

Study on Different Techniques Used In Video Forgery Detection Using Machine Learning

¹ Sreevidhya J Menon, ²Sandy Joseph

¹PG Student, ²Assistant Professor,

¹Department of Computer Science and Engineering,

¹ Rajagiri School of Engineering and Technology, Cochin, Kerala, India.

Abstract : In recent times, easy access and use of digital video editing tools become a challenge for forensic video professionals, to prove authenticity accurately in any case of suspected digital video content. Fake video detection aims to expose and test the basic facts about the video file to detect if the video content has been subjected to unscrupulous conduct. The need to test new or more efficient (or blind) Passive Fraudulent video detection methods are gaining value daily. This study has been undertaken to investigate the latest developments in the field of digital video forgery detection. This work is a detailed description of the different techniques used in machine learning methods to detect forged videos and gives researchers a broader perspective on the various aspects of forgery detection

Index Terms - Digital Video Forgery Detection, SVM, Motion Feature, Noise Feature, Machine Learning.

I. INTRODUCTION

Deliberate modification of digital video for fabrication is called digital video forgery [1]. Its definition depends on the circumstances and the context in which it is used. Especially in movies, politics, and medicine, its effects have been profound when it came to manipulating personalities, or concealing information that was devious or deceptive. Video forgery detection is a task that includes machine learning and image processing concepts to apprehend the forged video. The features can be extracted, pre-processed, and classified. The virtual content material can now be without difficulty manipulated, synthesized, and tampered without leaving any clues visibly [2]. Digital video integrity cannot be taken for granted. It has turned out to be difficult to distinguish between a tampered and a unique video. Dissatisfaction and distrust about video authenticity are growing [3]. The fast availability and usefulness of digital video modifying software have emerged as a chief mission to forensic specialists. Different papers on digital video detection are studied and in this review exceptional strategy used for feature extraction, feature pre-processing, and feature classification or patten recognition are explained.

II. RESEARCH METHODOLOGY

In most of the machine learning models the main steps involves feature extraction, feature preprocessing and classification. In the course of study a descriptive and analytical methodologies are used.

III. FEATURE EXTRACTION

3.1 Scale Invariant Feature Transform

Scale Invariant Feature Transform (SIFT) is image-based image definition comparisons and observations made by David Lowe (1999, 2004)[4]. These descriptors as well as related image descriptors are used for a large number of purposes on a computer vision related to the point of point between different views of the 3-D scene and based on the view object recognition. SIFT descriptors are not compatible with translation, rotation, and scaling transformations in the image domain and robust to moderate perspective transformations and illumination variations. For example, check the image below.

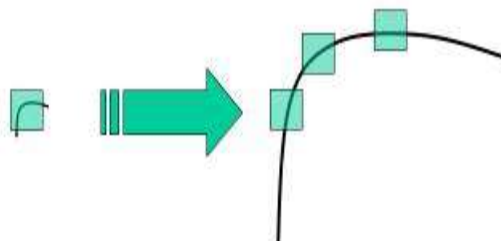


Fig.1. Scaling of an image

Corner of a small image in a small window will become flat when it is zoomed in the same window [27]. So we can say Harris corner cannot be considered as scale invariant. Through experimentation, the SIFT descriptors have proven that it becomes very useful in the process of image simulation and object recognition in the real world [4].

The steps involved in SIFT algorithm are

- Scale-space Extrema Detection.
- Keypoint Localization
- Orientation Assignment
- Keypoint Descriptor.
- Keypoint Matching

3.1.1 Scale-space Extrema Detection

Obviously, we cannot use the same window to get important points at a different level [5]. It is ok with a small corner. But to get bigger corners we need bigger windows. In this case, the scale-space filter is used. In the scale-space filter, using various σ values Laplacian of Gaussian is found for the image. LoG behaves as a blob detector that detects blobs of various sizes due to the change of σ . In short, σ serves as a scaling parameter. A Gaussian kernel with a low σ gives a higher value for a smaller corner while a high for a Gaussian kernel fits well for a larger corner. Therefore, we can find the local maxima across the scale and the space that gives us the list of values (x,y,σ) which means that there is a possible keypoint on (x,y) on the σ scale. But this LoG is a lightly expensive, so the SIFT algorithm uses the Difference of Gaussians which is an approximation of LoG [6]. The Difference of Gaussian is found as the difference of Gaussian blurring image with different textures, let alone and k . This process is performed by the different octaves of the image in the Gaussian Pyramid [27]. It is represented in figure below.

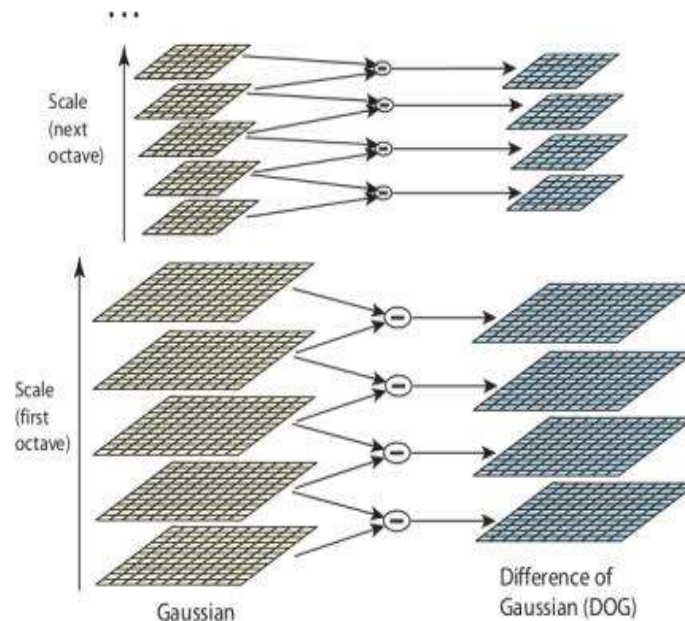


Fig.2. Difference of Gaussian for different octaves of the image.

Once this DoG has been detected, images are searched for local extrema on scales and in space. For example, one pixel in a picture is compared to its eight neighbors and 9 pixels on the next scale and 9 pixels on previous scales. If it is a local extrema, it is a key point that may exist. It means that the key point is well represented on that scale.

3.1.2 Keypoint Localization

Once potential key locations are identified, they should be adjusted to obtain accurate results. They used the Taylor series extension on a scale-space to determine the exact location of the extrema, and if the intensity at this extrema is below a threshold value, it is discarded. It is called the contrast threshold. DoG has a higher edge response, so the edges also need to be removed. A concept similar to Harris's corner Detector is used for this. A 2×2 Hessian matrix (H) can be used to calculate the principal curvature. From Harris' corner detector we know that for edges, one amount of Eigen is greater than the other. So here a simple function is used. If this ratio is greater than a threshold (edge threshold), that keypoint is discarded. It, therefore, removes any low-contrast key points and edge key points and what remain are strong interest points [6].

3.1.3 Orientation Assignment

Now the orientation is given to each keypoint to achieve the invariance to the image rotation. The area is taken around the keypoint location by scale, and gradient magnitude and direction are calculated in that region. Covering 360 degrees an orientation histogram with 36 bins has been created. It is measured by the magnitude of the gradient and a gaussian-weighted circular window equal to 1.5 times the keypoint scale. From the histogram the highest value is taken and any peak greater than 80% of it is also taken to calculate the position. Key points are created with the same scale and location, but with different directions [6].

3.1.4 Keypoint Descriptor

The keypoint identifier is now created. 16×16 neighborhoods are taken around the key point. It is then divided into 16 sub-blocks of 4×4 sizes. For each sub-block, an eight bin orientation histogram is developed [7]. In total the available bin values are 128. It is represented as a vector to create the keypoint descriptor. In addition to this, several steps are taken to achieve stability in changing light, rotation, etc.

3.1.5 Keypoint Matching

By identifying their nearest neighbors Keypoints between two images are matched. But in some cases, the second closest match can be very close to the first. It could be due to noise or some other reason. The ratio of closest-distance to second-closest distance is taken in such cases. If it is higher than 0.8, it is rejected [8]. It removes about 90% of fake matches while discarding only 5% of relevant matches.

3.2 Noise Feature Extraction

To extract noise features from a video, a frame is extracted from the video whose authenticity is to be found and converted to grayscale. DWT is performed in the extracted grayscale frame using four-level decomposition (HH, HL, LH, and LL) to obtain wavelet coefficients. After decomposition, only high-frequency components (HH) are considered using non-linear thresholding such as Hard and Soft with SURE shrinkage [25]. The threshold value is nothing but the estimation of the noise level, which is generally computed. By subtracting a frame block from denoised frame-block, noise residue of it is extracted from the frame. The same procedure is followed for all frames in the video. Scale-space Extrema Detection [9].

3.2.1 DWT Thresholding

In the process of denoising the value of threshold plays a major role. The main step is to find a threshold value which is optimum. At a time, non-linear thresholding operates on one wavelet coefficient. In this process each coefficient is smaller than the limit value set to zero, either retained or converted. Smaller coefficients contain greater noise while coefficients with higher absolute values carry more signal details [10]. For all thresholding method, the image is taken and a discrete wavelet transform is performed, which decompose image into the various sub-band. It can be represented as shown in Figure below.

LL2	HL2	HL1
LH2	HH2	
LH1		HH1

Fig.3. Two level 2D DWT of an image

In fig, the sub-bands HHi, HLi, LHi, LLi i = 1, 2... k are called the detail, where i denoted the scale and k is the largest or coarsest scale in decomposition. LLi is the low resolutions components [9].

In Hard thresholding, if the value of the detail coefficient is slightly below the threshold value λ then this value is set to zero, on the other hand, the value with a greater magnitude than λ is left unchanged. Therefore, hard thresholding is not so good for noise removal [10]. The hard thresholding is defined as follows:

$$\delta_{\lambda}^{Hard} d_{jk} = \begin{cases} d_{jk}, & |d_{jk}| > \lambda \\ 0, & |d_{jk}| \leq \lambda \end{cases} \quad (3.1)$$

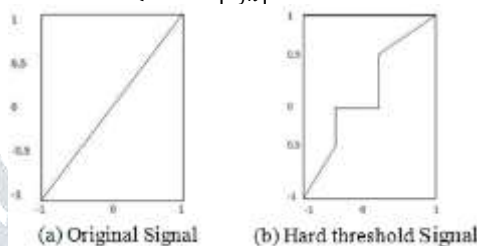


Fig.4. Original & Hard threshold signal

The Soft thresholding removes most of the transform values, which are needed for accurate real-time signal creation. Wavelet shrinkages overcome this problem and provide excellent results with Soft thresholding [10]. Soft thresholding is better than hard thresholding and is defined as follow:

$$\delta_{\lambda}^{Soft} d_{jk} = \begin{cases} 0, & |d_{jk}| > \lambda \\ d_{jk} - \lambda, & |d_{jk}| \leq \lambda \\ d_{jk} + \lambda, & |d_{jk}| < -\lambda \end{cases} \quad (3.2)$$

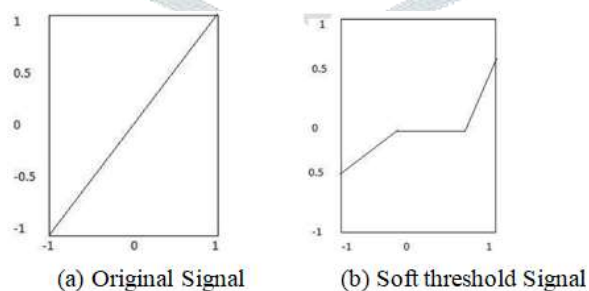


Fig.5. Original & Soft threshold signal

3.3 Motion Feature Extraction

The most frequent way for motion capture in a video is to use motion vectors. P-frame and B-frame motion vectors can be easily extracted from the video bit stream. I-frame motion vector is not readily available and extracting it from each frame is expensive computationally.

3.3.1. Intra-coded (I) frames/slices (key frames)

I Frames contain the whole picture. They are coded without reference to any frame-work other than themselves Orientation Assignment. It can be generated to create a random access point (allowing the decoder to start decoding smoothly from the beginning in that image area). It can also be done when separating image details prevents the creation of active P or B-frames. They usually require more encoding bits than other types of frame.

Typically, I-frames are used for random access and are used as code references for other images. Internal half-time refreshments are common in applications such as digital television broadcasting and DVD storage. Longer renewal times can be used elsewhere. For example, in video conferencing systems, it is common to send I-frames very infrequently.

3.3.2. Predicted (P) frames/slices

It requires prior decoding of another image (s) to decode a picture. It can contain image data as well as motion vector displacements and combinations of both. It can point to previous images in decoding order. Old-fashioned designs use only one previously decoded image as a reference at the time of decoding, and require that image to precede the P picture sequence of display. In H.264, it can use multiple pre-decoded images as indicators during decoding, and may have a any arbitrary display-order relationship associated with the image (s) used for its prediction. They usually require fewer encoding bits than the I pictures do

3.3.3. Bi-directional predicted (B) frames/slices (macro blocks)

It requires the prior decoding of subsequent frame(s) to be displayed. It may contain image data and / or motion vector displacements. Old standards only allow for a global motion compensation vector for the entire frame or a single motion compensation vector for each macro block. Include other predictive methods that create a prediction of a motion region (e.g., macro block or small area) by averaging the predictions obtained using two previously decoded reference regions. Some standards allow two compensation of motion for each macro block (biprediction). At older standards (such as MPEG-2), B-frames are not used as prediction references for other images. As a result, low-level encoding (requiring less space) can be used for such B-frames as detail loss will not affect the quality of prediction of the following images.

H.264 lowers this limit, and allows B-frames to be used as decoding references for other frames at the discretion of the encoder. Old standards use two pre-decoded images as references during decoding, and require one of those images to precede the B-frame in sequence and the other to follow it. H.264 allows one, two, or more pre-decoded images as references during decoding, and may have an explicit order relationship in respect of the image (s) used for its prediction. The retrieval of information means that B-frames typically require fewer encoding bits than I or P frames.

In the computer vision field, without a doubt, the discovery of moving objects on the scene is an important topic. It is one of the basic steps in many of the systems designed for vision. For example, applications such as human-computer interface (HCI), robot visions, and intelligent monitoring systems require the detection of moving objects [23]. Various methods have been suggested and proven to be effective in detecting moving objects using stationary cameras [19-22]. There are a number of suggested ways to moving object detection with a non-stationary camera, but their effectiveness is still questionable. The most critical reason for hindering the effectiveness of these methods is the amount of calculation required to work.

In the process of identification of the motion, above all else, the base image (the first edge) should be converted to greyscale and after that, the current image is converted to a dark scale image because in greyscale the scope of the shading appears 0-256 as it is. Reducing the point quantity including testing each model expects us to be sharper about picking them in any case. It doesn't make sense to just get the motion of a point across the image, and keep all the following highlights into the model in a single assessment. Above all else, the number of highlights will be expansive which will be resulting in poor performance of execution. Second, we will be providing a huge number of external feature points, such as those in the background. To avoid wasting time in the background, we can use image segmentation to separate the foreground portions of the scene from the background by subtracting the image that has already been converted into the original image. Movement detection methods can be divided into two main categories, namely pixel-based and region-based algorithms [24].

Continuous edges on a pixel by pixel layout in a motion picture are calculated and the limit is connected which sets them as position or movement. In any case, as things move, their homogeneous inside does not change the picture forces in the short term, so the movement should be separated at the limits. Similarly, it does not indicate a pixel relationship with its neighbor.

One method for finding the motion residual for feature extraction is to find the basic frame by calculating the average of pixels across video frames. Extract the residual of the motion by subtracting the pixel of each frame to the base pixel frame to get the residue of the motion of each frame. Motion residue is the residual after an object has been taken or copied.

IV. FEATURE PREPROCESSING

4.1. Correlation Matrix

Correlation is an indication of the changes between the two variables. The correlation matrix is a symmetrical matrix with all diagonal objects equal to + 1. The correlation matrix provides data about correlation only not the cause [13]. Correlation tells us how much one variable changes with a little variation in another variable. Depending on the direction of change, correlation can take positive, negative, and zero values. The high degree of correlation between a variable that is dependent and a variable that is independent indicates that independent variables are very important in output determination.

In a multiple regression setup where there are multiple factors, it is important to find a correlation between the dependents and all the independent variables to create a model that works with high accuracy. One must always keep in mind that a large number of factors do not necessarily mean better. Many features can lead to a decrease in accuracy if they contain any unnecessary features that create unnecessary noise in our model. Various metrics such as Pearson r correlation, Kendall rank correlation, Spearman rank correlation, etc. can be used to find the correlation between the two variables [14]. For example of using correlation matrix, frame features can be separated by blocks of 8×8 pixels each. After that for two consecutive frames, i.e., frame at t and frame at t + 1, correlation is found in blocks in the same place as $R(x, y)$. $R(x, y)$ is the block level Cross correlation lying between - 1 and 1. The correlation values indicate the difference or similarity between the two blocks. Correlation values are found in each block pair and are stored in a matrix in their respective locations called correlation matrix. These matrixes are designed for all consecutive independent frame pairs in the video.

The steps of block-level correlation are as follows: Step 1. Take two consecutive residue frames and divide each of them into the size of 8×8 pixels blocks. Step 2. Find the correlation between the two blocks in the same location and keep the value in the matrix in the corresponding location. Repeat the two for all the blocks and save the matrix in a data structure two consecutive frames. Repeat these steps for all consecutive video frames. When object removal forgery arises, the correlation values of temporal features change in the forged region. The change may be an increase or decrease in the amount of value of correlation and depends on the fraudulent scheme used [10].

4.2 Probability Transition Matrix

Markov's first-order process is a stochastic process in which the future state depends only on the present state. Markov's first-order process is often called the Markov process. When in a discrete space, it is called the Markov chain.

$$P(X_{n+1}=x|X_1=x_1, X_2=x_2, \dots, X_n=x_n) = P(X_{n+1}=x|X_n=x_n) \tag{4.1}$$

The speculation about Markov's process may not be true at all in reality. But even if that is not the case, we can model more states in the system to bring us closer to Markov's process at some point. Markov's chains are often defined by a set of states and the probabilities of transition between each state. Markov's process can be an appropriate measure in solving complex Machine learning problems and reinforcement learning problems. An example of Markov's process is shown below:

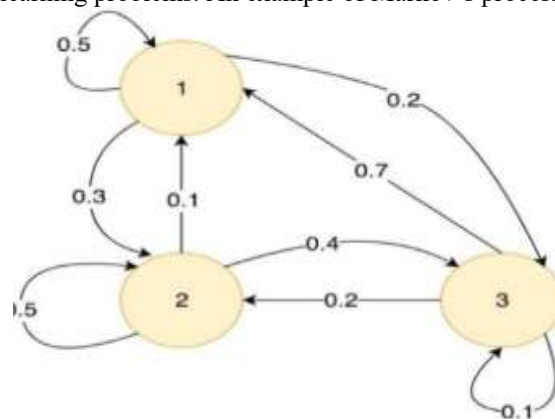


Fig.6. Markov's Process

In addition, the probability for the transition from one state to another can be represented in the form of a Matrix called the Transition Matrix. This transition matrix is also called the Markov matrix.

	S ₁	S ₂	S ₃
S ₁	0.5	0.1	0.7
S ₂	0.3	0.5	0.2
S ₃	0.2	0.4	0.1

Fig.7. Transition Matrix

The element i,j is the transition probability from state j to i . Transposed notation are used in some cases, so that element ij represents the transition from state i to state j . Markov's matrix columns add up to one, which means the probability of reaching a state from any other possible state is one. Once the matrix is defined, we can use linear algebra and eigenvector to determine its stable state [26].

V. CLASSIFICATION

5.1. Brute Force Matching

Brute-force matching is the most basic in the matching of substring. It has two inputs: pattern (string of characters you can want to search) and text (string of characters you search in). The algorithm begins by aligning the pattern at the beginning of the text. After that each character of the pattern is compared with the corresponding character of the text. Left-to-right matching match, until all characters are found to be the same or different. While the pattern is not found and the text is not finished, the pattern is redirected to the right place and compared to the corresponding letter of the text [16]. Brute-force matching is one of the easiest ways to match a feature in OpenCV-Python. Before using brute force matching, label the image segments uniquely. After that it is matched in brute-force. In the first segment, the descriptor of one feature is taken and compares it with the other entire feature in the remaining segments using a specific distance calculation method and the closest is restored [17].

When any features are matched, copy-move forgery occurs in the image. Then compare the features of the objects detected using a powerful brute force comparison method. This method takes a descriptor from the first frame and is matched to all other descriptors in the next frame uses a certain distance calculation and the closest one is restored. In brute-force matching method, two thresholds are used namely: feature point matching threshold (TRp) and block matching threshold (TRb). When the TRp becomes large, get good match for the feature point. A good match is found when the feature point threshold becomes high. In the proposed route a feature point that corresponds to a set limit 0.95. Finally, find matched frames according to the TRF framework [18].

We can use Brute Force Matcher to determine similarities between the same key points in both images. On the basis of Euclidean Distance between two key points matches are evaluated. The Euclidean distance of all the keypoint in the first image is calculated by all the other keypoints in the next image. Good match and then separated by certain conditions of minimum distance criteria [17].

5.2 Block Matching Algorithm

Block matching algorithm (BMA) in the video copy-move forgery can be used to find regions that represent forged or non-forged moving objects. Various search methods have been compared using standard forensic data used to find the most relevant search method for video copy-move forgery detection scheme. Adaptive rood pattern search (ARPS) has been selected as the most suitable BMA to find regions of moving objects in video forensic because of its functionality. It is shown in the figure below.

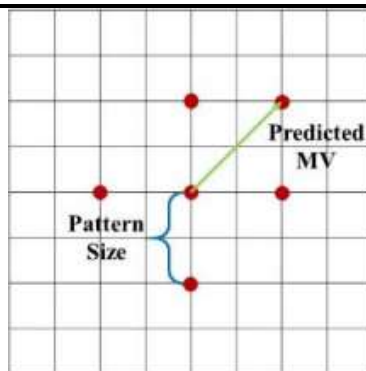


Fig.8. Adaptive rood pattern search

There are three key elements in the BMA which are block determination, search method, and the matching criteria. Determining a block is the first element in the BMA that defines the first search point in the reference frame, position, and block size in the current frame. Next, the second part is a search method that specifies where to look for candidate blocks in the reference frame. Finally, the third component is the matching criteria that determine the best match for candidate blocks [12].

The main idea of BMA is to divide the current frame into pixel sizes for $N \times N$. These MBs in the current frames are compared to a corresponding block with their adjacent neighbors in the reference frame to create a vector that determines the movement of the MB from one location to another in the reference frame. The displacement between the current MB and reference MB is found from this vector called motion vector (MV). The first BMA is the full search (FS) or also known as the exhaustive search (ES). This search also has a very high computational complexity due to the calculation of the cost function of each location that can occur in the search window. However, ES finds best possible match that has led to a very high peak-signal-to-noise- (PSNR) high within the BMA even though more computations are needed with the increasing size of the search window. Therefore, the researchers started the study on fast block matching algorithm (FBMA) in the early 80s to solve complex computational problems while maintaining the value of the PSNR by reducing the number of points searched.

The most suitable BMA is determined by its performance to the various sizes of the block and the search area. In addition, the severity of these BMAs under compression is also being investigated. The size of the MB did not affect the calculation of all FBMA because each BMA has its own specific steps. Currently, for ES, the large block size reduces the calculation. Search (SA) for the best MB match requires a search parameter on all four sides of the corresponding MB in the previous frame. BMA becomes more expensive in computation when there is greater movement because it requires a larger p . The matching MB is determined by the lowest cost function the Mean absolute difference (MAD) and mean square error (MSE) given by the equations given below [15].

$$\text{MAD} = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (C_{ij} - R_{ij}) \quad (5.1)$$

$$\text{MSE} = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |C_{ij} - R_{ij}| \quad (5.2)$$

5.3 Support Vector Machine

In machine learning, support vector machines are supervised learning models with associated learning algorithms that analyze the analysis of the data used to classify and regression. To identify patterns they analyze a large number of data. We classify each data object as a point in a n -dimensional space (where n is number of features you have) by the value of each feature being the value of a particular coordinate. After that, we do the splitting by finding a hyper plane that separates the two classes very well. SVM produces parallel partitions by creating two parallel lines. In a single pass, it separates the space to generate flat and linear partitions. Divide into 2 categories with a clear gap that should be as wide as possible. This partition is done with a plane called a hyperplane [11]. This is shown in the figure below:

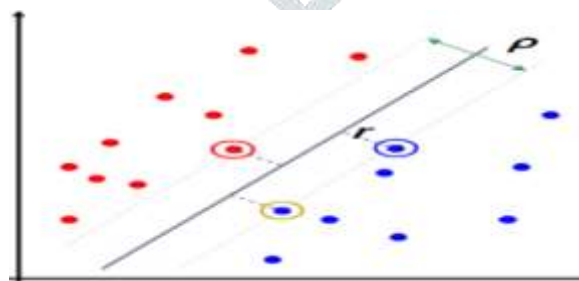


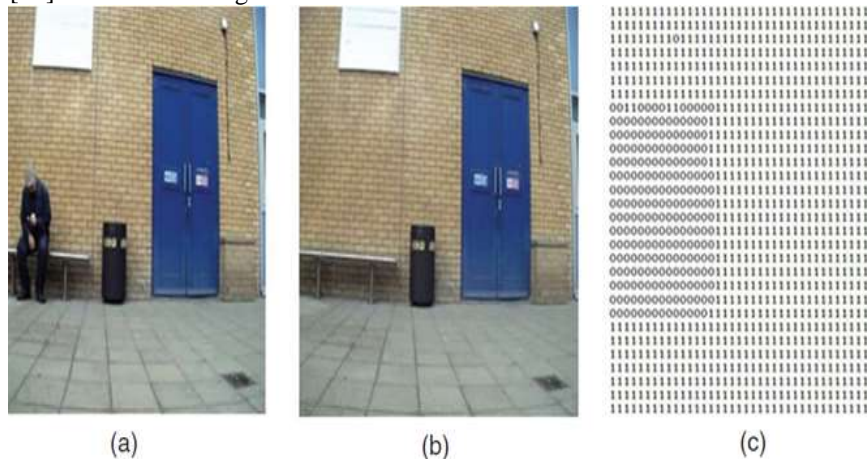
Fig.8. Hyper line selection from data set to classify

The margin between the 2 categories represents the longest distance between the data points closest to those classes. Given a set of training examples, each marked as one of two categories, the SVM training algorithm creates a model that provides new examples in one category or alternatively, which makes it a non-probabilistic binary linear classifier [11]. The SVM model represents examples as points in a space, mapped so that the examples of the different categories are separated by a clear gap as wide as possible. Then based on which side of the gap they fall on, new examples are mapped into that same space and predict category they belong. The main advantage of the SVM network used as a classifier is its excellent ability for generalization and a very powerful learning process, which leads to the global minimum of the defined error function.

5.4 Gaussian Mixture Density Based Bayesian Classifier

Unlike most Learning Machine (ML) algorithms, Expectation Maximization (EM) is an unsupervised learning algorithm, and for input it does not take responses (class labels or performance values). Instead, it incorporates the Maximum Likelihood Estimate of the Gaussian Mixture parameters from the input sample and maintains all parameters within the structure. A trained model can be used to predict, as with any other variations such as the Bayesian Classifier. Using means and variations of the correction matrix, the GMD-based Bayesian classifier determines whether the video frame is fake or authentic.

If we test the Gaussian Mixture Density parameter, which is made up of two classes namely '0' for fake and '1' for real. The Expectation-Maximization algorithm helps determine the parameters of the Gaussian Mixture Density. The goal is to maximize potential function with respect to parameters that include means and covariance of elements and mixing coefficients of the Gaussian Mixture Density model. Consider the feature set of N vectors $x_1, x_2, x_3, \dots, x_N$ from a d -dimensional Euclidean space drawn from a Gaussian mixture. Measure the initial value of the log likelihood by initializing the means μ_k , covariance Σ_k and mixing coefficient μ_k [10]. It is shown in figure below.



(a) Original frame (b) Tampered frame (c) Bayesian classifier output for tampered frame

Fig.9. GMD based Bayesian classifier, 0 for forged and 1 for authentic

VI. CONCLUSION

Security is the significant worry for any framework to keep up the trustworthiness, classification and picture legitimacy. Widespread use or video sharing on social media platforms such as WhatsApp, Facebook, Youtube, and news channels has a huge impact on our daily lives. "Seeing is no longer believing". Now a day the integrity and authenticity of the video shown cannot be unquestionably accepted. In order to prove the integrity of the video, various forgery detection methods have been proposed by several scientists. Three main steps in machine learning based video forgery detection are Feature extraction, feature preprocessing and classification or pattern recognition. This work is concatenation of various techniques used for these steps. With this study the idea behind advancement in various technologies in the related field is identified some of the best methods are taken.

VII. FUTURE SCOPE

In the future, this effort can be enhanced by considering more techniques used in the process of video forgery detection. Also a working system can be implemented using the techniques mentioned in this work.

References

- [1] Rohini Sawant and Manoj Sabnis. 2018. A review of video forgery and its detection. IOSR Journal of Computer Engineering 20.2, 01-04
- [2] S Dhivya and B Sudhakar. 2020 A scrutiny of copy move video forgery detection in different methods. International Journal of All Research Education and Scientific Methods. 7.429, 2455-6211
- [3] KN Sowmya and HR Chennamma. A survey on video forgery detection. International Journal of Computer Engineering and Applications 2015, 2321-3469
- [4] Tony Lindeberg. 2012 Scale Invariant Feature Transform. Scholarpedia. DOI: 10.4249/scholarpedia.10491
- [5] Opencv python tutorials, Available at: <https://opencv-python-tutorials.readthedocs.io/>, accessed April 2021
- [6] Researchgate, Available at: <https://www.researchgate.net>, accessed April 2021.
- [7] S Sheena and Sheena Mathew. 2016. A comparison of sift and surf algorithm for the recognition of an efficient iris biometric system. International Journal of Advanced Research in Computer and Communication Engineering, DOI 10.17148/IJARCCCE
- [8] GK Rajini, Raghav Katyal, S Naseera, N Amutha Prabha and S Balaji. 2018. Object identification through image processing algorithms, International Journal of Pure and Applied Mathematics, 3995-4007.
- [9] Pallavi L Patil and VB Raskar. 2015, Image denoising with wavelet thresholding method for different level of decomposition. International Journal of Engineering Research and General Science, 2091-2730
- [10] Ramesh C Pandey, Sanjay K Singh and KK Shukla, 2017. A passive forensic method for video: Exposing dynamic object removal and frame duplication in the digital video using sensor noise features. Journal of Intelligent & Fuzzy Systems, DOI: 10.3233/JIFS-169275
- [11] PJ Mercy and S Amsa Vani. 2021, Sign language identification using image processing, International Journal of Creative Research Thoughts (IJCRT), 2320-2882
- [12] Rahul Bhandari and Ashutosh Vyas. 2014 Analysis the performance of three step search algorithm for motion estimation, International journal of engineering research & technology (ijert) etrasct, 2278-0181
- [13] <https://www.vertica.com/blog/in-database-machine-learning-2-calculate-a-correlation-matrix-a-data-exploration-post/>
- [14] <https://towardsdatascience.com/understanding-feature-extraction-using-correlation-matrix-and-scatter-plots-6c19e968a60c>

- [15] Saleha Safie, Azurah A Samah, Ghazali Sulong, Hairudin Abd Majid, Rafidah Muhammad and Haswadi Hasan. 2017, Block matching algorithm for moving object detection in video forensic. In: Proceedings of the 6th ICT International Student Project Conference (ICT-ISPC), Skudai, 978-1-5090-62553 DOI: 10.1109/ICT-ISPC.2017.8075330
- [16] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. 2001 Introduction to Algorithms. Vol II, MIT press , Cambridge, Massachusetts, London, England, p. 906-923.
- [17] Jakubovic, Amila & Velagic, Jasmin. 2018, Image Feature Matching and Object Detection Using Brute-Force Matchers. In: Proceedings of the International Symposium ELMAR, Zadar, Croatia, p.83-86. DOI:10.23919/ELMAR.2018.8534641.
- [18] Neema Antony and Binet Rose Devassy. Implementation of image/video copy- move forgery detection using brute-force matching. 2018 In : Proceedings of the 2nd International Conference on Trends in Electronics and Informatics, Tirunelveli, India, 978-1-5386-3570-4 DOI: 10.1109/ICOEI.2018.8553953
- [19] C Stauffer and W Grimson. 1999 Adaptive background mixture models for real-time tracking. In: Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), Fort Collins, CO, USA, 1063-6919
- [20] A Elgammal, R Duraiswami, D Harwood, LS Davis, R Duraiswami, and D Harwood. 2001, Background and foreground modeling using nonparametric kernel density for visual surveillance. In: Proceedings of the IEEE, p: 1151– 1163.
- [21] T Ko, S Soatto, and D Estrin. Warping background subtraction. 2010, In: Proceedings of the 23rd IEEE Conference on Computer Vision and Pattern Recognition, CVPR, San Francisco, CA, USA, p: 1331–1338.
- [22] Y Sheikh and M Shah. 2005, Bayesian modeling of dynamic scenes for object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 11 p:1778–1792, DOI: 10.1109/TPAMI.2005.213
- [23] I Kim, H Choi, K Yi, J Choi, and S Kong. 2010 Intelligent visual surveillance — A survey International Journal of Control, Automation and Systems, 8, 926–939. <https://doi.org/10.1007/s12555-010-0501-4>
- [24] R Kannadasan, N Prabhakaran, S Rajkumar and Rishin Haldar. 2018, Web Camera Based Motion Detector Application. In : Proceedings of the 2nd International Conference on Trends in Electronics and Informatics, Tirunelveli, India, 978-1-5386-3570-4, DOI: 10.1109/ICOEI.2018.8553798
- [25] Mohideen, S Kother, S Arumuga Perumal, and M Mohamed Sathik. 2008, Image de-noising using discrete wavelet transforms. International Journal of Computer Science and Network Security , 8.1, 213-216.
- [26] <https://jonathan-hui.medium.com/machine-learning-hidden-markov-model-hmm-31660d217a61>
- [27] https://docs.opencv.org/4.5.0/da/df5/tutorial_py_sift_intro.html

