PROPOSED METHOD

This section explains the system proposed to recognize whether the blind video obtained as an input is forged or unforged. In this Multi-Filter Algorithmic System [Figure 1], the processing of Video for detection of Copy-Move Attack is divided in multiple phases such as Preprocessing, Detection of all the Objects in the video, Tracking of all the Objects in the video and finally applying S4 filters on the Objects tracked in the video to detect the pair of Forged and Original object if present and hence conclude whether the video is forged or not.

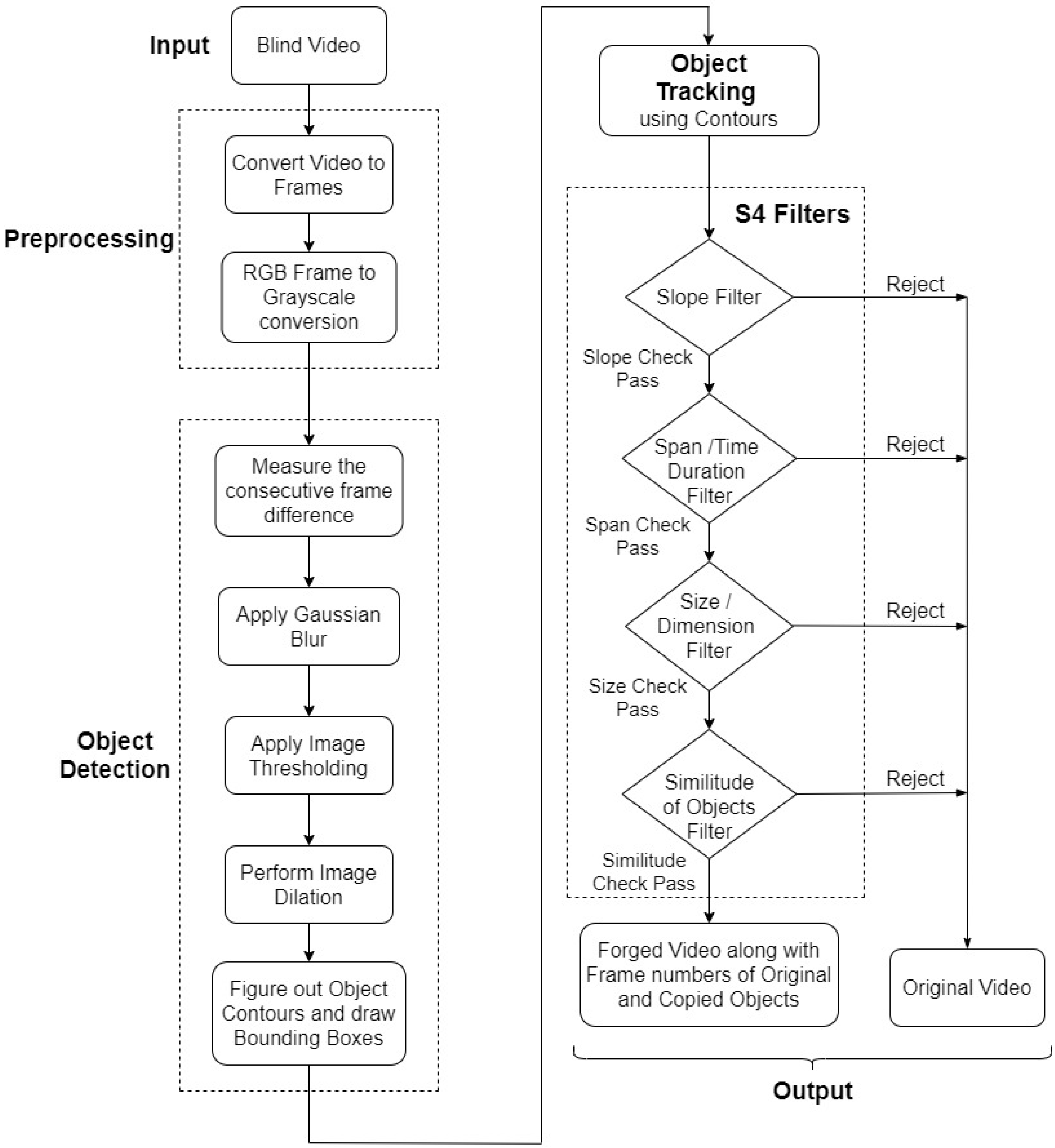


Figure 1. Proposed System

1. Preprocessing phase: -

The Proposed System's performance has already been evaluated on the standard Copy-Move Forgery dataset SULFA (Surrey University Library for Forensic Analysis) [6] as well as on the Handcrafted Custom Dataset. The Proposed System has been successfully tested on videos having different file formats as .avi, .mp4, etc. In the preliminary phase, the input video obtained is converted into cluster of frames. Later on, conversion of frames from RGB to GrayScale colour system is to be performed as shown in [Figure 3]. For this conversion Average Method can be used in which we simply take average of Red, Green and Blue values of each pixel of the frame to obtain that pixel's value in GrayScale output frame. But the GrayScale image obtained by this method is less accurate as each colour of RGB system has varying wavelength and thus varying contribution to the image construction. So, a more accurate method is used in converion of a RGB image to a GrayScale image that is Luminosity Method which assigns weight to each colour of RGB based on their respective wavelength contribution and thus overcome the drawback of Average Method.

Pixel value of the GrayScale image can be obtained from (R,G,B) value of the pixels by the Eq.(1) (https://en.wikipedia.org/wiki/Grayscale) as given below,

$$\textbf{GrayScale\_Image} = \textbf{0.299*Red} + \textbf{0.587*Green} + \textbf{0.114*Blue} \qquad \textbf{(1)}$$

Figure 2. RGB Frame

Figure 3. GrayScale Frame

2. Object Detection phase: -

All the objects in the video are detected using Contour Based Object Detection Technique. Initially difference of consecutive GrayScale frames is calculated as shown in [Figure.4]. Changing of pixels in the consecutive frames highlight that there apparently was something changing in the image (something in motion for example) which helps in object detection. Then apply Gaussian Blur or Gaussian Smoothing to blur the image obtained after the difference of consecutive GrayScale frames. Its prime purpose is to reduce noise as well as reduce unnecessary details to minimize computations. Image Noise is the granular random variation in the brightness and color of some pixels that interfere in image processing functioning and accuracy. Gausssian Blur works as the convulation of two-dimensional input image pixel data with the pre-computed

Gaussian Kernel. All pixels are assigned weighted average of the neighbouring elements, the closer the pixel is to the center would have higher weight and weight would reduce as the focal distance increases from the center. Gaussian Filter basically filters out or reduces high frequency constituents of the image and is therefore low pass filter. Gaussian Blur is similar to the concept of Normal distribution in Statistics, it utilizes similar characteristics to perform manipulation of all the pixels in the frame and leading to image blurring as shown in [Figure 5].

Gaussian Function Eq.(2) (https://en.wikipedia.org/wiki/Gaussian_function) is given below,

$$G_0(x, y) = Ae^{\frac{-(x-\mu_x)^2}{2\sigma_x^2}+\frac{-(y-\mu_y)^2}{2\sigma_y^2}} \tag{2}$$

Here μ is the mean, σ_x and σ_y is the standard deviation for each pair of (x,y) in x and y direction respectively.

Later, Image Thresholding is applied on this blurred GrayScale image. To obtain an efficient object detection, binary images are used which is obtained through Image Thresholding. It's a technique to convert a GrayScale image to a Binary image where a pixel can take any one of the two values that is 1(pure white) or 0(pure black) as shown in [Figure 6]. Thus, thresholding limits the complexity of image data and simplifies more accurate recognition of objects. For a resultant thresholded image that is Binary_image (x, y) thresholding function is defined by Eq. (3) as,

$$\mathbf{Binary\_Image}(\mathbf{x}, \mathbf{y}) = \begin{cases} \mathbf{1}, & \mathbf{GrayScale\_Image}(\mathbf{x}, \mathbf{y}) > \mathbf{Threshold} \\ \mathbf{0}, & \mathbf{GrayScale\_Image}(\mathbf{x}, \mathbf{y}) \leq \mathbf{Threshold} \end{cases} \tag{3}$$

Here, In the proposed system value of Threshold is taken as 20.

Later, Morphological operation is performed that mainly deals with structural geometry of the image. Morphological operations mainly consist of Dilation (grow operation) and Erosion (shrink operation). The main motives behind the use of Morphological operation that is Dilation operation is,

I. Noise Removal from the image.
II. Shape Simplification by separation of individual elements and merging similar elements in the image.
III. To figure out intensity clusters or holes in the picture.

Image Dilation is a technique of expansion of the image clusters which depends on the structuring element. Structuring element is made up of matrix of 1's and 0's. The shape and size of the Structuring element mainly depends on number of pixels to be added. Dilation operation adds pixels to the boundary of the object in an image. Dilation consist of an image P with kernel (Q), which can have diverse shape or size, generally a circle or square. The center of kernel Q is known as anchor point. Later, the kernel is coincided over the thresholded binary image to evaluate the maximum pixel values. After the evaluation, the image is replaced with an anchor at the center as shown in [Figure 7]. Thus, the brighter areas increase in size that eventually leads to the increment in the size of the image.

Dilation is performed using mathematical Eq. (4) as given below,

$$\textbf{Dilation\_input } (a+a', b+b')$$

$$\textbf{Dilation\_output } (a, b) = \max (a', b'): \text{element } (a', b') \neq 0 \tag{4}$$

Here, (a, b) is a co-ordinate of image P pixel and (a', b') is a co-ordinate of pixel in kernel Q.

Finally, Contour are to determined from the dilated binary image as shown in [Figure 8]. All the continuous boundary points having similar intensity or color can be linked using curvature known as contour which is further useful for object detection. Finding contour requires preprocessing of image by thresholding or edge detection by canny algorithm. Edges and Contours are commonly used as synonyms but they are different, Edge has local scope in the image while the Contour has global scope as it's used to identify overall boundary of the object. Contour Detection is performed by Topological Structural Analysis by Border Following [5]. Border Following algorithm's main purpose is to figure out boundary pixels from the blob of objects in the dilated image. It forms a linkage of border between a connected component of white pixels (object) and a connected component of black pixels (background or hole). It segments image based on Superpixels, fuzzy k-mean. Topological Analysis signify scanning of the image to figure out relative ordering of chain of pixels (boundary) and identify whether the boundary is outer bounndary or hole boundary. Unique identifier is allocated to each boundary as well store the recent parent boundary of all the boundaries found. Eventually, these boundaries form the desired contours.

Figure 4. Difference of Consecutive GrayScale Frame

Figure 5. Gaussian Blur

Figure 6. Image Thresholding

Figure 7. Image Dilation

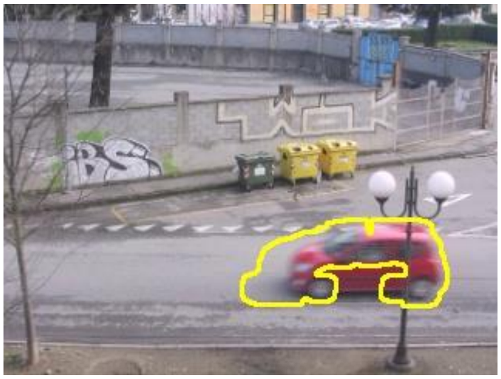


Figure 8. Finding Object Contour

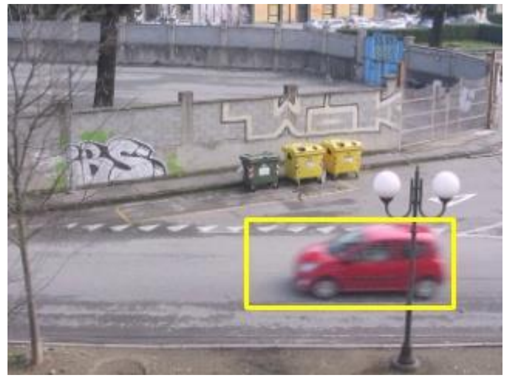


Figure 9. Draw Rectangular Bounding Box

All the contours obtained are later filtered based on their contour areas to remove redundant, unnecessary and extraneous contours mainly generated by external noise of the image. In our proposed system, we figured out 900 as the most pertinent Contour Area Threshold for all the SULFA Dataset [6] and custom dataset video object contours. Finally, Rectangular Bounding Boxes are to be drawn as shown in [Figure 9] around all the significant object contours having contour area above the Contour Area Threshold.

3. Object Tracking phase: -

Tracking of all the respective objects in the video is done using contours. In the proposed system, we analyze the relative position of the contours in the consecutive frames for same object. Objects can be checked for the Similarity by comparing Structural Similarity Index (SSIM) [7] of the objects present inside the Bounding Boxes in the consecutive frames and as the Bounding boxes are drawn only along the moving objects our computation is minimized to only moving objects. SSIM is used for figuring out the similarity between any two images and is discussed with great details in the Similtude Filter of the S4 filters phase below. Utilizing the observation that for a moving object in the video, the center (c_i) of the Rectangular Bounding Box in the current frame (f_i) always lies inside the Rectangular Bounding Box of the preceding frame (f_{i-1}). Thus by storing of the bounding box co-ordinates (x, y, width, height) of the object in the previous frame and evaluating the center of the same object's bounding box in current frame we can track the next position of the object's path depending on if the center in current frame lies inside the bounding box co-ordinates of the previous frame and the object is same across both frames which is checked using SSIM.

Figure 10. Working of the Path Tracking

Figure 11. Complete Tracked Path of the Object

As show in the [Figure 10], the rectangular bounding box in yellow color having center C denotes the object's position in the current frame while the rectangular box in red color indicates the object's position in the previous frame. So here first we need to check whether the object's inside bounding boxes from the consecutive frame is same or not using Structural Similarity Index [7]. If same then check if the center C of the object's bounding box in the current frame (yellow box's center) lies inside the bounding box of the previous frame (red box). If both the conditions are satisfied, then we conclude that the center C is the next location of the object's path and connect it using white line. The complete path of the object tracked is displayed using white color as shown in [Figure 11]. This path will be used later in S4 filters for classifying whether the object is copied/forged or not. Contours based tracking gives excellent results with the stable enviornment having one or multiple objects moving simultaneously in the video as object contours can be detected more precisely. This Object Tracking technique has successfullly tracked all the objects in the Benchmark dataset SULFA [6] as well as our Handcrafted dataset with highest precision. It's also capable of tracking Multiple Objects moving in a video simultaneosly.

4. S4 Filters phase: -

In this Proposed Algorithm, we are proposing a Novel S4 Filters technique for the detection of pair of original and copied objects as well as Novel way of Tracking of the objects path based on contours and SSIM. S4 filters signify Slope, Span, Size and Similitude Based Filters. They are signified as filters as they are used to filter out the correct pair of Original and Copied object in Copy-Move Tampered video if present. In this tampering an object and its complete motion in the video is copied and moved/ pasted in some other set of frames in the same video. For the detection of such a precise attack we need to first figure out all the moving objects in the video and later check which two objects along with their motion are copy of each other. This Algorithm can also be implemented in an iterative manner to resolve detection problems of the videos having Multiple pair of different Original-Copied pair of objects or Single object with its Multiple copies in the same video. An image has some characteristic features as edges, histogram representation of the intensity, blobs and ridges which can be used for detection or recognition purpose. Similarly, an original and copied object in the video also has some implicit and immutable features common features as regular interval slope of the path object takes in the video, duration of time of the object in the video, size of the object and most important as it is the same object then higher value of Structural Similarity.

In this Proposed Algorithm, as we have already traced the path using Contours and SSIM, now we need to evaluate and compare regular interval slope of the path taken by all the objects detected in the video which act as the first S4 filter for detecting the orginal and copied pair of object. For that, complete path of each object has to be stored by sequentially storing the center C (x, y) of the bounding boxes of that object in each frame which it appears. Such as storing C_1 (x_1, y_1) of the bounding box (x_1, y_1, w_1, h_1) of first frame in which object enters the video where (x, y, w, h) represents x-coordinate, y-coordinate, width and height of the bounding box respectively, then storing C_2 (x_2, y_2) of the bounding box (x_2, y_2, w_2, h_2) of the second frame in which it appears and continue storing till that object exists the video. After having the complete path co-ordinates stored for each object, we have evaluated the slope at regular interval for that object. The path co-ordinates are divided into **(Number of coordinates / 5)** regular intervals of size 5 and any co-ordinates are remaining they form a last intervel eg, if any object's path consists of 43 co-ordinates then it would have (43/ 5) i.e, 8 regular intervals having size 5 and one more extra interval having remaining 3 co-ordinates and in total 9 intervals. Then slope has to be calculated for each interval and stored, for the path of all the objects in the video. If the last interval of the object's path has only 1 co-ordinate, then the slope of the last interval is considered to be 0. Now we need to compare the objects by finding out the objects having same number of intervals and if number of intervals are same then later compare the slope value of each interval of an object with the slope value of the corresponding number of interval of the other object. Finally, we might get a group of objects which have passed this filter by having same number of path intervals and also same slope at each corresponding number of regular interval and become eligible for the next Span Filter. If no group of objects satisfying both the conditions is found, then video can be classified as Original Video as it does not have any pair of Original and Copied Objects alongwith the same motion.

Slope of i^{th} interval of the path can be calculated by Eq. (5) as,

$$\boldsymbol{Slope}_{\,i^{th}\,interval} = \frac{(y_b - y_a)}{(x_b - x_a)} \tag{5}$$

Where (x_a, y_a) and (x_b, y_b) are the first and the last co-ordinates of the i^{th} interval respectively.

Span Filter basically checks the Span or the Duration of the Object's presence in the video. Consider a scenario in which a car and a truck are moving on the road following the same path, as their path is same so same would be there regular interval slope hence, they will surpass first filter and come to the Span filter. Here Duration is calculated by subtracting object's exit frame number and object's entry frame number. Entry frame number is the first frame in which object appears and exit frame number is the first frame in which object disappears. Comparing number of intervals in the first Slope filter is not same as comparing Duration. As in the Slope filter last interval has all the remaining co-ordinates, so objects having same interval can have different durations for eg, Consider object N1 with 36 co-ordinates and object N2 with 39 co-ordinates both have same number of intervals in Slope filter phase that is 8 but both N1 and N2 will have different Span's.

Span can be calculated by the Eq. (6) which is given below,

$$\boldsymbol{Span}_{\,i^{th}object} = \boldsymbol{Exit\_Frame}_{No} - \boldsymbol{Entrance\_Frame}_{No} \tag{6}$$

Thus, we need to calculate the Span of all the objects are compare it to obtain a cluster of objects fullfilling second filter and proceed them towards Size Filter. If no pair of objects is found the video can be concluded as Original Video.

In the third phase of S4 Filters, Size or Dimensions of the object are equated. Now consider the same scenario of the car and the truck, if the regular interval Slope and the Span both of the car and truck are same then it will successfully bypass intial 2 filters of the S4. But as here in the Size filter, dimensions of the bounding box in the first frame of the detection of all the objects are compared. Factually, if the original object is copied alongwith it's motion in the same video then Copied object must be discovered with the same dimensions as the contour-based detection algorithm acts in equivalent way for the original and copied object. But a Dimension_tolerance is declared to contain a scenario in which dimensions of Original and Copied object can be obtained with minor tolerance/ difference because CPU frame processing delay due to scheduling or detection of Multiple objects simultaneously. At the same time, objects whose size have some major variations more than the Dimension_tolerance are filtered out. As discussed in the example above, Car and Truck which have bypassed earlier 2 filters will get filtered out here due to large variations in their dimensions.

Objects Obj_1 and Obj_2 will qualify Size filter if they satisfy conditions (7) and (8) given below,

Condition 1: - (Width_Obj$_2$ – Dimension_tolerance) <= Width_Obj$_1$ and
Width_Obj$_1$ < (Width_Obj$_2$ + Dimension_tolerance) (7)

Condition 2: - (Height_Obj$_2$ – Dimension_tolerance) <= Height_Obj$_1$ and
Height_Obj$_1$ < (Height_Obj$_2$ + Dimension_tolerance) (8)

In our Implemented System, Dimension_tolerance = 4 gave highest accuracy for the Standard SULFA as well as Custom Dataset. The group of objects satisfying the Size or Dimension filter eventually qualify for the final and most decisive Similitude of the Objects filter. And if no group of objects qualify this filter video is termed as Original Video.

Post Slope, Span, and Size filtering of the group of all the objects detected in the video, finally Similitude based filtering is done. Similitude or Similarity of the object's are figured out using Structural Similarity Index [7]. Structural Similarity Index gives the measure of resemblance between 2 images. As the Original and Copied Object are the same, so the images of the objects in the bounding boxes must also be same in their corresponding entrance frames in the video. Input images to SSIM for matching of all the objects, are taken of the size of object's bounding box ignoring the rest of the background from the frame when they entered the video. SSIM basically evaluates the similarity by means of 3 major comparisons that are Luminence, Structural and Contrast Comparison (LSC Comparison). Prior well-known techniques such as MSE or PSNR used the concept of estimating absolute errors whereas Structural Similarity implements model which utilizes perception that interprets the image degradation as apparent change in the structural information of the image. The Structual information implies that pixels geometrically closer are highly inter-dependent and this inter-dependence carry some of the most crucial information about the structures of the objects in the image. The Luminance of the image dependes on reflectance and illumination, but structual information is independent of this. So, SSIM represents structure of the objects independent of the luminence and contrast. The SSIM results are obtained on the scale of -1 to 1 where SSIM as 1 of two images denotes complete Similarity between them and it's vice versa SSIM value as -1 denotes complete dissimilarity.

$$\boldsymbol{SSIM(x, y) = \frac{(2\mu_x\mu_y + c1)(2\sigma_{xy} + c2)}{(\mu_x^2 + \mu_y^2 + c1)(\sigma_x^2 + \sigma_y^2 + c2)}} \quad \textbf{(9)}$$

With:
μ_x is the average of x
μ_y is the average of y
σ_x^2 is the variance of x
σ_y^2 is the variance of y
σ_{xy} is the covariance of x and y
$c1 = (k_1L)^2, c2 = (k_2L)^2$ two variables to balance the division with less value denominator
L is the dynamic range of pixel values
$k_1 = 0.01$ and $k_2 = 0.03$ is by default value.

Formula components: - The SSIM formula is based on three comparison measurements between the sample of {\displaystyle x} x and y: luminance (l {\displaystyle l}), contrast (c {\displaystyle c}) and structure (s {\displaystyle s}).

The individual comparison functions are:

$$l(x,y) = \frac{(2\mu_x\mu_y + c1)}{(\mu_x^2 + \mu_y^2 + c1)}$$

$$c(x,y) = \frac{(2\sigma_{xy}+c2)}{(\sigma_x^2+\sigma_y^2+c2)} \tag{10}$$

$$s(x,y) = \frac{(\sigma_{xy} + c3)}{(\sigma_x\sigma_y + c3)}$$

Where $c3 = c2/2$

Experimental Results and Analysis

The proposed methodology has been tested on benchmark SULFA [6] dataset as well as 30 handcraft videos which has been created by software-Adobe Premier Pro CC. The 20 videos are taken from standard dataset, with each video having frame-rate of 30 fps and resolution of 320*240 pixels. Few handcrafted videos are filmed on mobile having EIS (Electronic Image Stabilization) as EIS technology is gaining popularity and also provides greater stability to shaky videos that is moving camera videos. Thus, testing the system on robust dataset so as to obtain more accurate results and thereby covering all scenarios. The proposed system was implemented on system having following configuration.

Processor: Intel(R) Core (TM)2 i5-4700MQ 2.4GHz;
Primary Memory: 8GB;
Graphics card: AMD Radeon™520
OS: Microsoft Windows 10
Platform/Tools used: Python 3

1.1 Detection result of a tampered area with respect to S-4 filters: -

The experiment was conducted on sample videos some are taken from standard dataset and some are handmade forged videos. The Table 1 and Table 2 shows the S-4 filters that is slope, span, scale and similitude of the different videos of SULFA [6] and handcrafted custom videos. For the sake of simplicity, Slope value taken over first interval is only shown in the table.

Table 1: Video Samples from Benchmark dataset SULFA [6] with corresponding S-4 Parameters

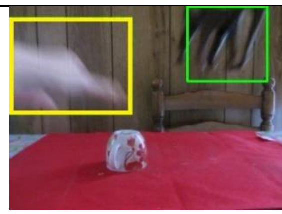
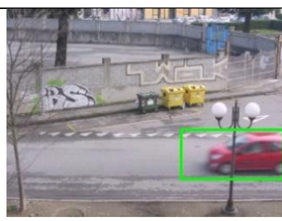
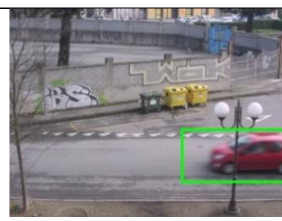
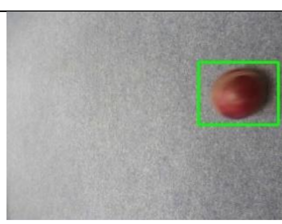
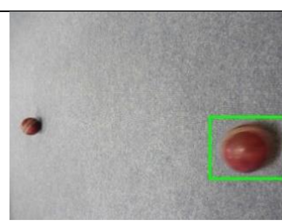
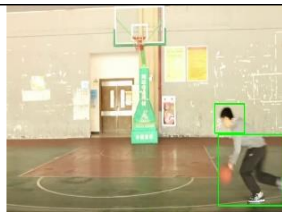
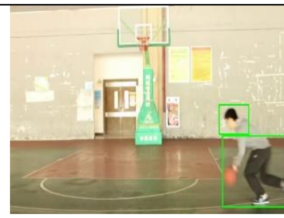
	Original Frame	Duplicated Frame	Slope of First Interval	Span	Size (Height* Width)	Similitude Index
Video 1	 Frame No.304	 Frame No.499	1.167	52	40*41	0.983
Video 2	 Frame No.13	 Frame No.215	-0.631	24	96*88	0.877
Video 3	 Frame No.95	 Frame No.370	0.161	11	119*57	0.952
Video 4	 Frame No.35	 Frame No.118	-0.454	19	118*35	0.947
Video 5	 Frame No.52	 Frame No.237	-0.384	42	92*72	0.851

Table 2: Video Samples from handcrafted dataset with corresponding S-4 Parameters

	Original Frame	Duplicated Frame	Slope of First Interval	Span	Size (Height* Width)	Similitude Index
Video 1	 Frame No.45	 Frame No.145	-0.267	21	77*57	0.995
Video 2	 Frame No.44	 Frame No.247	2.258	19	74*183	0.987
Video 3	 Frame No.85	 Frame No.458	0.240	17	167*202	0.987
Video 4	 Frame No.26	 Frame No.354	1.939	23	106*165	0.987
Video 5	 Frame No 26	 Frame No.212	-0.803	21	169*168	0.995

1.2Comparative Analysis of Proposed System with Existing

Table 3: Performance Evaluation of System with existing (Precision, Recall, F1-Score and Detection Accuracy%)

State of the Art	Precision	Recall	F1-Score	Accuracy
Wang [1]	0.385	0.817	0.523	65.80%
Tan [9]	-	-	-	83.30%
Joshi [2]	0.833	0.898	0.864	87.50%
Su [4]	0.952	0.514	0.668	92.60%
Lin [3]	0.879	1.00	0.935	94.00%
Li [8]	0.997	1.000	0.997	-
Proposed System	1.000	0.980	0.989	98.00%

$$\mathbf{Precision} = \frac{\mathbf{A}}{\mathbf{A+B}} \tag{11}$$

$$\mathbf{Recall} = \frac{\mathbf{A}}{\mathbf{A+C}} \tag{12}$$

$$\mathbf{F1\text{-}Score} = \frac{\mathbf{2*Precision*Recall}}{\mathbf{Precision+Recall}} \tag{13}$$

Here A represents all True Positives which means the authenticate videos or Forged videos are marked accurately means, suppose video is forged and if proposed system labels it as forged then those are true positives. B represents False Positives means if video is not forged but system somehow manages to label it as forged then it is false positive. C represents False Negatives means if for particular forged video if algorithm claims that video is authentic then that is case of false negative. Total 50 videos with each video having frames between ranges of 300 to 500 are tested. The tests are conducted on two datasets as mentioned earlier, one is standard dataset SULFA [6] and the other is customized and then final result is combined to obtain the performance metrics. As conformance to structural similarity index score (ssi_score) threshold of 0.5, each copied portion's ssi_score is above the threshold value as it can be seen from Table 1. The algorithm mainly relies on object detection technique so more robust object detection module can enhance the system's performance giving F1-score greater than 98% as can be seen from Table 3. The videos with greater static stability can reach more higher precision and recall rate.

For the purpose of comparative analysis ,the various video forgery detection methodologies are considered, they are - the methods proposed by Wang W, Farid H (2007), Lin GS, Chang JF (2012), Li F., Huang T. (2014) , S. Tan, S. Chen and B. Li(2015), Su, L., Li, C.(2018), Joshi, V.,AND Jain, S(2020).All these methodologies are compared on the basis of Precision, Recall and Detection Accuracy(DA)

$$\textbf{Detection Accuracy (DA)} = \frac{A+D}{A+B+C+D} \qquad \textbf{(14)}$$

Where A, B, C, D stands for True Positive, True Negative, False Positive and False Negative respectively.

The Figure 1, Figure 2, Figure 3 and Figure 4 compares the Detection Accuracy, Precision, Recall and F-score of proposed system with existing ones. The methods define by Wang [1] and Su [4] fails to determine the mirror operations thus resulting in poor detection accuracy rate. The methodologies proposed by Lin [3] heavily relies on candidate selection, faults in determination of candidate selection results in less detection accuracy. The method based on Group of Pictures (GOP) given by Tan [4] gives poor performance of standard SULFA [6] dataset. The Proposed algorithm also outperforms the method given by Joshi [2] and Li [8] based on structural similarity score.

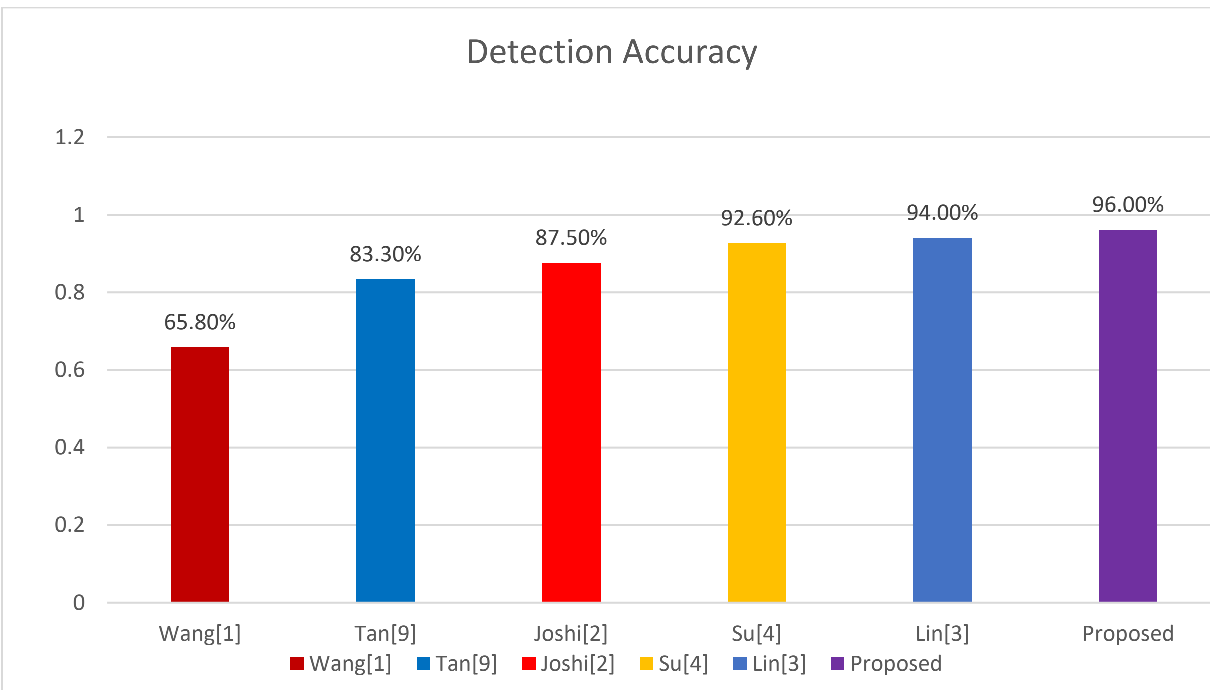


Figure 4: Comparison of DA

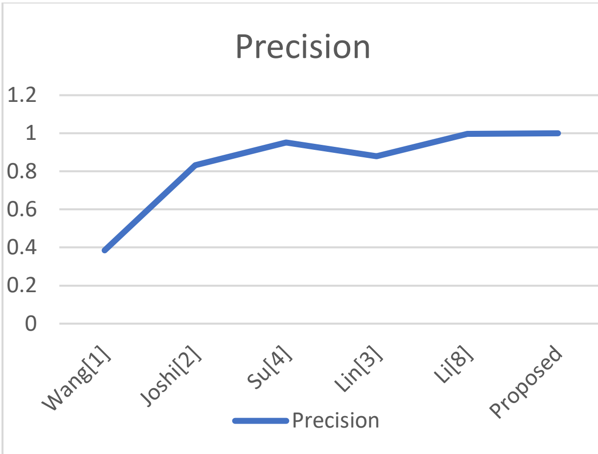


Figure 2: Comparison of Precision

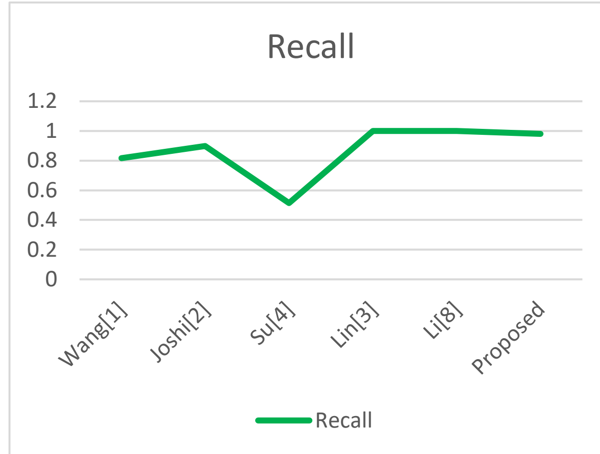


Figure 3: Comparison of Recall

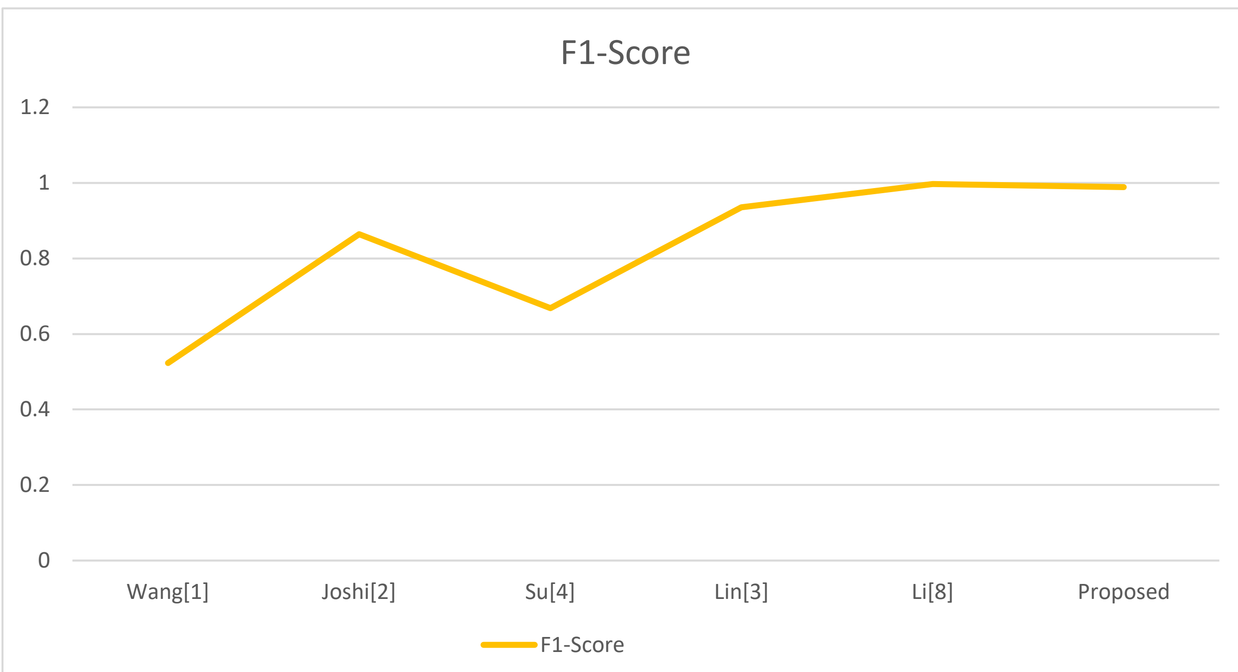


Figure 4: Comparison of F1-Score

Conclusions and Future Work

In this paper, we proposed novel approach for detection of copy-move attack in temporal domain. At abstract level, the system is divided into three phases. First phase is preprocessing which involves conversion of video into frames and in grayscale. The second phase is object detection and object tracking phase. This phase is primarily based on frame difference and there by detecting the object contours, hence it is computationally inexpensive. Later, the video sequence must suffice S-4 filters. The specialty of this scheme is, among all the tracked objects in the frames, only those objects who fulfill initial three filters are eligible for similarity matching and if any one of filter fails it directly predicts given video sequence as authentic. Also, the first three filters have algorithmic time complexity O(1), thus giving better performance than all other existing methods.

These filters are-

1) Slope(S-1) of moving object which is simply, the rate of change of its speed which is going to stay same for copied objects.

2) Span(S-2) also known as duration, is the time for which object persists in video or simply frames across which it exists which is going to stay the same for copied regions.

3) Size(S-3) which is in terms of Height*Width of contour box of detected object as mentioned earlier which will also be same in both original and tampered frame.

4) Similitude Index(S-4) which is similarity measurement is used to ensure that the two copied objects are similar so as to confirm the presence of copy move attack.

In this way, the simplicity of proposed system gives not only 98% accuracy and 100% precision rate but also is independent of video format used, thereby giving enhanced approach to detect copy-move attack. For the future work, the system can also be enhanced in terms of object detection and tracking phase by using machine learning advancements like convolutional neural network (CNN) for performance gain. Also, goal is to build copy-move attack dataset.

References

1. Wang W, Farid H (2007) Exposing digital forgeries in video by detecting duplication. In: Proceedings of 9th workshop on Multimedia & Security, pp. 35–42.

2. Joshi, V., Jain, S. Tampering detection and localization in digital video using temporal difference between adjacent frames of actual and reconstructed video clip. *Int. j. inf. tecnol.* **12,** 273–282 (2020).

3. Lin GS, Chang JF (2012) Detection of frame duplication forgery in videos based on spatial and temporal analysis. Int J Pattern Recognit Artif Intell 26(07):1–18.

4. Su, L., Li, C. A novel passive forgery detection algorithm for video region duplication. Multidim Syst Sign Process 29, 1173–1190 (2018).

5. Suzuki, S., and Be, K. (1985). Topological structural analysis of digitized binary images by border following. Computer Vision, Graphics, and Image Processing 30, 32–46. doi:10.1016/0734-189X(85)90016-7.

6. Surrey University Library for Forensic Analysis (SULFA) Data available: http://sulfa.cs.surrey.ac.uk/videos.php.

7. Wang, Zhou; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. (2004-04-01). "Image quality assessment: from error visibility to structural similarity". IEEE Transactions on Image Processing. 13 (4):600-612. Bibcode:2004ITIP...13..600W. CiteSeerX 10.1.1.2.5689. doi:10.1109/TIP.2003.819861. ISSN 1057-7149.

8. Li F., Huang T. (2014) Video Copy-Move Forgery Detection and Localization Based on Structural Similarity. In: Farag A., Yang J., Jiao F. (eds) Proceedings of the 3rd International Conference on Multimedia Technology (ICMT 2013). Lecture Notes in Electrical Engineering, vol 278. Springer, Berlin, Heidelberg.

9. S. Tan, S. Chen and B. Li, "GOP based automatic detection of object-based forgery in advanced video," 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, 2015, pp. 719-722.