized to secure different media content sorts, including normal 2-D recordings, new 3-D recordings, pictures, sound clasps, tunes, and music cuts. The framework can keep running on private mists, open mists, or any mix of open private mists. Our configuration accomplishes fast arrangement of substance security frameworks, since it depends on cloud bases that can rapidly give processing equipment and programming assets. The outline is financially savvy since it utilizes the registering assets on interest. The configuration can be scaled here and there to bolster changing measures of sight and sound substance being ensured. The proposed framework is genuinely mind boggling with various parts, including:

- Crawler to download a large number of mixed media objects from web facilitating locales.
- Signature technique to make delegate fingerprints from sight and sound articles.
- Distributed coordinating motor to store marks of unique questions and match them against inquiry objects.

We propose novel strategies for the second and third parts, and we use off-the-rack instruments for the crawler. We have built up a complete running arrangement of all parts and tried it with more than 11,000 3-D recordings and 1 million pictures. We sent parts of the framework on the Amazon cloud with changing number of machines (from eight to 128), and alternate parts of the framework were conveyed on our private cloud. This arrangement model was utilized to demonstrate the adaptability of our framework, which empowers it to proficiently use changing figuring assets and minimize the expense, since cloud suppliers offer distinctive valuing models for registering and system assets. Through broad examinations with genuine arrangement, we demonstrate the high exactness (as far as accuracy and review) and additionally the adaptability and versatility of the proposed framework. The commitments of this paper are as per the following.

- Complete multi-cloud framework for sight and sound substance insurance. The framework underpins diverse sorts of interactive media content and can adequately use differing figuring assets.

- Novel technique for making marks for 3-D recordings. This strategy makes marks that catch the profundity in stereo substance without figuring the profundity signal itself, which is a computationally costly process.

New plan for a circulated coordinating motor for high-dimensional interactive media objects. This configuration gives the primitive capacity of discovering - closest neighbors for huge scale datasets. The configuration likewise offers a helper capacity for further preparing of the neighbors. This two-level configuration empowers the proposed framework to effectively bolster distinctive sorts of interactive media content. For instance, in discovering video duplicates, the fleeting perspectives should be considered notwithstanding coordinating individual casings. This is not at all like discovering picture duplicates. Our configuration of the coordinating motor utilizes the Map Reduce programming model. Rigorous assessment study utilizing genuine execution to evaluate the execution of the proposed framework and look at it against the nearest works in the educated community and industry. In particular, we assess the whole end-to-end framework with 11,000 3-D recordings downloaded from YouTube. Our outcomes demonstrate that a high accuracy, near 100%, with a review of more than 80% can be accomplished regardless of the fact that the recordings are subjected to different changes, for example, obscuring, trimming, and content insertion. Likewise, we think about our framework versus the Content ID framework utilized by YouTube to ensure recordings. Our outcomes demonstrate that in spite of the fact that the Content ID framework gives vigorous discovery of 2-D video duplicates, it neglects to identify duplicates of 3-D recordings when recordings are subjected to even straightforward changes, for example, re-encoding and determination change.

Our framework, then again, can recognize all duplicates of 3-D recordings regardless of the fact that they are subjected to complex changes, for example, incorporating new virtual perspectives and changing over recordings to anaglyph and 2-D-in addition to profundity positions. Besides, we disconnect and assess singular segments of our framework. The assessment of the new 3-D signature technique demonstrates that it can accomplish more than 95% accuracy and review for stereoscopic substance subjected to 15 diverse video changes; a few of them are particular to 3-D recordings, for example, view union. The assessment of the disseminated coordinating motor was done on the Amazon cloud with up to 128 machines. The rest of this paper is organized as follows. We summarize the Existing and Proposed systems in Section II. In Section III, we present the Implementation. In Section IV. Presents Results and Finally Section V Presents Conclusion of this paper.

II. EXISTING AND PROPOSED SYSTEMS

A. Existing System

The problem of protecting various types of multimedia content has attracted significant attention from academia and industry. One approach to this problem is using watermarking, in which some distinctive information is embedded in the content itself and a method is used to search for this information in order to verify the authenticity of the content. Many previous works proposed different methods for creating and matching signatures. These methods can be classified into four categories: spatial, temporal, color, and transform-domain. Spatial signatures (particularly the block-based) are the most widely used. YouTube Content ID, Mobile VDNA, and Mark Monitor are some of the industrial examples which use fingerprinting for media protection, while methods such as can be referred to as the academic state-of-the-art.

B. Proposed System

We present a novel system for multimedia content protection on cloud infrastructures. The system can be used to protect various multimedia content types. In our proposed system we present complete multi-cloud system for multimedia content protection. The system supports different types of multimedia content and can effectively utilize varying computing resources. Novel method for creating signatures for videos this method creates signatures that capture the depth in stereo content without computing the depth signal itself, which is a computationally expensive process new design for a distributed matching engine for high-dimensional multimedia objects. This design provides the primitive function of finding -nearest neighbors for large-scale datasets as shown in Fig.1. The design also offers an auxiliary function for further processing of the neighbors. This two-level design enables the proposed system to easily support different types of multimedia content. The focus of this paper is on the other approach for protecting multimedia content, which is content-based copy detection (CBCD). In this approach, signatures are extracted from original objects. Signatures are also created from query (suspected) objects downloaded from online sites. Then, the similarity is computed between original and suspected objects to find potential copies.

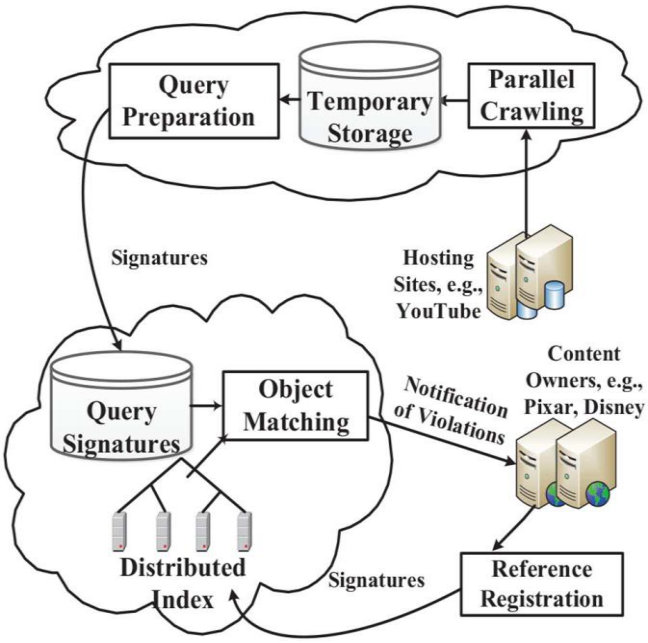

Fig.1.System Architecture.

Advantages of Proposed System:

- Accuracy.
- Computational Efficiency.
- Scalability and Reliability.
- Cost Efficiency.
- The system can run on private clouds, public clouds, or any combination of public-private clouds.
- Our design achieves rapid deployment of content protection systems, because it is based on cloud infrastructures that can quickly provide computing hardware and software resources.
- The design is cost effective because it uses the computing resources on demand.
- The design can be scaled up and down to support varying amounts of multimedia content being protected.

III. IMPLEMENTATION

A. Modules

- Data owner Module
- Data User Module
- Encryption Module
- Rank Search Module

B. Modules Description

Data owner Module: Protect different multimedia content types, including 2-D videos, 3-D videos, images, audio clips, songs, and music clips. The system can be deployed on private and/or public clouds. Our system has two novel components: (i) method to create signatures of 3-D videos, and (ii) distributed matching engine for multimedia objects. The signature method creates robust and representative signatures of 3-D videos that capture the depth signals in these videos and it is computationally efficient to compute and compare as well as it requires small storage.

Data User Module: Matching engine achieves high scalability and it is designed to support different multimedia objects. We implemented the proposed system and deployed it on two clouds: Amazon cloud and our private cloud. Our experiments with more than 11,000 3-D videos and 1 million images show the high accuracy and scalability of the proposed system. In addition, we compared our system to the protection system used by YouTube and our results show that the YouTube protection system fails to detect most copies of 3-D videos, while our system detects more than 98% of them

Encryption Module: Multimedia content protection systems using multi-cloud infrastructures .The proposed system supports different multimedia content types and it can be deployed on private and/or public clouds. Two key components of the proposed system are presented. The first one is a new method for creating signatures of 3-D videos. Our method constructs coarse-grained disparity maps using stereo correspondence for a sparse set of points in the image.

Rank Search Module: Rank needs to store the whole reference dataset multiple times in hash tables; up to 32 times. On the other hand, our engine stores the reference dataset only once in bins. Storage requirements for a dataset of size 32,000 points indicate that Rank needs up to 8 GB of storage, while our engine needs up to 5 MB, which is more than 3 orders of magnitude less. These storage requirements may render Rank not applicable for large datasets with millions of points, while our engine can scale well to support massive datasets.

IV. EVALUATION

We have implemented and integrated all parts of the proposed content protection system: from a web user interface to control various parts of the system and its configurations, to tools to allocate, release, and manage cloud resources, to all algorithms for creating and matching signatures, as well as all distributed MapReduce algorithms for processing thousands of multimedia objects. This is a fairly complex system with tens of thousands of lines of code in different programming and scripting languages. We validated our proposed multi-cloud architecture by deploying part of our system on the Amazon cloud and the other part on our local private cloud. The Amazon cloud had up to 20 machines and our private cloud had 10 machines each with 16 cores. We deployed the Parallel Crawling and Query Preparation components on the Amazon cloud. This is because the Amazon cloud has large Internet links and it can support downloading thousands of multimedia objects from various sites, such as YouTube. The relatively close proximity and good connectivity of Amazon data centers in North America to major multimedia content hosting sites accelerates the download process. More importantly, at the time of our experiments, the Amazon pricing model did not charge customers for inbound bandwidth while it charged for outbound bandwidth. Since the majority of our workload is downloading multimedia objects (inbound traffic), this deployment minimized our costs, and it indeed shows the benefits of our architecture, which can opportunistically utilize resources from different clouds after downloading each multimedia object, we create signatures from it and immediately delete the object itself as it is no longer needed–we keep the object URL link on the hosting site from which we downloaded it. This minimizes our storage cost on Amazon.

Signatures from multiple multimedia objects are then grouped, compressed, and transferred to our private cloud for more intensive processing. Once uploaded to the private cloud, the signatures are deleted from Amazon to save storage. On our private cloud, we deploy the matching engine and all of its related operations. These include building the distributed index from reference objects and matching query objects versus reference objects in the index. The crawling and matching operations are done periodically; in our system we do it once daily, when our local cloud is lightly loaded. We rigorously evaluate the proposed system using real deployment with thousands of multimedia objects. Specifically, in the following subsections, we evaluate our system from four angles: (i) complete system performance, (ii) comparison with YouTube, (iii) analysis of the signature method, and (iv) accuracy, scalability and elasticity of the distributed matching engine component.

A. Performance of the Complete System

Videos: We assess the performance of the whole system with a large dataset of 11,000 3-D videos downloaded from YouTube. These videos are from different categories, and have diverse sizes, durations, resolutions, and frame rates. Thus, they represent a good sample of most 3-D videos on YouTube. 10,000 of these videos make the bulk of our query set, while the other 1,000 videos make the bulk of our reference set. We downloaded both the 10,000 query videos and the 1,000 reference videos in a similar manner as follows, while keeping a list of all previously downloaded video IDs to ensure that the reference set does not include any of the downloaded query videos. First, we used the APIs provided by YouTube to download the top 100 videos in terms of view count in each video category. YouTube has 22 categories: Music, Entertainment, Sports, Film, Animation, News, Politics, Comedy, People, Blogs, Science, Technology, Gaming, How to, Style, Education, Pets, Animals, Autos, Vehicle, Travel, and Events. Since some categories did not have any 3-D videos and some of them had a small number, we added a set of seed queries to expand our dataset. The queries we used were: 3d side by side, 3d trailer, 3d landscape, 3d animation, 3d football, 3d gaming and 3d ads. For the reference video set, in addition to the 1,000 downloaded videos, we used 14 other videos that were manually downloaded to be as diverse as possible and the likelihood of them being in the query set is low. We chose 10 out of these 14 videos and manipulated each of them using the ffmpeg video processing tool in five different ways: cutting clips or segments, scaling, blurring, logo insertion, and cropping. Thus, we created 50 videos in total. We added these 50 manipulated videos to the query set to ensure that the query set has matches to some of the videos in the reference set, which made the query set have 10,050 videos. We created signatures from all reference videos and inserted them in the matching engine.

Methodology: For each frame of the query videos, the signature is computed and the closest signatures to it are retrieved from the distributed index. Candidate matches undergo an additional step to ensure that the number of matching frames in the two videos is enough. For example, if frame x in the query video matches frame y in the reference video, we expect frame x+1in the query video to match frame y+1 in the reference video. In order to consider this, a matching matrix is computed for each pair of candidate reference video and query video. The size of a matching matrix is the number of frames in the considered reference video times the number of frames in the query video against which the reference video is being compared. A value of 1 in the (i, j) position of the matching matrix means that the ith frame of the reference video has matched the jth frame of the query video the longest diagonal sequence of 1s in this matrix indicates the largest number of matching frames and is considered as a potential copy. The final matching score between videos is the number of matches on the diagonal divided by the diagonal length. We introduce a threshold parameter to decide whether two videos match. If two videos have a matching score less than, they are not considered a match; otherwise they are a match. We vary the threshold between 0 and 1.0. We measure the performance in terms of two basic metrics: precision (percentage of returned videos that are true copies) and recall (percentage of true video copies that are returned). In this large experiment, we compute the exact precision, which

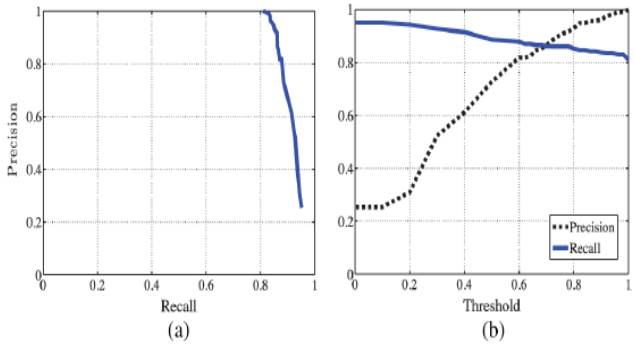

Fig.2. Performance of the complete system for 3-D video copy protection on more than 11,000 3-D videos. (a) Precision-recall curve. (b) Precision and recall versus threshold s.

is possible as we can check whether a match declared by the system is a true match or not by watching the videos. Computing the exact recall is tricky though, since we cannot be 100% sure that the large query set does not contain any copies other than the added 50 videos, although we tried to minimize this possibility. The only way to be sure is to manually check each of the 10,000 videos, which is a formidable task. To partially mitigate this issue, we compute an approximation of the recall assuming that there are no other copies in the 10,000 videos. Thus, the computed recall should be viewed as an upper bound on the achievable recall of our system. We compute the exact recall on small datasets in later sections.

Results: We plot the results of this experiment in Fig. 2. Fig. 2(a) shows the precision-recall (PR) curve, where we plot the approximate recall as discussed above. To get this PR curve, we change the threshold from 0 to 1, and compute the precision and recall for each threshold. PR curves are a standard evaluation method in image retrieval, as they contain rich information and can easily be read by researchers. The results clearly show that our system can achieve both high precision and recall. For example, a precision of 100% with a recall of more than 80% can be achieved. To further analyze the results, we show in Fig. 2(b), how the precision and recall vary with the threshold parameter s. The results show that our method can achieve precision and recall values of more than 80% for a wide range of thresholds from 0.6 to 1. This means that our system does not only provide high accuracy, but it is not very sensitive to the threshold s, which is an internal system parameter. In other words, the system administrator does not need to accurately fine tune s. In summary, this large-scale experiment with 11,000+ 3-D videos and the whole system deployed on multiple distributed machines confirms the accuracy of the proposed system.

B. Comparison with YouTube

YouTube is one of the largest online video sharing and streaming sites in the world. It offers a video protection service to its customers, which employs a sophisticated system to detect illegal copies of protected videos. The system used in YouTube is called Content ID,[2] which is a proprietary system and we cannot know much details about it beyond what YouTube disclosed in its patent [10]. The goal of this subsection is not to conduct full comparison between our system and Content ID, which is not possible. Our goal is to show that while the Content ID system provides robust copy detection for traditional 2-D videos, it fails to detect most copies of 3-D videos and that the proposed system, which employs our new 3-D signature method, outperforms Content ID by a large margin.

Methodology: To test the Content ID system, we download several copyrighted 2-D and 3-D videos from YouTube. We perform various transformations on these videos and then upload them back to YouTube to see whether the Content ID system can detect them as copies. In case of detection, YouTube shows the message "Matched third party content" when a copyrighted video is uploaded similarly, we test our system by using the same 3-D videos downloaded from YouTube and subjected to the same transformations. In particular, we downloaded six 3-D and six 2-D protected videos from Warner Bros. and 3net YouTube channels. The video lengths are in the range of 30 seconds to 2 minutes. When we uploaded each of these 12 videos back to YouTube without any modifications, the Content ID system correctly identified them all as copies.

Results for 2-D Videos: We tested YouTube for detecting modified 2-D videos. We applied six transformations on each of the six 2-D videos: blur, format change, frame dropping, re-encoding with different resolution (scale), cutting 30 second clip, and cutting 40 second clip. Then, we uploaded all 36 modified versions of the 2-D videos to YouTube. We found that the Content ID system can detect all blur, format change, frame dropping, and re-encoding with different resolution transformations as copies, resulting in a recall of 100% for these transformations. For the clip transformation, only one of the 30 seconds clips was detected, but all 40 seconds clips were detected as copies. Therefore, this experiment shows that the YouTube Content ID is quite robust for 2-D videos.

Results for 3-D Videos: Now, we test the Content ID system on 3-D videos and compare it against our system. Recall that our system is general and can support different types of media, but we focus in this paper on designing signatures for 3-D videos as there has been little work on these videos, whereas there are many works for 2-D copy detection. The original 3-D videos downloaded from YouTube are in side-by-side format. We applied 15 different transformations; the first ten of these transformations are common between 2-D and 3-D videos, while the other five are specific to 3-D videos. The transformations on each 3-D video are: blur, file format change, re-encoding with same bit-rate, re-encoding with different bit-rate, re-encoding with different resolution (scale), frame dropping, 30 seconds clip, 35 seconds clip, 40 seconds clip, 45 seconds clip, anaglyph, row-interleaved, column-interleaved, 2-D-plus-depth, and view synthesis. Anaglyph means changing the video format such that the right and left images are encoded in different colors to render 3-D perception on regular 2-D displays using anaglyph glasses (color filters). Row and column-interleaved indicate changing the format of the left and right images to suit row- and column-interleaved types of displays. The 2-D-plus-depth transformation computes a depth for the video and presents the video as 2-D stream and depth stream, which can be rendered by certain types of 3-D displays. View synthesis is used to create additional virtual views from the basic stereo video. This is done to enhance user's experience or to evade the copy detection process.

TABLE I: Comparison Against Youtube in Terms of Recall

Transformation	YouTube	Proposed System
Blur	0/6	6/6
File format change (mp4 to avi)	6/6	6/6
Re-encoding: same bit-rate	0/6	6/6
Re-encoding: different bit-rate	0/6	6/6
Re-encoding: different resolution	0/6	6/6
Frame dropping	0/6	6/6
30 seconds clip	1/6	6/6
35 seconds clip	2/6	6/6
40 seconds clip	4/6	6/6
45 seconds clip	5/6	6/6
Anaglyph	5/6	5/6
Row-interleaved	5/6	6/6
Column-interleaved	6/6	6/6
2D-plus-depth	0/6	6/6
View synthesis	0/6	6/6

As in 2-D videos, we uploaded all modified 90= (15×6) 3-D videos to YouTube in order to check whether the Content ID system can identify them as copies. We explicitly specified that these videos are 3-D when we uploaded them to YouTube. The results are shown in the second column of Table I. The results clearly show the poor performance of the Content ID system on 3-D videos. For example, the Content ID system could not detect even a single copy of the six 3-D videos when they are subjected to seven different transformations. Some of these transformations are as simple as blurring and re-encoding while others are more sophisticated such as view synthesis and 2-D-plus-depth conversion. Furthermore, except for few transformations, the Content ID system misses most of the modified copies. The three transformations anaglyph, row-interleaved, and column-interleave result in videos that are similar to their corresponding 2-D versions. Since the 2-D versions of the used 6 3-D videos are also under copyright protection, the videos resulting from such transformations are most probably matched against the 2-D versions of the original videos.

To assess the accuracy of our system, we use the same six 3-D videos as our reference dataset. We also add the 14 videos mentioned in Section IV-A to the reference dataset. We apply the same 15 transformations on the six 3-D videos, resulting in 90 query videos. We add 1,000 other 3-D videos downloaded from YouTube to the query dataset. We add

these noise videos in order to check whether our system returns false copies. We report the results from our system in column three of Table I. To be fairly comparable with Content ID, we only report the recall from our system that is achieved at 100% precision, since through all of our experiments with YouTube the precision was 100%. As Table I shows, our system was able to detect 89 out of the 90 modified copies of the 3-D videos, including complex ones such as view synthesis. In summary, the results in this section show that: (i) there is a need for designing robust signatures for 3-D videos since the current system used by the leading company in the industry fails to detect most modified 3-D copies, and (ii) our proposed 3-D signature method can fill this gap, because it is robust to various transformations including new ones specific to 3-D videos such as anaglyph and 2-D-plus-depth format conversions as well as synthesizing new virtual views.

C. Performance of 3-D Signature Creation Component

We conduct small-scale, controlled experiments to rigorously analyze the proposed 3-D signature method. We need the experiment to be small because we manually modify videos in different ways and check them one by one. This is needed to compute the exact precision and recall of the proposed method. The reference video set contains the 14 videos mentioned in Section IV-A. The query set is created by modifying some of the reference videos in many different ways. Specifically, we apply the following transformations:

- **Video Blurring:** Reduces the sharpness and contrast of the image. Radius of blur is in the range of [3, 5];
- **Video Cropping:** Crops and discards part of an image. The discarded pixels are chosen at the boundaries. The number of discarded pixels is in the range [19, 40];
- **Video Scaling:** Reduces the resolution of the video and the scale factor is in the range [0.5, 1.5];
- **Logo Insertion:** Puts a logo on one of the corners of the video, the logo size is in the range of [19, 40] pixels;
- **Frame Dropping:** Periodically drops frames from the original video. The period is in the range [2, 9], where 2 means every other frame is dropped and period 10 means every tenth frame is dropped. This transformation changes the video frame rate;
- **Video Trans-coding:** Changes the video from one file format to another;
- **Text Insertion:** Writes random text on the video at different places;
- **Anaglyph:** Multiplexes the left and right views of the 3-D video over each other with different colors, typically red and blue;
- **Row Interleaved:** Interleaves the left and right views of the 3-D video horizontally row by row such that the odd rows belong to the left view and the even rows belong to the right view;
- **Column Interleaved:** Interleaves the left and right views of the 3-D video vertically column by column such that the odd columns belong to the left view and the even columns belong to the right view;
- **2-D-Plus-Depth:** Converts the video from left and right views stacked together horizontally side-by-side into another format which is a 2-D video and its associated depth;
- **View Synthesis:** Uses the original left and right views to create another two virtual views to be used instead of the original views.

We conduct two types of experiments: (i) individual transformations, in which we study the effect of each video transformation separately, and (ii) multiple transformations, in which we assess the impact of multiple transformations applied to the same video. In the first individual transformations experiments, we apply the above listed individual transformations (except view synthesis) on each of the 14 videos mentioned in Section IV-A using ffmpeg. View synthesis is applied using the VSRS view synthesis tool. View synthesis is applied on two videos other than the 14, which are Ballet and Break-Dancers. We create 18 different versions from each of these two videos with synthesized views. In the multiple transformations experiments, we choose 10 videos and apply on each of them three transformations at the same time. These transformations are blurring, scaling and logo insertion. Although the combined transformations are not likely to occur in many videos, they show the robustness of our method applying all these types of transformations on different videos results in a diverse and large query video set, which stresses our signature method.

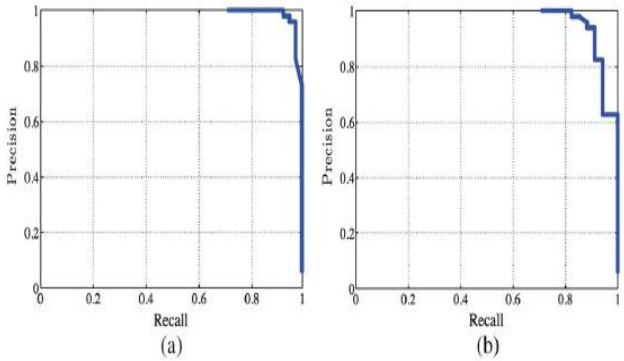

Fig.3. Average accuracy of the proposed 3-D signature method. (a) Single transformation. (b) Three transformations.

Results for Individual Transformations: We first present the average results across all videos and all transformations. We plot the aggregate results in Fig. 3(a) in the form of the precision versus recall curve. The precision-recall curve shows the achievable high accuracy of the proposed method. For example, a precision of 100% can be achieved with a recall up to 95%, by controlling the threshold value. To better understand the impact of the threshold parameter, we analyze the achieved average precision and recall by our method for all possible values of the threshold s. The figure is not shown due to space limitations. Our results show that the proposed method can concurrently achieve high precision and recall. For example, both the precision and recall are more than 90% for a large range of the threshold s. We note that this high accuracy is achieved when comparing significantly modified

copies of videos versus reference videos. Our method achieves 100% accuracy (both precision and recall) when we compare unmodified versions of the videos against reference videos.

Next, we analyze the accuracy of the proposed signature method for each video transformation separately. This is to understand the robustness of the method against each transformation. We computed the precision-recall curve for each case. In addition, we computed the precision and recall for each value of the threshold. Due to space limitations, we omit these figures. The results show that our method is highly robust against the quite common logo insertion transformation as it can achieve 100% precision and recall at wide range of thresholds. This is because a logo can affect one or a few blocks in the video frames, which is a relatively small part of the signature. Similar high accuracy results are achieved for two other common transformations: video blurring and trans-coding or format/bit rate change. In addition, for scaling, cropping, and frame dropping, our method still achieves fairly high accuracy. For example, our method is robust against video scaling. This is because during the creation of the signature, it is normalized by the frame resolution as described in Section IV. Moreover, for frame dropping our method achieves high precision and recall at low thresholds, which means that true matches are found but with low matching score, due to the gaps in the matching diagonal between both videos. Finally, for cropping, the results show that our signature method can identify 80% of the matches with 100% precision, or identify all matches with about 90% precision, which indicates that our method is also robust against cropping. This is because our 3-D signature contains the depth values of each block. Since depth is usually smooth, neighboring blocks usually have similar values, causing our 3-D signature to be less sensitive to block misalignments.

Results for Multiple Transformations: The aggregate results in the form of precision-recall curve are shown in Fig. 3(b). The results clearly show the high accuracy of the proposed method, as high precision of more than 90% can be achieved with at least a recall of 90%. We note that the recall in this experiment is slightly less than the achieved recall in the previous experiment, because of the combined transformations applied in this case.

Running Time for 3-D Signature Creation: In this experiment, we measure the running time of our 3-D signature creation and compare it to the method in [12], which requires computing the depth maps thus, we chose a practical depth estimation method we compute the running time for only the depth estimation step of the signature creation method in [12] and compare it to the whole running time of our method. We run the experiment for 362 frames on the same machine. The results show that the average running time for our 3-D signature creation is 0.87 sec, with minimum and maximum values of 0.39 sec and 1.62 sec, respectively. Whereas the average running time of depth estimation step alone is 68.91 sec, ranging from 61.26 sec and up to 85.59 sec. It can be seen that depth estimation method is far more expensive than our proposed signature extraction; the cost for just estimating the depth can be 100 times more than our signature extraction. As a result, [12] is only a suitable solution for 2-D-plus-depth formats where the expensive operation of depth estimation is not needed. Moreover, [12] uses the depth signature as a filtering step before the visual fingerprint to reduce the cost of computing visual signatures while this argument is valid for 2-D-plus-depth videos, this is not the case for stereo videos because computing the dense depth map will be more expensive than the visual signature. Our results show that a coarse-grained disparity map is robust against multiple kinds of transformations without compromising the precision, which suggests that computing the dense depth map is not needed for such a problem.

Effect of Frame Sampling on Performance: In order to speed up the matching process, simple techniques such as frame sampling can be used. With a sampling rate of 1/n, only one frame every n frames will get processed, therefore the matching process will get n times faster. In order to ensure the robustness of our system when using frame sampling, we repeated the individual transformations experiments with different sampling rates. The figures are omitted due to space limitations. Our results show that our system can achieve high accuracy even with low sampling rates, although the recall drops slightly with the decrease of sampling rate, because of the reduced amount of data. For example, we can achieve 92% recall at 96% precision with a sampling rate of 1/10, while without frame sampling we achieve a recall of 96% at 96% precision. That is, there is a loss of 4% in recall with a rate of 1/10. Also, with a sampling rate of 1/25 (about one frame per sec), we can achieve 86% recall at 92% precision, while without frame sampling the achieved recall at precision 92% is 96%. Thus, there is only a loss of 10% in recall with sampling rate of 1/25. As a result, we can speed up the matching process significantly, while still having high accuracy.

D. Accuracy and Scalability of the Matching Engine

We evaluate the accuracy and scalability of the matching engine component of the proposed system. We also compare our matching engine versus the best results reported by the Rank Reduce system, which is the closest to our work. Accuracy and Comparison Against Rank Reduce. We focus on evaluating the accuracy of the computed nearest neighbors, which is the primitive function provided by the engine. The accuracy of the retrieved K nearest neighbors for a point p is computed using the Precision @K (p) metric, which is given by

$$\text{Precision@K(p)} = \frac{\sum_{i=1}^{K}\{T_i \leq K\}}{K} \quad \textbf{(1)}$$

where T_i is the rank of a true neighbor T_i<=K. equals 1 if a true neighbor is within the retrieved K, and 0 otherwise. The average precision of the retrieved nearest neighbors across all points in the query set Q is

$$\text{AvgPrecision@K} = \frac{\sum_{i=1}^{|Q|}\{\text{Precision@K(i)}\}}{|Q|} \quad \textbf{(2)}$$

We use the AvgPrecision @K metric in our experiments.

We compare against Rank Reduce, which implements a distributed LSH index. It maintains a number of hash tables over a set of machines on a distributed file system, and it uses MapReduce for searching the tables for similar points. We compare the results achieved by our matching engine against the best results mentioned using the same dataset and the same settings. We did not implement Rank Reduce; rather we used the best stated results in its paper. We use the same dataset size of 32,000 points extracted from visual features of images. We measure the average precision at 20 nearest neighbors at the same percentage of scanned bins, which are called probed buckets in Rank Reduce terms. We plot the comparison results in Fig.4. The results first show that the proposed matching engine produces high accuracy, which is more than 95% by scanning less than 10% of the data. In addition, the results show that our matching engine consistently outperforms Rank Reduce, and the gain is significant (15–20%) especially in the practical settings when we scan 5–10% of the data points. For example, when the fraction of scanned data points is 5%, the average precision achieved by our engine is about 84%, while the average precision achieved by Rank Reduce is less than 65% for the same fraction of scanned data points. For Rank Reduce to achieve 84% average precision, it needs to scan at least 15% of the dataset (3X more than our engine), which incurs significantly more computation and I/O overheads.

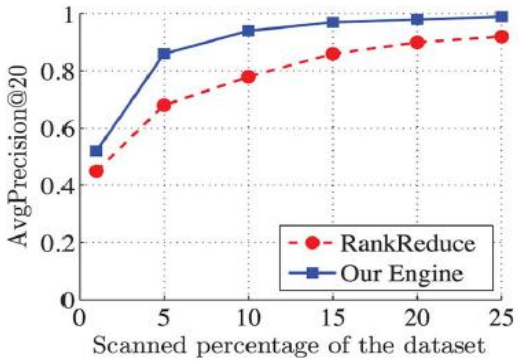

Fig.4. Comparing our matching engine versus the closest system in the literature, Rank Reduce.

In addition to the superior performance in terms of average precision, our engine is more efficient in terms of storage and computation. For storage, Rank Reduce needs to store the whole reference dataset multiple times in hash tables; up to 32 times. On the other hand, our engine stores the reference dataset only once in bins. Storage requirements for a dataset of size 32,000 points indicate that Rank Reduce needs up to 8 GB of storage, while our engine needs up to 5 MB, which is more than 3 orders of magnitude less. These storage requirements may render Rank Reduce not applicable for large datasets with millions of points, while our engine can scale well to support massive datasets. For computation resources, our engine and Rank Reduce use similar scan method to reference points found in bins or buckets. However, as discussed above, Rank Reduce needs to scan more buckets to produce similar precision as our engine. This makes our engine more computationally efficient for a given target precision, as it scans fewer bins.

Scalability and Elasticity of Our Engine: We conduct multiple experiments to show that our engine is scalable and elastic. Scalability means the ability to process large volumes of data, while elasticity indicates the ability to efficiently utilize various amounts of computing resources. Both are important characteristics: scalability is needed to keep up with the continuously increasing volumes of data and elasticity is quite useful in cloud computing settings where computing resources can be acquired on demand. We run our engine on datasets of different sizes from 10 to 160 million data points, and on clusters of sizes ranging from 8 to 128 machines from Amazon. These data points are visual features extracted from 1 million images download from Image Net [8]. From each image, we extract up to 200 SIFT features, which results in a dataset of up to 200 million data points. In all experiments, we compute the K=10 nearest neighbors for a query dataset of size 100,000 data points. We measure the total running time to complete processing all queries, and we plot the results in Fig.5. The figure shows that our engine is able to handle large datasets, up to 160 million reference data points are used in creating the distributed index. More importantly, the running time grows almost linearly with increasing the dataset size on the same number of machines. Consider for example the curve showing the running times on 32 machines.

The running times for the reference dataset of sizes 40, 80, and 160 million data points are about 40, 85, and 190 minutes, respectively. In addition, the results in Fig. 5 clearly indicate that our engine can efficiently utilize any available computing resources. This is shown by the almost linear reduction in the running time of processing the same dataset with more machines. For example, the running times of processing a reference dataset of size 80 million data points are 160, 85, 52, and 27 minutes for clusters of sizes 16, 32, 64, and 128 machines, respectively. The scalability and elasticity of our engine are obtained mainly by our design of the distributed index, which partitions the datasets into independent and non-overlapping bins. These bins are allocated independently to computing machines for further processing. This data partitioning and allocation to bins enable flexible and dynamic distribution of the computational workload to the available computing resources, which is supported by the MapReduce framework.

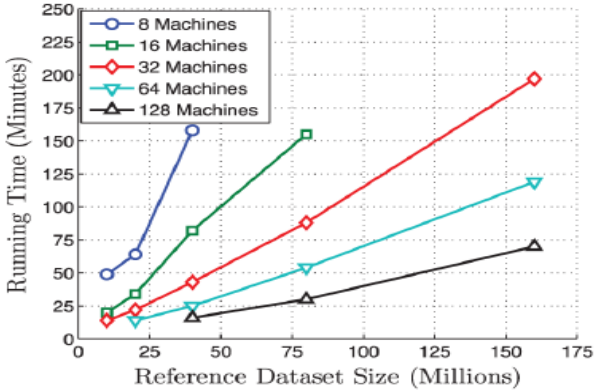

Fig.5. Running times of different dataset sizes on different number of machines.

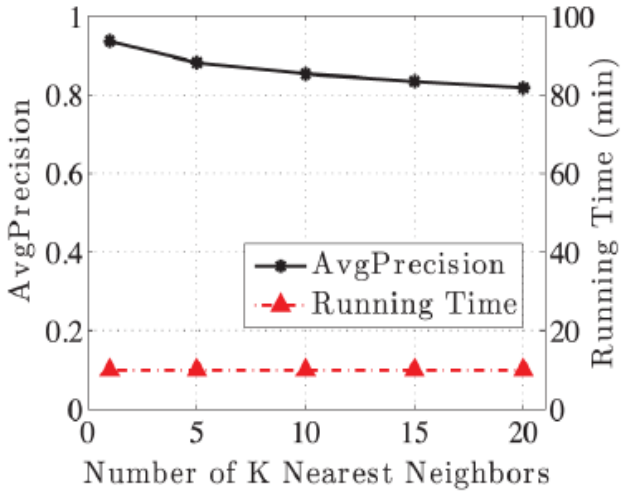

Fig.6. Effect of on running time and accuracy.

Effect of Number of K nearest Neighbors: In this experiment, we study the effect of changing the number of K nearest neighbors retrieved. We measure the running time and the average precision for different values of K, while maintaining a fixed scanned percentage of the reference dataset. The results are plotted in Fig.6, which show that while we achieve high precision for returning the closest neighbor (i.e., K =1), with value of 94%, the average precision achieved is not significantly impacted by increasing K. For example at K =5 the average precision is 88%, and at K =20 the average precision is 82%, losing only 6% of the precision when returning 4 times more neighbors. The results also show that the effect of K on running time is negligible, since running time is mainly related to the size of the scanned data points.

V. CONCLUSION

Dispersing copyrighted interactive media objects by transferring them to internet facilitating destinations, for example, YouTube can bring about critical loss of incomes for substance makers. Frameworks expected to discover illicit duplicates of sight and sound items are perplexing and extensive scale. In this paper, we exhibited another outline for sight and sound substance assurance frameworks utilizing multi-cloud foundations. The proposed framework bolsters distinctive sight and sound substance sorts and it can be conveyed on private and/or open mists. Two key parts of the proposed framework are exhibited. The first is another technique for making marks of 3-Dvideos. Our technique develops coarse-grained uniqueness maps utilizing stereo correspondence for a meager arrangement of focuses in the picture. Subsequently, it catches the profundity sign of the 3-D video, without unequivocally processing the accurate profundity map, which is computationally costly. Our examinations demonstrated that the proposed 3-D signature delivers high exactness as far as both accuracy and review and it is powerful to numerous video trans-arrangements including new ones that are particular to 3-D recordings, for example, blending new perspectives.

The second key segment in our framework is the dispersed file, which is utilized to match mixed media objects portrayed by high measurements. The conveyed file is actualized utilizing the Map Reduce outline work and our analyses demonstrated that it can flexibly use fluctuating measure of figuring assets and it creates high exactness. The trials additionally demonstrated that it beats the nearest framework in the writing regarding precision and computational productivity. Likewise, we assessed the entire substance insurance framework with more than 11,000 3-D recordings and the outcomes demonstrated the versatility and precision of the proposed framework. At long last, we thought about our framework against the Content ID framework utilized by YouTube. Our outcomes demonstrated that: (i) there is a requirement for planning powerful marks for 3-D recordings since the present framework utilized by the main organization as a part of the business neglects to recognize most adjusted 3-D duplicates, and (ii) our proposed 3-D signature strategy can fill this hole, since it is vigorous to numerous 2-D and 3-D video changes.

VI. REFERENCES

[1] Mohamed Hefeeda , Senior Member, IEEE, Tarek ElGamal , Kiana Calagari, and Ahmed Abdelsadek, "Cloud-Based Multimedia Content Protection System", IEEE Transactions on Multimedia, Vol. 17, No. 3, March 2015.

[2]A.Abdelsadek, "Distributed index for matching multimedia objects," M.S. thesis, School of Comput. Sci., Simon Fraser Univ., Burnaby, BC, Canada, 2014.

[3] A. Abdelsadek and M. Hefeeda, "Dimo: Distributed index for matching multimedia objects using MapReduce," in Proc. ACM Multimedia Syst. Conf. (MMSys'14), Singapore, Mar. 2014, pp. 115–125.

[4] M. Aly, M. Munich, and P. Perona, "Distributed Kd-Trees for retrieval from very large image collections," in Proc. Brit. Mach. Vis. Conf. (BMVC), Dundee, U.K., Aug. 2011.

[5] J. Bentley, "Multidimensional binary search trees used for associative searching," in Commun. ACM, Sep. 1975, vol. 18, no. 9, pp. 509–517.

[6] P. Cano, E. Batle, T. Kalker, and J. Haitsma, "A review of algorithms for audio fingerprinting," in Proc. IEEE Workshop Multimedia Signal Process., Dec. 2002, pp. 169–173.

[7] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," in Proc. Symp. Oper. Syst. Design Implementation (OSDI'04), San Francisco, CA, USA, Dec. 2004, pp. 137–150.

[8] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-8ei, "Image net: A large-scale hierarchical image database," in Proc. IEEE Conf. Comput. Vis. Pattern Recog. (CVPR'09), Miami, FL, USA, Jun. 2009, pp. 248–255.

[9] A. Hampapur, K. Hyun, and R. Bolle, "Comparison of sequence matching techniques for video copy detection," in Proc. SPIE Conf. Storage Retrieval Media Databases (SPIE'02), San Jose, CA, USA, Jan. 2002, pp. 194–201.

[10] S. Ioffe, "Full-length video fingerprinting. Google Inc.," U.S. Patent 8229219, Jul. 24, 2012.

[11] A. Kahng, J. Lach, W. Mangione-Smith, S. Mantik, I. Markov, M. Potkonjak, P. Tucker, H. Wang, and G. Wolfe, "Watermarking techniques for intellectual property

protection," in Proc. 35th Annu. Design Autom. Conf. (DAC'98), San Francisco, CA, USA, Jun. 1998, pp. 776–781.
[12] N. Khodabakhshi and M. Hefeeda, "Spider: A system for finding 3D video copies," in ACM Trans. Multimedia Comput., Commun., Appl. (TOMM), Feb. 2013, vol. 9, no. 1, pp. 7:1–7:20.

Author's Profile:

Sirisha.A Department of CSE with Annamacharya Institute of Technology And Sciences, Hyderabad, India,
Email: siri04vinny@gmail.com.

Mr.Ramesh Babu Varugu, received the Master of Technology degree in Information Technology from the Gurunanak Institute of Science and Technology-JNTUH, he received the Bachelor of Technology degree from Lakireddy Balireddy College of Engineering, JNTUK. He is currently working as Associate Professor and a Head of the Department of CSE with Annamacharya Institute of Technology and Sciences, Hyderabad. His interest subjects are operating Systems, Cloud Cimputing and etc, Email: ramesh.vnl@@gmail.com.