

# Using Deep Learning Algorithms, Video Indexing Through The Human Faces Represented As Ean-8 Linear Bar Code

Sanjoy Ghatak<sup>1</sup>, Debotosh Bhattacharjee<sup>2</sup>

<sup>1</sup>Department of Computer Science and Engineering, Sikkim Manipal University, Majhitar, 737136, India, [sanjoy1cs@yahoo.co.in](mailto:sanjoy1cs@yahoo.co.in)

<sup>2</sup>Department of Computer Science and Engineering, Jadavpur University, Kolkata-700052, India  
[debotoshb@hotmail.com](mailto:debotoshb@hotmail.com)

---

**Abstract.** This research paper proposes a technique that uses a facial image as a key frame to create an EAN-8 linear bar code from video. Recognition of faces has gained popularity because it can be used in various industries, including information security, smart cards, video surveillance, and law enforcement. Additionally, it aids in recognizing patterns and imagery processing. Numerous approaches for indexing videos use human faces as cues, but none are effective in identifying faces in videos due to factors like direction changes in the face, image brightness variations, illuminations of the face, etc. This research uses a novel method called "Video indexing through the human face as an EAN-8 Linear bar code using machine learning and deep learning algorithm" to overcome these problems. Viola-Jones, DSFD, Multi-Task Cascaded Convolutional Neural Networks (MTCNN), Blaze Face, and YOLO v3 extract key frames, or the human face, from video frames to address this problem. Following key frame identification, this key frame is converted to an EAN-8 linear bar code for video indexing and recognition. The main application case for this research is the identification of human face frames from videos and their representation as an EAN-8 linear video indexing bar code. This method is helpful for various tasks, including security, human activity recognition, video surveillance, and communication channel description. After describing each one, this approach indexes the video as a Linear EAN-8 bar code based on each human face in the movie. This research first compares the performance of several widespread face recognition and detection models (like Viola Jones (Haar Cascade), MTCNN, DSFD, Blaze face, and YOLO v3) that use machine learning and deep learning concepts. Following that, it is advised to use the top-performing model (MTCNN) to extract specific faces from the provided video footage. These extracted faces are used to generate EAN-8 linear barcodes, giving us a simple and fast way to produce a particular facial individuality predicated on natural features and thus reducing bandwidth, storage, and time complexity. The Hollywood video dataset, FDDB face dataset, WIDER FACE dataset, and LFW dataset were employed to assess the suggested method, which was proven to efficiently index videos using human faces.

**Keywords:** MTCNN, Viola Jones, DSFD, Blaze face, YOLO v3, EAN-8 linear bar Code, Key frame, Human face, Window Technique.

---

## INTRODUCTION

Methods for content-based automatic access to visual data are in more demand as the number of photos and videos increases daily. As a result, numerous strategies for video indexing and retrieval have been developed. Most image indexing techniques rely on texture and color, which are low-level features. Persons who utilize low-quality image and video indexing technologies are hard to identify, and you cannot index persons according to them. A person is one of the main components in the scenario shown in the photo and video. The document [1] covers a procedure for merging picture and video sequence recognition with person detection. It takes more time and space to index the video using the method described in the paper [1] based on a person's face. Nowadays, most Internet users enjoy and enjoy themselves through videos. It serves as a source of inspiration for individuals, companies, and commerce through communication channels. Accordingly, the communication channel's bandwidth is crucial for transmitting data from one device to another. Transferring a human face as data over a communication channel will need more time, space, and bandwidth. The complete problem could not be covered in Papers [1] and [2]. In the past, most face recognition research concentrated on identifying faces in still photos. However, recognition from a single image is challenging because of common problems such as changing illumination, variable positioning, occlusion, angular face, directional face change, and facial expression. These causes frequently produce more differences in face image than identity changes. Faces videos may now be recorded, stored, and analyzed thanks to the development of low-cost video cameras and increased computational power.

Multiple frame video inputs result in duplicated and expensive data. By precisely capturing extra information, video-based recognition is thought to eliminate the inherent uncertainties in image-based recognition, like low-resolution sensibility, pose changes, and occlusion, resulting in greater accuracy and dependable visage recognition. Inputs for videos also make it possible to record facial dynamics that are helpful for facial recognition. The author of the paper [3] proposed an EAN-8 linear barcode approach for video indexing utilizing a human face as a cue. In this paper, the author employed the Viola-Jones algorithm to detect faces from input videos. However, the issue with this method is that it cannot accurately identify faces in photos or videos if the faces are angular or have changed direction. The same author proposed a method for video indexing using photos of human faces using the LGFA and window methodology in the paper [4]. To enhance the quality of the EAN-8 linear bar code of the illumination invariant face images, the author of this research employed LGFA in conjunction with the Viola-Jones method for face detection from the input video. The window approach is employed to produce a stable barcode. However, the research does not address the topic of face detection from angle faces or whether faces are rotated in the input video. Therefore, this work investigates an indexing method that uses an EAN-8 linear bar code representation of a human face to overcome the same challenges as previously stated. An idea to create facial image QR bar codes from a video's key frames is presented in Paper [53]. In this paper, face images are represented by a QR bar code. This technique extracts key frames from videos where faces are present at key frames using Multi-Task Cascaded Convolutional Neural Networks (MTCNN).

Following key frame detection, a QR bar code is created from the key frame for the purposes of video recognition and indexing. AdaBoost and Haar-Like features are used by Viola and Jones [8] to develop classifiers with cascades that are efficient in the present for their cascade face detector. Numerous publications [9, 10, 11] indicate that this detector may experience significant degradation in practical uses where there is bigger visual variability of faces in humans, even with classifiers and features that are more advanced. In addition to the cascade structure, [12, 13, 14] present models of deformable parts (DPM) to achieve exceptional performance for face detection. Nevertheless, they often require expensive annotation during the training phase and have a high computational cost. Convolution neural networks (CNNs) have made significant strides recently in several computer vision applications, including face recognition and image classification [15–16]. Certain CNN-based visage identification techniques have been introduced lately due to CNNs' excellent results in computer vision tasks. Deep learning neural networks, developed by Yang et al. [17], require a high response in face regions to recognize facial attributes, which produces potential facial windows. However, this method still takes much practice because of the intricate CNN structure. Li et al. [18] use CNNs in a cascade for face detection; however, this approach needs face detection to calibrate the bounding box at an additional computing cost, disregarding the obvious connection between bounding box regression and facial landmark localization. Additionally, face alignment is crucial. Two popular types are regression-based methods [19–21] and template-fitting methods [22, 23, 14]. Recently, Zhang et al. [24] suggested using facial attribute identification as a stand-in task to boost deep convolutional neural networks' face alignment performance. Nevertheless, most face alignment and identification methods disregard the innate connection between these two assignments. These functions have limitations even though there are a number of them that try to solve them jointly. For example, Chen et al. [25] use pixel value difference characteristics to run random forest alignment and detection simultaneously. However, the handcrafted elements used restrict the execution. The accuracy of multi-view face detection is enhanced by Zhang et al. [26] using multi-task CNN; however, the accuracy is constrained by the initial detection windows created by a subpar face detector. Conversely, increasing the detector's effectiveness during training requires mining hard samples. However, conventional hard sample mining usually works offline, resulting in a significant increase in manual procedures. It would be ideal to create an online hard sample mining technique for face alignment and detection that automatically adjusts to the current training procedure. Paper [27] suggests integrating these two assignments using unified cascaded CNNs and multi-task learning. There are three stages to the recommended CNNs. First, it uses a shallow CNN to create candidate windows quickly. It then uses a more sophisticated CNN to refine the windows so that many non-face windows are rejected. Lastly, it uses a stronger CNN to hone the findings and output the locations of the facial landmarks. Three new contributions—enhanced feature acquisition, design for progressive loss, and anchor assigned using augmented data—are addressed in the manuscript [5] that suggest a unique face detection network, the DSFD (Dual Shot

Face Detector). Paper

[6] proposes a new face detection framework called Blaze-face, adapted from the Single Shot Multi-Box Detector (SSD) framework and optimized for inference on mobile GPUs. Blaze Face has continuously outperformed other face detection algorithms while preserving real-time performance across various benchmarks. Because of this, it has been widely used in various industries, including social media, augmented reality, and video conferencing. However, this neural network is mainly used in lightweight devices. Redmon Joseph Farhadi Ali suggested the YOLO v3, a lightweight face detector model discussed in a paper [7]. Compared to earlier YOLO detection networks, YOLO v3 is an improvement. It has improved feature extractor network strength, multi-scale detection, and some loss function modifications over previous iterations. This network can now detect a much greater number of large and small targets. YOLO v3 operates quickly and enables real-time inference on GPU devices, much like other single-shot detectors. However, this neural network is mostly utilized by lightweight gadgets. Paper [3] covers "Video Indexing through the human face. "Within paper [3,4], the Viola-Jones algorithm—which uses AdaBoost and Haar-Like features— detects faces from video and face datasets. Although the Viola-Jones algorithm uses an outdated framework, it is a fairly strong, quick, and reliable face detection (not recognition). This algorithm's disadvantage is that it can only detect fully frontal upright faces effectively. The Viola-Jones algorithm and paper [3,4] do not address major challenges like posture change, lighting invariant features, and angular changes of the face. On the other hand, face detection takes longer in real-time lightweight devices that use the Viola-Jones algorithm. Major issues like posture change, lighting invariant features, and facial angular changes are not adequately addressed in Paper [53]. MTCNN is unable to accurately identify small faces from input video after using lightweight devices. Despite the fact that QR codes are more compact representations of human faces, they require more storage space than EAN-8 linear bar codes. This paper, "Video indexing through the human face as an EAN-8 linear bar code using Machine learning and Deep learning algorithm," mentions a novel solution to these issues. This study addresses several major challenges, including time and spatial complexity, lack of storage space, angular changes of the face, lighting invariant features, changing posture, and the inability to save important video frames. Using machine learning and deep learning algorithms, the research's primary contribution is video indexing through human faces as an EAN-8 linear bar code. This technique is offered to address these problems. Using machine learning and deep learning algorithms (Viola Jones, DSFD, MTCNN, Blaze Face, and YOLO v3), cropped face portraits were produced to extract the key frames from the frame and address the key frame storage issues.

Following this, a comparison is conducted using the key frame extraction and face detection algorithms of Viola Jones, DSFD, MTCNN, Blaze Face, and YOLO v3.

A unique linear EAN-8 barcode, discussed in the paper [3,4,28], will be issued to each individual, which they can use to access a database. Additionally, a linear EAN-8 barcode will be generated. Since not all linear EAN-8 bar codes are compatible with human faces, the proper kind of bar code must be used. Selecting the appropriate linear EAN-8 barcode size is also necessary to avoid potential problems. Small face detection from input video can also benefit from this technique.

As a result, issues with shifting posture, storage capacity constraints, storing video key frames, and time and space complexity are all addressed by the recommended method of indexing videos.

The remaining portions of the manuscript are arranged as follows: The associated works of the earlier methodology are described in Section 2; the functions of the various algorithms and the suggested method are covered in Section 3; the experimental results and discussion are covered in Section 4; and the research is concluded in Section 5.

## LITERATURE SURVEY

As digital technology, web streaming, and social networking have advanced, more users can modify video objects and want to use them for broader applications. Experts are especially interested in camera faces because they are common in everyday life. As a result, the human face is currently regarded as a significant object for video indexing. According to the paper [29], three data sources or modalities are considered when creating a video report for video indexing. These three modalities of communication are spoken, written, and visual. This work focused on using semantic analysis for video search. This work discusses the three elements of a video—visual, audio, and textual— and their specifics. The hybrid Hidden Markov model for facial recognition and its support for vector machine-based clustering of

video system indexing were discussed in Paper [30]. The five parts of the human face—the brow, eye, nose, mouth, and chin—are classified in this study using the SVM, and any independent traits of each component are searched for. In paper [31], the video's narrative structure is subsequently divided into more manageable, shorter segments by AI algorithms, making it easier for users to scan the content and locate a particular segment of the video. A deep learning architecture was created to use discourse interactions and visually elaborated labels to separate the film into storylines and provide commentary on pertinent outlines. Images, sound, discourse, and literary substance were all incorporated into this architecture.

Paper [32] suggested a procedure for arranging and obtaining video using a condensed illustration of the Bag-of-Faces. This method encodes face tracking as a minimalist depiction of a single bag of faces, enabling efficient processing of large amounts of facial data. Mingtao Pei [33] focused on deep learning the binary hash representation while creating a face video retrieval system. In this work, the researcher developed a deep convolution neural network (deep CNN) for face-to-face video retrieval to learn from compact and discriminative binary representations. The subjects of papers [34] and [35] were significant intra class facial variations and the urgent need to preserve time and space; these problems were tackled and resolved in this paper. A paper [36] discusses a solution to the issue with subpar apps related to the indexing of videos. The author of this study described an indexing system for videos that included facial recognition solutions and face detection. The researcher employed a neural network-based face detection and recognition system, a K-Means clustering method, and a pseudo-two-dimensional HMM for face detection. This technique does not allow for the tracking of the faces. In lip-based audio-based video indexing, speech is one of the most essential indexing factors. An obstacle to video indexing using audio modality is speech variance. The paper [37] described the method for controlling this variability as selecting frames for key frames from movies by analyzing the timing of the lip motion patterns. A document discussed how to automatically recognize a human face from an unbranded video series [38]. The author employed an iterative algorithm to provide a confidence level for the presence or absence of faces in the video images. To handle independent video data seen through a person's face, sort it, and get it back when needed. Paper [3] offers an innovative approach. The main objectives of this study are differentiating proof using standardized identification, extracting information from the video's key frames, using facial recognition with essential frames, and ordering video using a barcode.

Video frame preprocessing before accessing hidden compilations of videos has various disadvantages, according to the analysis by Gayathri et al. [39]. Techniques for feature extraction and classification are considered to prevent pre-processing errors. Here, it is expected that the dominant frame structure of the incoming video frame and video indexing will be used with multiple extraction capabilities. Using a fuzzy-based SVM classifier, separate the incorporated frame structures into dominating structures. A video clip's texture information is extracted using color attribute extraction and the multidimensional directional gradients (HOGs) histogram. Storage space is limited, and the classifiers in this method cannot focus on signal description applications using video processing. Deep neural networks, particularly facial recognition, have been the focus of extensive research and study. Deep learning models frequently identify artifacts (Lin et al., 40).

Consequently, this research suggested a cloud-based deep-learning video recovery system. Following matching the residual pictures, the dataset is extracted and preprocessed to create a format appropriate for CNN templates. The final dataset is created and fed into CNN's Face Net, Arc Face, and VGG Face Recognition models for pertaining. In this sense, neither the system's efficiency nor its ability to obtain new datasets is improved.

Li et al. [41] initially suggested a traffic location quantization index that relies on backbone traffic features to tackle problems related to link management programming for intelligent town protection video retrieval. This made it possible to assess the traffic region characteristics in the back-end communication quantitatively. Deep learning-based fundamental frame abstraction and retrieval are suggested to improve the efficiency and precision of video recovery. Key frame features are extracted using the existing convolutional neural network architecture, an adaptive key frame selection algorithm is created, and supervised, semi-supervised, and unsupervised retraining models are built. The approach doesn't preserve the intricacy of space and time or store key frames.

Bastard et al. and associates [42], to forecast the main impacts of face images with varying looks, a method known as E-appearance is presented. These techniques use the face anthropometric hypothesis

to explain facial deformation and the wrinkle-in painting technique to eliminate wrinkles. Technology may not precisely capture people's features due to wide variations in lighting, facial expressions, and other factors.

Bastard et al. and associates [43], Two different approaches are used in this study's modeling of facial rejuvenation. These techniques use the face anthropometric hypothesis to explain facial deformation and the wrinkle-in painting technique to eliminate wrinkles. As such, technology may not accurately capture people's features due to wide variations in lighting, facial expressions, and other factors.

In a paper, Goutam Dutta used deep learning techniques to identify attributes and generate a sentence for a picture [44]. In addition, a term for video frames will be generated utilizing the same model as the caption for the image. Important frames can be extracted by running a video through the program's built-in framework for key frame extraction while the video is being created. The same image captioning model used to make the captions for the pictures is also used to caption the key frames captured from the video. The movie frames create captions using a pre-trained and pre-programmed model.

Nonetheless, one of the difficulties was figuring out what to say when confronted with a vast vocabulary, and another was creating a logical frame sequence out of video images. By combining video storytelling and indexing techniques, Jacob et al. present a novel way to analyze video content and find the needed video clip from an extensive video in a paper [45]. Video material is analyzed using the video storytelling method to create a video explanation. After that, an index is made from the video description using the wormhole technique, guaranteeing that a keyword with a set length  $L$  can be located as soon as feasible. Video search engines may utilize this video index to find the necessary portion of the movie because the term frequently appears in the keyword search for the video index. The user can download and transfer only the relevant parts of the video, saving them from having to download and upload the entire thing. This process could, therefore, take a long time.

Krishnaraj et al. [24] discovered that despite the effectiveness of picture indexing provided by cloud services, the semantic gap between the user query and the diverse semantics of the extensive database is why this issue still exists. An RTI model for cloud platform photography based on visual semantic indexing will be presented in this paper. An interactive optimization model is first used to establish the typical semantic and visual descriptor space. The optimal strategy for searching for larger data sets is then determined by merging an RTI architecture with the semantic visual space-sharing model. Finally, the distributed model Spark is extended with an online image retrieval service. The effectiveness of the suggested system is confirmed in terms of average precision (mAP) and processing time in various data set sizes using two popular datasets, Holidays 1 M and Oxford 5 K. Nevertheless, neither machine learning nor computation time is enhanced by this method.

In the paper [5], the authors use a Dual Shot Face Detector (DSFD) to address three novel techniques: feature learning, loss design, and anchor matching, respectively. First, the author of this paper [5] introduced a feature enhancement module (FEM) that combines the advantages of FPN in Pyramid Box and the advantages of Receptive Field Block (RFB) in RFBNet [48] and the durability of properties. Secondly, the pyramid anchor [49] and hierarchical loss [50] in Pyramid Box served as inspiration for the author of this paper's Progressive anchor sizes, which are used for different shots and levels in Progressive Anchor Loss (PAL). They use larger sizes in the second shot and smaller sizes in the first, to be more exact. Lastly, they propose an approach known as Improved Anchor Matching (IAM), which enhances the regressor's initialization by combining an anchor partition strategy with anchor-based data augmentation to match anchors and ground truth faces more precisely. Because the three elements complement one another, these methods can be combined to enhance performance further. The popularity of smartphones and other low-end hardware has stoked a well-known desire for improved models compatible with these devices. Blaze Face is a sophisticated face detection algorithm that uses deep learning, is lightweight, and is highly effective. It is covered in my paper [6]. Its main goal is to recognize faces in real-time with speed and accuracy, which makes it perfect for use cases requiring quick processing. The Blaze Face algorithm consists of a single neural network at its core that recognizes faces at various scales and resolutions using a feature pyramid. Blaze Face can function with little computational resources and power because its design is optimized for mobile and embedded devices.

YOLO v3, mentioned in the paper [7], can perform object detection tasks quickly because it only needs one neural network to forecast class probabilities and bounding boxes. An entire neural network

method called YOLO v3 (You Only Look Once version three) predicts bounding boxes and class probabilities at the same time. This differs from previous algorithms' conventional approach for object detection, which modified classifiers for detection needs.

[39] Storage capacity is constrained, and classifiers cannot be directed toward the program that sets signal for video processing. [40] No attempt has been made to create classifiers that perform better or collect more data. [41] It cannot preserve important frames or handle time and space complexities. Such methods may not capture the faces accurately due to wide variations in lighting, facial expressions, and other factors [42, 43]. Converting video images into a coherent series of frames is challenging. [44], and complexity of time may arise [45]. [51] Neither the computation time nor the machine learning method are enhanced. In response to the problems above with video indexing and retrieval, Paper [4] proposes a novel technique known as "Video Indexing with Human Face Images. using LGFA and the Sliding Window Technique" to address the problems above. In this work, the key frame, or human face, is detected using the Viola- Jones algorithm, and the EAN-8 linear bar code is used to generate the bar code from the face image. Although the Viola-Jones algorithm outperforms MTCNN in speed, it cannot detect angular and small faces in videos. It also pointed out that inherent ambiguities like position changes, partial occlusion of facial cavities, and sensitivity to low resolution affect video-based recognition. The paper [53], "Video indexing through human face as a QR code using MTCNN algorithm," mentions a novel method for resolving the issue of paper [4]. In this paper, a QR code is used for video indexing using a human face as a cue, and the MTCNN algorithm and Viola Jones are used to detect key frames. However, for small face detection from a crowd, the MTCNN algorithm and Viola Jones do not perform well on lightweight devices. QR codes are more space-consuming to store on devices and transfer over any communication channel than EAN-8 linear bar codes, despite being smaller for face representation. To solve the issues above, this paper suggests a novel solution: "Video indexing through the human face as an EAN-8 linear bar code using machine learning and deep learning algorithms". This technique uses a linear EAN 8 bar code of the face image rather than the face image to index the human face found in different videos. Less time and storage space are needed for the EAN 8 linear bar code than for the original face image. However, compared to the original face picture, sending this bar code over the communication channel uses less bandwidth. However, stability is a disadvantage of facial image barcode representation. The present study employs the sliding window technique, as previously discussed in papers [28,52], to produce a reliable bar code. The MTCNN algorithm primarily identifies and aligns faces in angular key frames (faces) in this method. With its real-time face detection capabilities and high-speed optimization, DSFD finds utility in numerous applications, including security cameras and facial recognition systems. A multi-scale approach and a batch complex mining strategy are used to increase accuracy and manage complex cases. Blaze Face can function with little computational resources and power because its design is optimized for mobile and embedded devices. YOLO v3 can perform face detection tasks quickly from input video because a single neural network predicts bounding boxes for every class. An entire neural network method called YOLO v3 (You Only Look Once version 3) predicts bounding boxes and class probabilities at the same time.

Proposed Methodology and function of various algorithms

The method proposed in this text consists of several steps, each illustrated with a block diagram in Fig. -1. (a) Frame extraction is the initial step in the input video. (b) Key frames are distinguished from the frames taken from the input video by using the difference in the color histogram. (c) Machine learning and deep learning algorithms (Viola Jones, DSFD, Blaze face, YOLO V3, and MTCNN) identify faces in the key frame. (d) This key frame from a human face or a face detected by a human produces a linear EAN-8 bar code.

Frame Extraction from    Color Histogram  
input video                    extraction for key frame

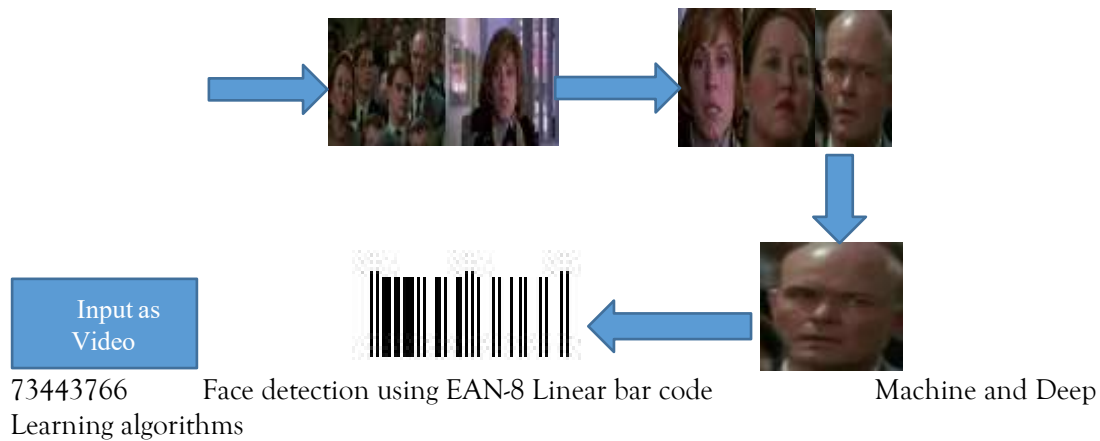


Fig.1: System Block Diagram Concept

Retrieving the frame:

A dynamic video combines the shot, frame, and scene. As a result, the first stage is extracting the still images, which are scenes, shots, and pictures represented by the input videos. A scene is an assembly of shots, and a shot is a set of frames. Traditional video has much content and runs at 20 to 30 frames per second. The image in the frame is still that appears in a video and is filled with unnecessary details. Figure 2 displays a frame from the Holly Wood Film Fargo.



Fig.2. The Fargo Video frame

Taking Out the Key Frame using a color histogram:

The most crucial details of every image are captured in the key frame. Key frames in this piece are characterized by human faces with unique expressions, poses, lighting, and illumination. The likelihood scale, the results of the curve saliency motion capture, and various other techniques are among the commonalities. Nonetheless, the key frame of a particular film is now extracted from each frame using the Color Histogram technique. Key frames can be extracted from the frames using the Difference in color histogram. When the color histogram disagreement threshold is reached, the frame is chosen as the next key frame if the observed difference exceeds the threshold's magnitude. Figure 3 displays a few key frames from the Holly Wood-starring Hollywood picture Fargo.



Fig. 3 Fargo's Key Frame, which centers on the face as the primary element

The formula and algorithm for splitting the two consecutive frames in the color histogram are described in a paper [3].

Face portraits from the key frames were cropped using machine learning and deep learning algorithms (Viola Jones, MTCNN, DSFD, Blaze-face, and YOLO v3):

Viola Jones, MTCNN, DSFD, Blaze-face, and YOLO v3 algorithms are used to detect faces from the key frame, compare the result, and identify the best face detection method from the input video's key frame. Several techniques have been developed over time to aid computers in recognizing faces. In this

work, the first method for identifying faces from the key frame is the Viola-Jones method, which uses Haar-Cascades to locate faces precisely. Regardless of an object's size or location within an image, the Haar cascade algorithm can detect it. It computes features in each cascading window to see if it might be an object. This is how it operates. A sample-sized window scan is used to evaluate and match Haar features.

Better techniques that can recognize the human face at different angles, lighting levels, and clarity levels have emerged with the introduction of deep learning. MTCNN is one of these techniques. MTCNN, or Multi-Task Cascaded Convolution Neural Network, is a face detection algorithm based on deep learning. It is very good at identifying faces in photos, even if they are partially hidden or have different sizes, orientations, or locations. Faces are recovered from the retrieved key frames using the MTCNN object detection technique. The algorithm uses a cascading technique combining three different neural networks to detect faces. The first network locates the face in the picture, the second network defines the face's contours, and the third network improves detection by identifying facial characteristics like the mouth, nose, and eyes. Face recognition, facial expression analysis, and facial attribute detection are just a few of the many applications that use MTCNN extensively.

A face detection system, the Dual Shot Face Detector (DSFD) algorithm, uses two distinct neural networks to identify faces in images. The first network produces a set of candidate face regions, and the second network then refines these regions to increase accuracy and reduce false positives. A multi-scale approach and a batch hard mining strategy are used to improve accuracy and manage complex cases.

The popularity of smartphones and other low-end hardware has stoked a well-known desire for improved models compatible with these devices. Blaze Face is a sophisticated, lightweight, and incredibly effective face-detection algorithm that uses deep learning. Its main goal is to recognize faces in real-time with speed and accuracy, which makes it perfect for use cases requiring quick processing. The Blaze Face algorithm consists of a single neural network at its core that recognizes faces at various scales and resolutions using a feature pyramid. Blaze Face can function with little computational resources and power because its design is optimized for mobile and embedded devices. Four key design considerations form the foundation of the Blaze Face model architecture: (a) expanding the receptive field sizes, (b) feature extraction, (c) anchor scheme, and (d) post-processing.

"You Only Look Once version 3," or "YOLO v3," is a real-time face detection algorithm that uses deep learning to quickly and reliably identify faces in pictures and videos. YOLO v3 splits the input image into a grid to detect faces and builds bounding boxes and class probabilities for every cell. The algorithm can identify multiple faces in a single image by predicting numerous bounding boxes per cell. YOLO v3 can quickly complete face detection tasks because a single neural network is sufficient for bounding box and class probability prediction. An all-encompassing neural network approach called YOLO v3 (You Only Look Once version 3) predicts bounding boxes and class probabilities at the same time. This is different from the conventional approach taken by previous methods for face detection, which modified classifiers for detection needs. A few screenshots of the key frame face detection feature are shown in Fig. 4.

This shows that the MTCNN algorithm has a higher face detection ratio than the Viola-Jones, DSFD,



Fig. 4: Dead Poets Society and Fargo key frames with faces cropped (from the movie Holly Hood)  
Image to grayscale conversion followed by a grayscale face image:

It is necessary to convert the picture to grayscale after obtaining the visages from the crucial frames to determine the facial image's gradient. Images of faces are, therefore, transformed into grayscale versions.

Sliding Window Technique image gradient:

From this grayscale image (faces), image gradients are computed using the Sliding Window method. This method's specifics were covered in papers [52] and [53]. When computing the image gradient values, the above method considers the upper 70% of the face image. The upper 70% of the face image is captured to generate a stable bar code. It was discussed in the paper [53] that If the sliding window moves up to 70% of the face image, a stable bar code can be generated from this gradient image.

Applying a human face index to a linear EAN-8 barcode:

A Linear EAN-8 bar code is generated from each face, identified from the input video's key frame, and stored as an index. Our approach uses the barcode as a linear depiction of the face image identified via video. These barcodes can index human faces in any video, as they are a linear representation of face images. Using barcodes derived from human faces requires less storage space and indexing time. Along with providing details of the bar code generation technique covered in papers [52] and [53], this paper also describes an algorithm for the bar code generation process. Examples of bar codes are displayed in Fig. 5. These were identified from various Hollywood face video datasets (Fargo and Dead Poets Society, respectively). The following algorithms are used to generate bar codes from the gradient values of face images that are obtained:





| Face   | Barcode Values | EAN-8 Barcode  |
|--|----------------|--|
|   | 99343435       |  |
|  | 73443766       |  |

Fig. 5: The cropped faces from the Dead Poets Society and Fargo video scenes and the matching EAN 8 barcodes.

The paper [3] discusses algorithms and algorithm accuracy for bar code generation from the face image gradient values that were obtained.

## RESULTS AND DISCUSSION OF THE EXPERIMENT

This section provides a detailed analysis of the system's functionality, strategies for comparison, and implementation outcomes.

Configuration for an experiment

The system specification needed to implement this work in Python 3.8 or later is below. The platform should be Python 3.8 or later.

Laptop or PC Processor i3 or higher 80 GB ROM or Higher 4GB of RAM

GPU is recommended

The cropped face of the key frame, which was taken from the human face-based video dataset, was used to generate the bar code for this experiment. Subsequently, the EAN- 8 linear bar code of the video's human visage is used for indexing. However, since the visage image key frame is explicitly present in some video datasets (such as the Face video dataset for FDDB, WIDER, and LFW), the EAN-8 linear bar code is generated from these datasets without removing the key frame. The method was validated using three distinct video data sets.

The Hollywood video dataset is used in the experiment's first step. Video snippets from thirty-two human action films are also included. The instance needs to be labeled using any of the eight available categories. The test set is split into two 12-film practice sets, combined to create the 20-film data set. With about 60% of the 233 video recordings in the automated learning set having accurate labels, automatic script-based action labeling was used to collect the data. Two hundred and ninety-one video samples with manually verified labels and two hundred and eleven videos with manually tested labels make up a clean training collection of Hollywood results. Labeled faces from the Faces in the Wild dataset comprise the Face Detection Dataset and Benchmark (FDDB) dataset. 5171 in all faces have been annotated, with the images varying in size from 229x410 to 363x450. A range of difficulties are present in the dataset, such as poor resolution, faces that are out of focus, and challenging stance angles. There are both color and grayscale images.

The publicly accessible WIDER dataset is the source of the benchmark face detection dataset known as WIDER FACE. As the sample images demonstrate, the size, location, and low contrast of the 32,203 photos in this data set—which identify 393,703 faces—vary significantly. The WIDER FACE dataset was planned using the 61 event classes. 40%, 10%, and 50% of the data are randomly chosen for training, verification, and test samples for each event class. The WIDER FACE data set, like the Caltech and MALF datasets, employs the PASCAL VOC dataset's evaluation metric. Lastly, data sets for LFW are gathered for the experiments. Unrestricted face identification is a problem that was investigated with the creation of the Labelled Faces in the Wild (LFW) database. The University of Massachusetts, Amherst researchers who created and maintained this database are listed in the Acknowledgments section with specific references. 13,233 internet-sourced photos were scanned, and 5,749 individuals were identified using the Viola-Jones face detector. For 1,680 of the people in the dataset, there are two or more different photo-graphs. Three kinds of "aligned" photos are included in the original database, in addition to four sets of LFW images. As the experiment ends, it is clear that MTCNN outperforms the YOLO v3, DSFD, Blaze Face, and Viola Jones algorithms regarding accuracy. This demonstrates that compared to the Viola-Jones, DSFD, Blaze Face, and YOLO v3 algorithms, the MTCNN algorithm has a higher face detection ratio.

The results of face detection on different data sets using the Viola-Jones (Haar Cascade), MTCNN, DSFD, Blaze Face, and YOLO v3 methods are shown in the following tables. Table 1 shows the number of frames detected, the number of faces detected, the ratio of face detection (measured in faces per millisecond), the number of EAN-8 linear bar codes, and the time required in milliseconds for several video clips from the Hollywood Data set (Considering 10 Video Data set of Hollywood movie). Tables 2, 3, and 4 discuss the time taken for face detection, the number of face detections, the face detection ratio, and the number of EAN-8 Linear bar codes on the FDDB, LFW, and WIDER data sets.

Table 1 Shows the number of frames detected, the number of faces detected, the ratio of face detection (measured in faces per millisecond), the number of EAN-8 linear bar codes, and the time required in milliseconds for several video clips from the Hollywood Data set (Considering 10 Video clip of Hollywood movie).

| Method Name  | Frames Detected | Time Taken (Millisecond) | Faces Detected | No. of EAN-8 Linear Barcode | Ratio (Face /Millisecond) |
|--------------|-----------------|--------------------------|----------------|-----------------------------|---------------------------|
| Haar-Cascade | 39883           | 3715.2667241096497       | 32452          | 32452                       | 8.7347                    |
| MTCNN        | 78717           | 2513.5215883255005       | 53478          | 53478                       | 21.2761                   |
| DSFD         | 114809          | 4937.640452861786        | 97292          | 97292                       | 19.7041                   |
| Blaze face   | 97292           | 929.1992914676666        | 26980          | 97292                       | 29.0357                   |
| YOLO         | 50652           | 13247.744824171066       | 39248          | 97292                       | 2.9626                    |

Table 2: Shows the Number of face detection, time taken for face detection (in milliseconds), Number of linear EAN-8 bar codes, and face detection ratio (Face per millisecond) on FDDB data sets

| Method Name | Faces Detected | No.of EAN-8 Linear Barcode | Time Taken (Millisecond) | Ratio(Face/Millisecond) |
|-------------|----------------|----------------------------|--------------------------|-------------------------|
|             |                |                            |                          |                         |

|              |       |       |                    |       |
|--------------|-------|-------|--------------------|-------|
| Haar-Cascade | 18697 | 18697 | 14061.437121391296 | 1.329 |
| MTCNN        | 19923 | 19923 | 8111.164217233658  | 2.456 |
| DSFD         | 20812 | 20812 | 14728.182427167892 | 1.413 |
| Blaze face   | 14632 | 14632 | 14835.403413057327 | 0.986 |
| YOLO         | 17736 | 17736 | 16226.314084529877 | 1.093 |

Table 3: Shows the Number of face detections, time taken for face detection (in milliseconds), Number of linear EAN-8 bar codes, and face detection ratio (Face per millisecond) on LFW data sets

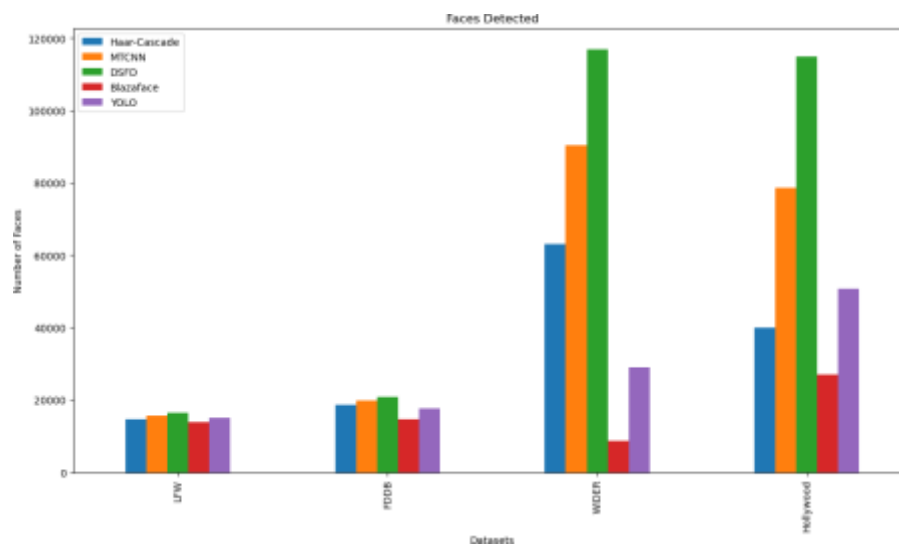
| Method Name  | Faces Detected | No. of EAN-8 Linear Barcode | Time Taken (Millisecond) | Ratio(Face/Millisecond) |
|--------------|----------------|-----------------------------|--------------------------|-------------------------|
| Haar-Cascade | 14759          | 14759                       | 1042.8512122631073       | 14.1525                 |
| MTCNN        | 15573          | 15573                       | 581.2226889133453        | 26.7935                 |
| DSFD         | 16426          | 16426                       | 3171.7797853946686       | 5.1787                  |
| Blaze face   | 13798          | 13798                       | 3292.380974292755        | 4.1908                  |
| YOLO         | 15069          | 15069                       | 5647.880829811096        | 2.6680                  |

Table 4: Shows the Number of face detections, time taken for face detection (in milliseconds), Number of linear EAN-8 bar codes, and face detection ratio (Face per millisecond) on WIDER Face data sets

| DSFD         | 20812          | 20812                       | 14728.182427167892       | 1.413                     |
|--------------|----------------|-----------------------------|--------------------------|---------------------------|
| Blaze face   | 14632          | 14632                       | 14835.403413057327       | 0.986                     |
| YOLO         | 17736          | 17736                       | 16226.314084529877       | 1.093                     |
| Method Name  | Faces Detected | No. of EAN-8 Linear Barcode | Time Taken (Millisecond) | Ratio (Face /Millisecond) |
| Haar-Cascade | 18697          | 18697                       | 14061.437121391296       | 1.329                     |
| MTCNN        | 19923          | 19923                       | 8111.164217233658        | 2.456                     |

### Comparative tactics

This section presents a performance contrasting the suggested approach with the following algorithms: YOLOv3, MTCNN, DSFD, Blaze face, and Viola Jones (Haar Cascade). Upon completion of the face detection process, it is evident that MTCNN outperforms other machine learning and deep learning algorithms discussed in our research paper regarding the accuracy of the face detection ratio from the input video. Upon comparing the results, it can be observed that while the MTCNN algorithm performs faster, DSFD and YOLO v3 detect more faces from the input video. From input, DSFD and YOLOv3 can identify the small face. After employing lightweight devices, Blaze Face and YOLO v3 can recognize faces. Face detection is faster with MTCNN. Because of this, the MTCNN algorithm's face detection ratio outperforms that of the DSFD, Viola Jones, Blaze-Face, and YOLOv3 algorithms. The main differences between the DSFD, YOLO v3, MTCNN, DSFD, and Haar-cascade algorithms are shown in a bar graph in Figure 6. This bar graph uses the LFW face dataset, FDDB face dataset, WIDER face data, and Hollywood movie video dataset to evaluate how well the previously mentioned algorithms performed. The graph displays the number of faces found using the algorithm discussed



above after the tests.

Fig. 6: Comparative analysis of the number of faces in the LFW face dataset, FDDB face dataset, WIDER face data, and Hollywood movie video datasets using the Haar-cascade, MTCNN, DSFD, Blaze-Face, and YOLO V3 algorithms.

The performance of the previously mentioned algorithms is compared in the bar graph in Figure 7 using the LFW face dataset, FDDB face dataset, WIDER face data, and Hollywood movie video dataset. Following the tests, the graph shows the time needed to identify the number of faces using the abovementioned algorithm.

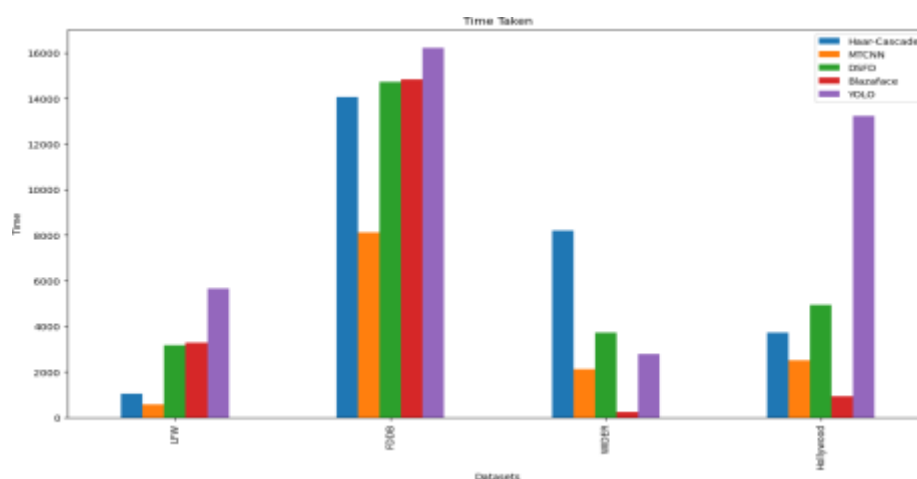


Fig.7: Based on the amount of time needed for face detection using the Haar- cascade, MTCNN, DSFD, Blaze-Face, and YOLO V3 algorithms, the LFW face dataset, Fddb face dataset, WIDER face data, and Hollywood movie video datasets were compared.

Comparing the MTCNN algorithm to Viola Jones, DSFD, Blaze Face, and YOLOV3, the former is less prone to face detection and more accurate. Furthermore, MTCNN is more effective at recognizing an excellent range of faces. When comparing MTCNN to YOLOV3, DSFD, Blaze Face, and Viola Jones, there is a noticeable decrease in frame rate. Following facial recognition using every machine learning and deep learning algorithm discussed in our research, a linear EAN-8 bar code is produced from each unique face image for indexing purposes. One of the benefits of using human-face linear EAN-8 bar codes for indexing is that they take less time to index and require less storage space. It is also possible to search the human face after using the bar code reader to scan this Linear EAN-8 bar code.

## CONCLUSION

Deep learning and machine learning are applied in image processing for video indexing and retrieval to enhance storage capacity for key frames from videos and lessen the time and space required to capture angular faces from videos. To address these issues, the MTCNN Algorithm has been proposed for use as a face detector, and the linear EAN-8 bar code is used as a bar code to face index. Furthermore, the case of illumination of the invariant facial picture is invented by employing this format, and the computation is straightforward. The recommended technique decreased time and space complexity while simultaneously increasing storage capacity. The video indexing and retrieval methods are thus successfully enhanced in the recommended manner. The Hollywood video, Fddb, LFW, and WIDER face datasets were used in the test. An account's personal search, affirmation, and authentication can be defined using the video indexing technique.

## REFERENCES

- Stefan Eickeler, Frank Wallhoff, Uri Iurgel, Gerhard Rigoll, "Content-based indexing of images and video using face detection and recognition methods," published in ICASSP 2001, IEEE Xplore.
- Lorenzo Baraldi, Costantino Grana, and Rita Cucchiara, "Neural Story: an interactive Multimedia system for Video indexing and re-use," In proceedings of CBIM, Florence, Italy, June 19-21, 2017.
- Ghatak, S., Bhattacharjee, D. (2021). Video Indexing Through Human Face. In: Sabut, S.K., Ray, A.K., Pati, B., Acharya, U.R. (eds) Proceedings of International Conference on Communication, Circuits, and Systems. Lecture Notes in Electrical Engineering, vol. 728. Springer, Singapore. [https://doi.org/10.1007/978-981-33-4866-0\\_13](https://doi.org/10.1007/978-981-33-4866-0_13).
- Ghatak, S., Bhattacharjee, D. Video indexing through human face images using LGFA and window technique. *Multimedia Tools Appl* 81, 31509–31527 (2022). <https://doi.org/10.1007/s11042-022-12965-2>.
- J. Li et al., "DSFD: Dual Shot Face Detector," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 5055-5064, doi: 10.1109/CVPR.2019.00520. keywords: {Recognition; Detection; Categorization; Retrieval; Face; Gesture; and Body Pose}.
- Blaze face: Sub-millisecond neural face detection on mobile gpus V Bazarevsky, Y Kartynnik, A Vakunov, K Raveendran, M Grundmann arXiv preprint arXiv:1907.05047, 2019 • arxiv.org.
- Redmon, J. and Farhadi, A. (2018) YOLOv3: An Incremental Improvement. *Computer Science*, arXiv: 1804.02767. <http://arxiv.org/abs/1804.02767>.
- P. Viola and M. J. Jones, "Robust real-time face detection. *International journal of computer vision*," vol. 57, no. 2, pp. 137-154, 2004.
- B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Aggregate channel features for multi-view face detection," in *IEEE International Joint Conference on Biometrics*, 2014, pp. 1-8.
- M. T. Pham, Y. Gao, V. D. D. Hoang, and T. J. Cham, "Fast polygonal integration and its application in extending haar-like features to improve object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 942-949.
- Q. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in *IEEE Computer Conference on Computer Vision and Pattern Recognition*, 2006, pp. 1491-1498.
- M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool, "Face detection without bells and whistles," in *European Conference on Computer Vision*, 2014, pp. 720-735.
- J. Yan, Z. Lei, L. Wen, and S. Li, "The fastest deformable part model for object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2497-2504.
- X. Zhu, and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2879- 2886.
- Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097-1105.
- Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in

- Advances in Neural Information Processing Systems, 2014, pp. 1988-1996
- S. Yang, P. Luo, C. C. Loy, and X. Tang, "From facial parts responses to face detection: A deep learning approach," in IEEE International Conference on Computer Vision, 2015, pp. 3676-3684.
  - H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5325-5334.
  - X. P. Burgos-Artizzu, P. Perona, and P. Dollar, "Robust face landmark estimation under occlusion," in IEEE International Conference on Computer Vision, 2013, pp. 1513-1520.
  - X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," International Journal of Computer Vision, vol 107, no. 2, pp. 177-190, 2012.
  - J. Zhang, S. Shan, M. Kan, and X. Chen, "Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment," in European Conference on Computer Vision, 2014, pp. 1-16.
  - T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 6, pp. 681-685, 2001.
  - X. Yu, J. Huang, S. Zhang, W. Yan, and D. Metaxas, "Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model," in IEEE International Conference on Computer Vision, 2013, pp. 1944-1951.
  - Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in European Conference on Computer Vision, 2014, pp. 94-108.
  - D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun, "Joint cascade face detection and alignment," in European Conference on Computer Vision, 2014, pp. 109-122.
  - Zhang, and Z. Zhang, "Improving multiview face detection with multi-task deep convolutional neural networks," IEEE Winter Conference on Applications of Computer Vision, 2014, pp. 1036-1041.
  - K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," in IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, Oct. 2016, doi: 10.1109/LSP.2016.2603342.
  - Y. Matveev, G. Kukharev, N. Shchegoleva, 'A simple method for generating facial barcodes', in WSCG2014 Conference on Computer Graphics, Visualization and Computer Vision in Co-operation with EUROGRAPHICS Association Exchange Anisotropy (Academic, Czech Republic, 2014), pp. 213-220.
  - Cees G.M. Snoek, Marcel Worring, "Multimodal video indexing: A Review of state of the art," Multimedia Tools and Applications, 25, 5-35, 2005
  - Yuehua Wan<sup>1</sup>, Shiming Ji<sup>1</sup>, Yi Xie<sup>2</sup>, Xian Zhang<sup>1</sup>, and Peijun Xie "Video program clustering indexing based on faced recognition hybrid model of Hidden Markov model and support vector machine," IWCIA 2004, LNCS 3322, pp. 739-749, 2004.
  - Lorenzo Baraldi, Costantino Grana, and Rita Cucchiara, "Neural Story: an interactive Multimedia system for Video indexing and re-use," In proceedings of CBIM, Florence, Italy, June 19-21, 2017.
  - Bor-Chun Chen, Yan\_Ying Chen, Yin-His Kuo, Thanh Duc Ngo, Duy-Dinh Le, Shin Ichi Satoh, Winston H Hsu, "Scalable face Track Retrieval in Video archives using Bag-of-faces sparse Representation," IEEE Transactions on Circuits and Systems for video technology, 2015
  - Zhen Dong, Su Jia, Tianfu Wu, and Mingtao Pei, "Face video Retrieval via Deep learning of binary hash Representations "Proceeding of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16).
  - Li Y, Wang, R, Huang Z, Shan S, and Chen X, "Face video retrieval with image query via hashing across Euclidean space and Riemannian manifold," In CVPR, 4758-4767, IEEE, 2015b.
  - Chen Y.C, Patel V.M, Shekhar S, Chellappa R., and Phillips P.J "Video-based face recognition via sparse joint representation," In FG, 1-8, IEEE, 2013.
  - Stefan Eickeler, Frank Wallhoff, Uri Lurgel, Gerhard Rigoll, "Content-based indexing of images and video using face detection and recognition methods," published in ICASSP 2001, IEEE Xplore.
  - Usman Saeed, Jean-Luc Dugely, "Temporally consistent key frame selection from video for face recognition," 18th European signal processing conference, 23-27th Aug. 2010, IEEE Xplore, 30th April 2015.
  - Csaba Czirik, Noel O'Connor, Sean Marlow, and Noel Murphy, "Face detection and clustering for video indexing applications" In ACIVS 2003 - Advanced Concepts for Intelligent Vision Systems, 2-5 September 2003
  - Gayathri N, Mahesh K (2020) Improved fuzzy-based SVM classification system using feature extraction for video indexing and retrieval. International Journal of Fuzzy Systems 22:1716-1729
  - Lin FC, Ngo HH, Dow CR (2020) A cloud-based face video retrieval system with deep learning. J Supercomput 76(11):8473-8493
  - Li C, Zhou B (2020) Fast key-frame image retrieval of intelligent city security video based on deep feature coding in high concurrent network environment. Journal of ambient intelligence and humanized computing 1-9.
  - Bastanfard A, Takahashi H, Nakajima M (2004) Toward E-appearance of human face and hair by age, expression and rejuvenation. International Conference on Cyberworlds. IEEE
  - Bastanfard A, Bastanfard O, Takahashi H, Nakajima M (2004) Toward anthropometrics simulation of face rejuvenation and skin cosmetic. Computer Animation and Virtual Worlds 15(3-4):347-352
  - "Create caption by extracting features from image and video using deep learning model," International Journal of Emerging Technologies and Innovative Research ([www.jetir.org](http://www.jetir.org)), ISSN:2349-5162, Vol.8, Issue 1, page no. 842-855, January-2021, Available:<http://www.jetir.org/papers/JETIR2101113.pdf>.
  - Jacob J, Sudheep Elayidom M, Devassia VP (2020) Video content analysis and retrieval system using video storytelling and indexing techniques. International Journal of Electrical & Computer Engineering 10(6): 6019

- Krishnaraj N, Elhoseny M, Lydia EL, Shankar K, and Aldabbas O (2020) An efficient radix trie-based semantic visual indexing model for large-scale image retrieval in cloud environment. *Software: Practice and Experience*
- Information Technology-Automatic Identification and Data Capture Techniques-QR code Bar code symbology specification (Adopted ISO/IEC 18004: 2015, Third Edition, 201502- 01)
- Songtao Liu, Di Huang, and Yunhong Wang. Receptive field block net for accurate and fast object detection. In *Proceedings of European Conference on Computer Vision*, 2018. 2, 4
- Xu Tang, Daniel K Du, Zeqiang He, and Jingtuo Liu. Pyramid box: A context-assisted single-shot face detector. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2018. 2, 3, 5
- Jialiang Zhang, Xiongwei Wu, Jianke Zhu, and Steven CH Hoi. Feature agglomeration networks for single-stage face detection. *arXiv preprint arXiv:1712.00721*, 2017. 2, 3
- Nagappan, Krishnaraj & Elhoseny, Mohamed & Lydia, Laxmi & Shankar, K. & ALDabbas, Omar. (2020). An efficient radix trie-based semantic visual indexing model for large-scale image retrieval in cloud environment. *Software: Practice and Experience*. 51. 10.1002/spe.2834.
- S. Ghatak, "Facial representation using linear barcode," In book *Springer Nature, Advanced computational and Communication Paradigms*, Vol.-2, PP.791-801.21 April 2018.
- Ghatak, S., Kollman, C., Bhattacharjee, D. (2024). Video Indexing Through QR Code of Human Faces Using MTCNN Algorithm. In: Das, N., Khan, A.K., Mandal, S., Krejcar, O., Bhattacharjee, D. (eds) *Proceedings of International Conference on Data, Electronics and Computing. ICDEC 2023. Lecture Notes in Networks and Systems*, vol 1103. Springer, Singapore. [https://doi.org/10.1007/978-981-97-6489-1\\_1](https://doi.org/10.1007/978-981-97-6489-1_1)