Faces Unveiled: A Deep Dive into Modern Face Detection and Recognition Techniques

DAA Deepal[1], MKA Ariyaratne[2], PR De Silva[2] and TGI Fernando[2]

[1]Faculty of Graduate Studies, University of Sri Jayewardenepura.
[2] Department of Computer Science, Faculty of Applied Sciences, University of Sri Jayewardenepura.
Email: mkanuradha@sjp.ac.lk

ABSTRACT

This paper provides a comprehensive overview of contemporary research in face detection, facial feature detection, and face recognition, categorizing methodologies into four primary types: knowledge-based, template matching, feature-based, and appearance-based. Analysis reveals a predominant focus on appearance-based techniques, particularly in recent studies. Literature showcases the increasing utilization of deep learning algorithms, such as CNN, DCNN, and Faster RCNN, to address challenges in face detection and recognition. Notably, these algorithms demonstrate high accuracy in complex scenarios, including variations in pose, scale, and occlusion. The overview highlights the effectiveness of knowledge-based methods in detecting facial features with low computational requirements, albeit with limited accuracy in complex situations. Appearance-based methods, particularly those employing deep learning, emerge as highly successful in face detection and recognition, achieving accuracy rates exceeding 99%. The integration of one-stage and two-stage algorithms, coupled with traditional classifiers, underscores their efficacy. Researchers enhance accuracy through data augmentation, multi-task learning, and network acceleration techniques. The paper concludes that deep learning algorithms significantly impact face detection, recognition, and feature extraction, reflecting their pivotal role in advancing computer vision. The comprehensive review of 28 selected papers emphasizes the importance of continued research to further enhance these essential aspects of object detection.

INDEX TERMS : Face Detection, Facial Feature Detection, Deep Learning, Review

I. INTRODUCTION

Detecting and classifying objects in digital images and videos is a critical task in computer vision, with face detection and recognition being among its most significant applications. Object detection techniques have evolved over the past two decades, transitioning from rule-based methods [1], feature-based methods [2], and region-based methods [3] to modern deep learning algorithms after 2020. While these advancements have significantly improved accuracy and computational efficiency, several **challenges** persist, particularly in handling diverse environmental conditions, real-time processing, and scalability across datasets.

Despite the success of deep learning methods such as Convolutional Neural Networks (CNNs), Single Shot Detector (SSD) [14], Faster R-CNN [15], and YOLO [16], gaps remain in comprehensively comparing these techniques, especially in their suitability for specific tasks like real-time face recognition or handling occlusions. Moreover, there is a lack of systematic reviews that delve into the underlying methods, datasets, and metrics in a way that bridges the gap between theoretical advancements and practical applications.

Problem Statement:

Traditional face detection methods often struggle with low accuracy in challenging scenarios, such as poor lighting, extreme poses, or occlusions. Although deep learning-based techniques have addressed many of these issues, there is a need for a **comprehensive review** that categorizes and evaluates these methods based on key factors such as:

- Robustness to environmental changes.
- Dataset requirements and performance evaluation.
- Computational efficiency and real-time feasibility.

Objectives:

The primary objective of this survey is to provide a systematic and comprehensive review of face detection and recognition techniques, with a focus on:

1. Categorizing the techniques into knowledge-based, feature-based, template-matching, and appearance-based methods.
2. Evaluating the performance of state-of-the-art algorithms, such as CNNs, Faster R-CNN, YOLO, and SSD, across various tasks (e.g., front face detection, feature point detection).

3. Discussing the datasets and evaluation metrics used for training and testing these algorithms.
4. Highlighting the strengths, limitations, and suitability of each algorithm for specific applications.

Rationale:
While several surveys have been conducted on object detection [22–25] and face recognition [26–27], they often lack a nuanced comparison of emerging algorithms and their applications. This survey aims to address these gaps by:

- Examining the evolution of face detection and recognition techniques, from traditional methods to deep learning-based approaches.
- Providing practical insights into dataset selection, algorithm suitability, and performance metrics.
- Offering a resource for researchers and practitioners to navigate the complexities of modern face recognition systems.

The remainder of this paper is structured as follows: Section 2 outlines the methodology used to conduct this systematic review. Section 3 discusses the applications, datasets, and evaluation metrics in face detection. Section 4 categorizes object detection and recognition techniques based on their underlying methods. Section 5 explores the categories of algorithms specifically used for face detection and recognition. Finally, Section 6 concludes with a summary and discussion.

II. MATERIALS AND METHODS FOR THE SYSTEMATIC REVIEW

To conduct this systematic review, we adhered to the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines to ensure transparency and reproducibility of the research process [28].

Research Objectives and Scope

The primary objective of this review was to comprehensively analyze advancements in **face detection** and **recognition technologies**, identify **challenges** in the field, and assess **state-of-the-art techniques**. Key focus areas included:

- The datasets used for training and testing algorithms.
- The reported accuracy of algorithms.
- Drawbacks and challenges associated with various techniques.
- Key methods employed to improve algorithm accuracy.

Search Strategy and Data Sources

Our systematic approach began with the identification of relevant research articles using a predefined set of keywords, including "face detection," "face recognition," "knowledge-based face detection," "feature-based face detection," "template matching," "appearance-based methods," "convolutional neural networks," "Faster R-CNN," "YOLO," and "deep learning-based approaches." To maximize the scope of our review, we searched popular repositories such as **Google Scholar**, **Scopus**, and **Semantic Scholar** for literature published between **1990 and September 2023**.

Inclusion and Exclusion Criteria

We adopted the following criteria to systematically include and exclude studies:

- **Inclusion Criteria**:
    - Peer-reviewed research articles, review articles, book chapters, and conference proceedings written in English.
    - Studies addressing face detection, recognition, facial feature extraction, or algorithms trained/tested on relevant datasets.
    - Publications discussing single or multiple face scenarios in real-time or static images.
- **Exclusion Criteria**:
    - Non-English studies without available translations.
    - Duplicates or redundant studies identified during the screening process.

- Articles unrelated to face detection or recognition, such as studies focusing solely on non-facial object detection.

Screening and Selection Process

Through an extensive web search, **37 articles** were initially identified. After screening for duplicates and applying the inclusion and exclusion criteria, **31 articles** were selected for detailed review. An additional **6 articles** were identified through references within the selected papers. Ultimately, **2 articles** were excluded due to redundancy, lack of translation, or irrelevance. This process is summarized in **Figure 2**.

Data Extraction and Analysis

Each selected article was carefully analyzed and categorized into four primary methods:

1. **Knowledge-based methods**.
2. **Template matching methods**.
3. **Feature-based methods**.
4. **Appearance-based methods**.

The analysis focused on:

- The datasets used to train and test algorithms.
- The reported accuracy and performance metrics.
- Challenges and limitations of the proposed approaches.
- Strategies for improving algorithm accuracy.

This systematic categorization and analysis provide insights into the evolution and trends within the field, highlighting advancements, challenges, and solutions that shape the landscape of facial recognition technology.

, further summarizes the selection of the articles.

Table 1 Summary of the search process

Duration of the search	Used research repositories	Keywords	Type of research work
March 2022 to September 2023	Google Scholar, Scopus, Semantic Scholar	face detection face recognition Knowledge-based face detection and recognition Feature-based face detection and recognition Template Matching face detection and recognition Appearance-based face detection and recognition Convolutional Neural Networks Faster R-CNN YOLO Deep learning-based approaches for face detection and recognition	research articles, review articles, book chapters, conference materials

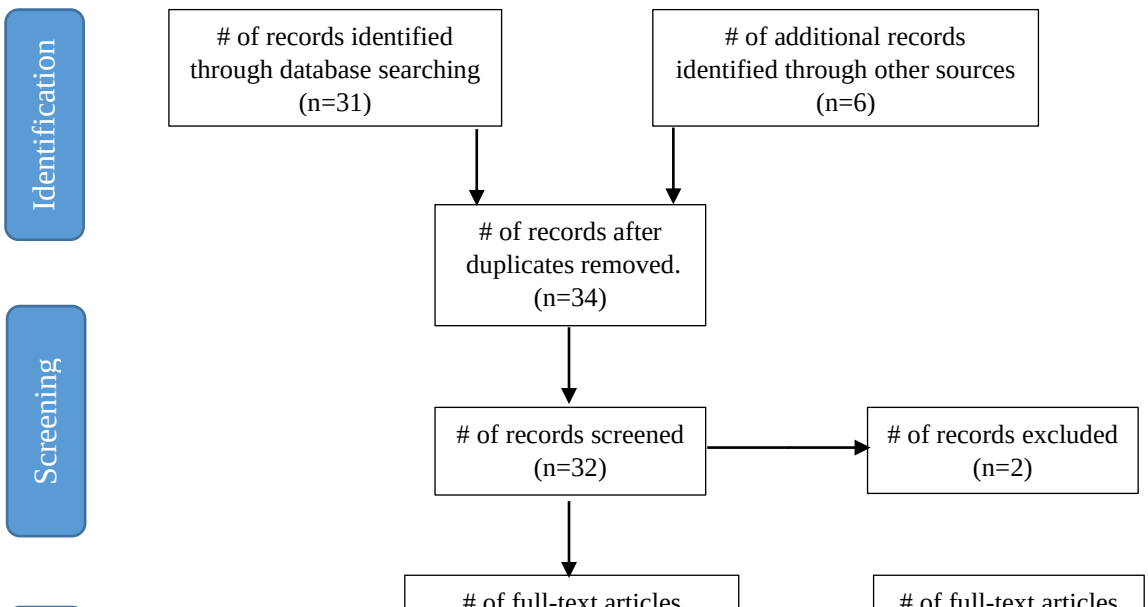

Figure 1 Flow chart of the comprehensive review process based on the PRISMA.

III. APPLICATIONS, RESOURCES, AND MEASURING METHODS FOR FACE DETECTION USING MACHINE LEARNING ALGORITHMS

Before a discussion of the previous studies, this section aims to provide a detailed description of the applications where machine learning-based face detection has been used and the importance, available resources used in research literature for such research, and, what kind of approaches have been taken to measure the quality of the algorithms.

1) Applications

Over the years, face detection and recognition technologies have been used in several applications across various industries. It is commonly used in security and surveillance systems for authentication and tracking purposes [29], [30], [31] [32]. Biometric identification also uses face detection to unlock devices and grant access to secure systems and attendance systems [33] [34] [35]. Social media platforms and photo-editing software also use this technology to identify faces in images, tag individuals, and apply filters and effects [36] [37] [38]. Gaming applications use face detection to capture facial expressions and movements of players for controlling characters or game elements [39] [40]. Marketing and advertising industries use this technology to target ads based on facial expressions and emotional responses [41] [42]. Additionally, face detection is used in medical diagnosis to identify and track facial features, such as skin lesions, and to analyze changes over time [43] [44].

Face detection technology can also be used for gender classification and age estimation [45] [46]. This technology can accurately detect facial features to determine a person's gender and estimate their age. Face detection and recognition are important in the field of social robotics for effective human-robot interaction [47] [48]. Gender estimation and facial expression analysis also play important roles in attracting user attention and providing personalized services, such as adjusting lighting and temperature based on an individual's facial features [49]. The large application domain further proves the importance of face detection and the need for a good survey encapsulating most of such recent advancements.

2) Available Datasets/ Resources /Programs/Software/etc.

At present, face detection applications can be implemented either using pre-trained models or by training customized models. There are several resources available to train new models, including face detection datasets like the WIDER FACE and CelebA datasets, as well as open-source libraries such as OpenCV and TensorFlow. This section briefly discusses the available resources, datasets, software, etc.

2.1 Datasets

2.1.1 DigiFace-1M

The DigiFace-1M dataset, introduced by Bae et al. [50] in 2023, contains over one million diverse synthetic face images for face recognition. It includes 720,000 images with 10,000 unique identities, each with 72 images generated using four sets of accessories. Additionally, it has 500,000 images with 100,000 unique identities, each with five images generated using one set of accessories. The dataset outperforms SynFace, a leading method for synthetic face recognition [50], demonstrating its robustness across various datasets. Fine-tuning with a smaller set of real-face images further enhances model accuracy.

2.1.2 FDDB: Face detection data set and benchmark

The FDDB dataset, introduced by Jain and Learned-Miller [51]at the University of Massachusetts Amherst, is a

collection of 2,845 labeled images featuring 5,171 faces for face detection. The images include varying resolutions, orientations, lighting conditions, out-of-focus faces, difficult poses, and low-resolution images. The FDDB is widely used in research, including by Zhang et al. [52], who trained and evaluated their MTCNN framework on this dataset, achieving superior accuracy in face detection and alignment. Sun X et al. [53] and Jiang H et al. [54] also utilized the FDDB dataset to test and fine-tune their respective Faster R-CNN and enhanced Faster RCNN models for face detection.

2.1.3 WIDER FACE

The WIDER FACE dataset, presented by Yang et al. [55] in 2016, includes 32,203 images and 393,703 labeled faces, selected to represent a diverse range of scales, poses, and occlusions. Wang J et al. [56] trained and evaluated the Face Attention Network (FAN) on this dataset, achieving an average precision (AP) of 94.6% (easy) and 88.5% (hard). Additionally, Sun X et al. [53] trained their Enhanced Faster RCNN approach on the WIDER FACE dataset, demonstrating its adaptability and effectiveness by generating hard negatives during testing.

2.1.4 CelebFaces Attributes Dataset (CelebA)

The CelebA dataset, introduced by Liu et al. [57] in 2015, is a large-scale face attributes dataset comprising over 200,000 images of famous persons. Each image contains 5 landmark locations and 40 binary attribute annotations, covering a wide range of pose variations and background clutter. With over 10,000 unique identities and 202,599 face images, CelebA is suitable for tasks such as face detection, recognition, landmark localization, and attribute recognition. Ranjan et al. [58] introduced the Deep Pyramid Single Shot Face Detector (DPSSD), a high-speed model effective for detecting faces with significant scale variations, including tiny faces. Their deep learning pipeline showed exceptional performance in face identification and verification across various benchmarks, including CelebA. Yang et al. [59] proposed a deep convolutional neural network (CNN) for face detection with facial attributes-based supervision. They enhanced their attribute-aware networks using the CelebA dataset for training, fine-tuning the model with a substantial number of face and non-face images from CelebA.

2.1.5 VGG Face2

The Visual Geometry Group introduced the VGG Face2 dataset [60] in 2018 at the University of Oxford. It is one of the largest datasets for face recognition, containing over 3.3 million face images and 9,131 unique identities. Cao et al. [60] trained ResNet-50 Convolutional Neural Networks, both with and without Squeeze-and-Excitation blocks, on VGGFace2, demonstrating that training on VGGFace2 significantly enhances recognition performance, especially with pose and age variations. Aghdam et al. [61] investigated factors affecting the identification performance of deep face recognition models under low-resolution and mismatched conditions, using models trained on MS-Celeb-1M and fine-tuned on VGGFace2. This approach achieved state-of-the-art accuracies on the SCFace and ICB-RW benchmarks, highlighting VGGFace2's effectiveness in addressing challenges related to appearance variety and low-resolution face recognition.

2.1.6 Flickr-Faces-HQ Dataset (FFHQ)

The Flickr-Faces-HQ Dataset (FFHQ), created by researchers at NVIDIA [62], contains 70,000 high-resolution human face images sourced from Flickr. It offers a varied set of images in terms of poses, ages, and ethnicities, making it ideal for training and evaluating face-related computer vision models. Karras et al. [63] redesigned the generator architecture for generative adversarial networks (GANs) and conducted extensive experiments using the FFHQ dataset. Their method significantly improved control over image synthesis, achieving state-of-the-art results in high-quality image generation. Bencheriet et al. [64] introduced a robust approach to fake face detection using a Deep CNN architecture with three distinct discriminators, each trained differently to enhance performance. They trained and evaluated their system on a dataset combining authentic faces from the FFHQ dataset and 70,000 synthetic faces generated with Nvidia's StyleGAN. Their system achieved impressive accuracy rates of 96% for detecting fake faces and 98% for identifying real faces.

2.1.7 Tufts Face Dataset

The Tufts Face Dataset [65] is a unique and extensive dataset offering seven distinct image modalities for face recognition: thermal, near-infrared, visible, LYTRO, computerized sketch, 3D images, and recorded video. It includes 10,000 images of 112 individuals (38 males and 74 females) from over 15 countries, aged 4 to 70 years. This dataset is valuable for benchmarking and evaluating algorithms across multiple modalities, such as thermal, sketches, heterogamous face recognition, and 3D face recognition. Martins et al. [66] used the Tufts dataset to develop a multi-spectral face recognition system. They tested three SSD-based methods: the S3FD algorithm, the facial detection deep neural network of OpenCV, and the DSFD algorithm. The system achieved impressive Rank-1 scores of 99.5% for pose variations and 99.6% for expression variations in the Tufts database.

2.1.8 Labeled Faces in the Wild (LFW) Dataset

The LFW Dataset [67] is a vital resource for researchers in unconstrained face recognition, specifically designed for studying face verification or "pair matching." It contains over 13,000 face images gathered from the Internet, featuring diverse faces with varying sizes, poses, and illumination conditions, and has a total size of 173 megabytes. Almabdy et al. [68] conducted a comprehensive study on face recognition, evaluating the performance of three CNN-based methods across various image databases, including LFW. Sun et al. [69] proposed a hybrid model for face verification in unconstrained conditions, combining convolutional networks (ConvNets) with Restricted Boltzmann Machines (RBMs), and tested their method on the LFW dataset. Additionally, Sanchez et al. [70] presented an efficient facial recognition

system for unconstrained environments, achieving an impressive accuracy of 99.7% on the LFW dataset.

2.1.9 UTKFace

The UTKFace dataset [71] is a versatile resource for various computer vision tasks related to faces, such as age estimation, gender classification, and facial landmark localization. It comprises over 20,000 face images spanning diverse ages, ethnicities, and variations in pose, illumination, and expression. Additionally, the dataset provides aligned and cropped faces along with landmark annotations for 68 points, making it suitable for training and evaluating machine learning models in facial recognition research. Nugroho et al. [72] conducted a significant study analyzing the impact of different color spaces on face detection accuracy and efficiency. They validated their method using the UTKFace dataset. Devaraj et al. [73] focused on effectively classifying individuals by ethnicity, employing datasets including UTKFace for training their Convolutional Neural Network (CNN) model. Their methodology, involving image preprocessing and CNN utilization, achieved an average accuracy of 88% upon rigorous evaluation.

2.1.10 Google Facial Expression Comparison Dataset

Introduced by Vemulapalli and Agarwala in 2018, the Google Facial Expression Comparison (FEC) dataset [74] contains approximately 87,517 unique photos with around 500K facial image triplets. Each triplet is annotated by humans to identify the two faces most similar in terms of facial expression. Vemulapalli et al. [75] proposed a novel approach for efficiently capturing similarities in facial expressions, benefiting applications such as expression retrieval, photo album summarization, and emotion recognition. Their trained network achieves an impressive 81.8% accuracy in predicting the most similar pair within a triplet from the FEC dataset.

2.1.11 YouTube Faces Dataset with Facial Keypoints

The YouTube Faces Dataset [76] comprises short videos of celebrities sourced from YouTube, with faces cropped and multiple frames extracted to create a collection of face images. This processed version also includes facial keypoint annotations, facilitating detailed analysis of facial expressions and movements. Almabdy et al. [68] conducted a thorough study on face recognition using three distinct CNN-based methods, evaluating their performance on various image databases, including the YouTube Faces dataset. FaceNet, introduced by Schroff et al. [77], capable of face verification, recognition, and clustering using deep convolutional networks, achieved a high accuracy of 95.12% on the YouTube Faces DB, among other datasets.

2.2 Other datasets

In the literature, we have identified a series of face databases that have been used to evaluate the performance of models. The Yale Face Database [78] contains grayscale images capturing individuals under different lighting conditions, pioneering research in illumination-invariant face recognition. The European ACTS M2VTS [79] dataset offers video sequences depicting individuals in different poses and lighting conditions. The Cohn-Kanade database [80] provides facial expression sequences of varying intensities. The AR face database [81] features frontal face images under diverse illumination and expressions. For large-scale research, the PubFig face verification datasets [69] offer celebrity face images, while the CUFS/CUFSF [5] datasets capture student faces with pose, illumination, and expression variations. The CASIA NIR-VIS 2.0 [5] dataset aids in cross-spectrum face recognition research by providing near-infrared and visible-light images of the same individuals. GTAV Face [68] offers real-world complexity by extracting faces from Grand Theft Auto V. AFW [82] focuses on facial attribute analysis, providing gender, age, and facial hair annotations.

Other datasets address specific challenges like pose and illumination invariance (MIT-CBCL, CMU PIE [83]), aging effects (CAS-PEAL-R1 [82]), and real-world complexities such as low resolution and occlusion (VOC2012 [82], IJB-A [54]). MAFA [56] and MALF [58] explore subtle facial movements via facial action coding, showcasing the diverse research enabled by these resources. These datasets, each with unique strengths and limitations, emphasize the multifaceted nature of facial analysis research, offering rich opportunities for exploration and refinement in this dynamic field.

2.3 Resources

2.3.1 TensorFlow.js

TensorFlow.js, a JavaScript library developed by Google [84], brings machine learning models directly to the browser or Node.js environment. It offers various pre-trained models for tasks like image classification, object detection, semantic segmentation, face detection and recognition, face landmarks detection, pose detection, body segmentation, hand pose detection, natural language processing, and speech recognition. Moreover, developers can fine-tune existing models with custom datasets and build/train models directly in JavaScript using flexible APIs. Face-api.js [85], built on TensorFlow.js, provides a comprehensive JavaScript module for face-related tasks like detection, recognition, landmark detection, expression recognition, age estimation, and gender recognition. It enables seamless integration of advanced facial recognition functionalities into web applications and Node.js projects. Additionally, the @tensorflow-models/blazeface [86] repository enhances this ecosystem with pre-trained models tailored for TensorFlow.js, easily accessible via NPM or unpkg. These models expand developers' capabilities, allowing effortless integration of state-of-the-art facial detection and recognition mechanisms.

2.3.2 OpenVINO

OpenVINO, an open-source toolkit developed by Intel [87], optimizes deep learning models to run efficiently on Intel CPUs, GPUs, and accelerators like FPGAs. It offers libraries and tools for developing, optimizing, and deploying models for tasks like image classification, object detection, face recognition, and segmentation. The Open Model Zoo provides free, pre-trained models and demo applications usable with Python, C++, or OpenCV Graph API (G-API). Wang and Hu

[88] utilize OpenVINO to optimize CNN inference speed in their intelligent lecture recording system, enhancing face detection performance, especially with MobileNet-SSD. This optimization enables efficient lecturer tracking even during rapid movements. Dane Brown's [89] study explores mobile attendance systems using face detection and recognition, powered by OpenVINO on a Raspberry Pi platform. Despite positioning constraints, the system achieves remarkable recognition accuracy and processing speed, demonstrating OpenVINO's versatility and efficacy in real-world applications.

2.3.3 Face-api.js

Face-api.js [85] is a JavaScript module for face detection, recognition, landmark detection, expression recognition, age estimation, and gender recognition in the browser and Node.js. It leverages TensorFlow.js and provides high accuracy and speed using pre-trained machine-learning models. The library supports various input sources such as image files, video streams, and webcams, making it versatile for different use cases. Being open-source and actively maintained, it has a growing community of contributors and users continuously improving its capabilities. Basurah et al. [90] highlight the importance of liveness detection in thwarting spoofing attempts in facial recognition systems. Using TensorFlow.js and face-api.js, they implement a method for detecting facial movements, achieving an impressive 85% accuracy for face recognition. Yadav et al. [91] explore an In-Browser Attendance System showcasing the versatility of face-api.js in real-world applications. Powered by serverless edge computing, their system seamlessly integrates face detection and recognition functionalities into web browsers, enhancing efficiency and eliminating backend latency issues.

2.3.4 OpenCV

OpenCV (Open Source Computer Vision Library) [92] is a widely used open-source software library for computer vision and machine learning. It offers a broad range of algorithms and tools for image and video processing, including filtering, feature detection, object detection and recognition, segmentation, and camera calibration. With support for various programming languages like C++, Python, Java, and MATLAB, OpenCV is accessible to a wide range of developers and researchers. Its large and active community constantly contributes to its development and provides support for users. OpenCV finds applications in robotics, surveillance, augmented reality, and medical imaging.

OpenCV has been instrumental in various research endeavors across diverse domains [93]. Hoque et al. [94] developed an Autonomous Face Detection System for real-time security intelligence. Similarly, during the COVID-19 pandemic, Das et al. [95] created a Face Mask Detection system using TensorFlow, Keras, and OpenCV, showcasing OpenCV's versatility in addressing contemporary challenges. In education, Mehariya et al. [96] used OpenCV to develop a method for Counting Students in classrooms, integrated with Firebase for efficient classroom allocation. Moreover, Soomro et al. [97] applied OpenCV in a real-time Electronic Voting System, where face recognition ensures secure and transparent electoral processes. These projects highlight OpenCV's adaptability and significance in fostering innovation across security, healthcare, education, and governance domains.

3) Tutorials

Tutorials serve as invaluable resources, especially for beginners, to grasp the fundamentals of face detection and related concepts. El Bruno's tutorial [98] provides a comprehensive walkthrough of face detection using deep neural networks in a Windows environment based on .NET and OpenCV. Utilizing the res10_300x300_ssd_iter_140000_fp16.caffemodel and interfacing with the camera feed via OpenCV, the tutorial demonstrates the real-time processing of video frames. Written in C# within a Windows Forms application, El Bruno's tutorial offers a reliable source for face detection using C#. In Akhtar Jamil's tutorial [99], viewers are guided through facial landmarks detection using Emgu CV 4.4 and C# within a Windows Forms application. The tutorial covers loading pre-trained models into the Emgu CV framework, essential for accurate facial feature identification. Practical implementation includes detecting faces in images or video streams and marking critical facial landmarks like eyes, nose, and mouth. Emgu CV 4.4, a .NET wrapper for OpenCV, ensures seamless integration of computer vision capabilities into C#. By utilizing a Windows Forms application, the tutorial enhances user interaction for intuitive exploration of detected facial landmarks. This comprehensive guide caters to beginners and enthusiasts, offering insights into leveraging Emgu CV for effective facial landmarks detection and visualization.

4) Measurement Methods in Model Evaluation

Accuracy is the basic measure of correctness for machine learning models that calculates the proportion of correct predictions out of all predictions made. However, when dealing with imbalanced datasets, accuracy alone can be misleading. In such cases, other measures such as **Precision**, **Recall**, and **F1-score** should be used. From the literature, we have found a series of measurements that have been used along with the accuracy to evaluate the performance and correctness of models.

- Precision [100] [46] [58] [82]: Examines the accuracy of positive predictions, providing insights into the proportion of correctly identified faces among the total predicted positive cases.
- Average Precision [5], [101] [56]: Measures the area under the precision-recall curve, providing a consolidated evaluation of a model's precision at various recall levels and offering a comprehensive assessment of its performance in ranking and classification tasks

- Mean Average Precision (mAP) [100] [56] [102]: Evaluates the precision-recall curve across multiple thresholds, providing a comprehensive measure of model performance.
- Recall (Sensitivity) [100] [46] [58] [82] [101]: Evaluates the model's ability to capture all relevant instances, focusing on the ratio of correctly identified faces to the total actual positive cases.
- F1 Score [102] [103]: Represents the harmonic mean of precision and recall, offering a balanced metric that considers both false positives and false negatives.
- Receiver Operating Characteristic (ROC) Curve [82] [53] [54]: illustrates the trade-off between sensitivity (true positive rate) and specificity (true negative rate) at various thresholds and Area Under ROC Curve provides a quantitative measure of a model's ability to distinguish between positive and negative instances, summarizing the overall discriminatory performance.
- Intersection over Union (IoU) [5] [101] [104] [54]: Quantifies the overlap between the predicted bounding box and the ground truth bounding box, commonly used in object detection tasks.
- Confusion Matrix: Summarizes the model's performance by presenting the counts of true positives [58], true negatives, false positives [102], and false negatives [102].
- False Positive Rate (FPR) [58] [82] [54] and False Negative Rate (FNR): Express the proportion of incorrect positive and negative predictions, respectively.
- Top-1 accuracy, Top-k accuracy [105]: Top-1 accuracy is the measure used in classification problems, which calculates the percentage of cases where the model's top prediction matches the actual label. Top-k accuracy, on the other hand, allows for more flexibility in the model's predictions. Instead of requiring that the top choice exactly matches the actual label, top-k accuracy counts a prediction as correct if the actual label is among the top-k predictions made by the model.
- Accuracy [104] [68]: The ratio of correctly predicted instances (both true positives and true negatives) to the total number of instances in the dataset. It assesses the overall correctness of the model's predictions.

In this section, the focus is on exploring the applications, resources, and measuring methods related to face detection using machine learning algorithms. The diverse applications of face detection are highlighted. Next, the focus was on an extensive array of datasets, resources, programs, and software available for face detection applications. Furthermore, we explained essential resources like TensorFlow.js, OpenVINO, Face-api.js, and OpenCV, providing an overview of their capabilities and applications, and making it a comprehensive guide for researchers and practitioners. Finally, about the measurement methods employed in model evaluation for face detection. It emphasizes the importance of metrics beyond accuracy. The section equips readers with the knowledge to assess and understand the performance of face detection models effectively.

IV. DIFFERENT MACHINE LEARNING ALGORITHMS FOR FACE DETECTION AND FACE RECOGNITION

This section provides a comprehensive exploration of four key methodologies employed in this domain: Knowledge-Based, Feature-Based, Template Matching, and Appearance-Based methods. From the manipulation of human knowledge to the integration of sophisticated statistical analysis and deep learning, these approaches represent a spectrum of techniques designed to address the intricate challenges posed by varying lighting conditions, facial expressions, and complex scenes. As we examine deeper into each method, we uncover the nuances of their design, their strengths, and the continuous efforts to enhance the accuracy and robustness of these methods.

1) Knowledge-Based methods

Knowledge-based methods offer the advantage of simplicity and computational efficiency. These methods rely on a set of rules designed using human knowledge, such as the distance between facial features like eyes, nose, and mouth, to detect faces. However, the accuracy of these methods depends on the quality of the designed rules, and designing appropriate rules can be challenging. Too many or too detailed rules may reduce accuracy, and these methods may struggle with multiple face detections. Translating human knowledge into well-defined rules is also difficult, and if a face does not meet at least one rule, it may not be detected.

Despite their limitations, knowledge-based methods can provide accurate localization of facial landmarks in applications like facial landmark detection. However, they may struggle with variations in lighting, pose, and expression, and may not be suitable for complex scenes or images with multiple faces. To address these challenges, knowledge-based methods are often combined with other approaches, such as deep learning, to improve accuracy and robustness [17], [19], [20].

2) Feature-Based methods

The feature-based method utilizes a pre-trained classifier to distinguish facial and non-facial regions in an image, relying on structural features extracted from a face. Unlike the Knowledge-Based method, which relies on human-defined rules, this method extracts facial features like eyes, eyebrows, and mouth using edge detectors and statistical models. The presence of a face is then verified using the classifier [17], achieving a reported success rate of 94% even with images containing multiple faces [19], [20]. However, challenges such as illumination, noise, and occlusion can affect its

performance, causing blurring of feature boundaries and shadow interference [17].

Despite these challenges, the feature-based method remains widely used in face recognition systems due to its high accuracy and robustness. Researchers have proposed various feature extraction techniques to enhance its performance, including hybrid methods that combine multiple approaches. For example, integrating the feature-based method with graph-based techniques or pose normalization has shown promising results in handling occlusions and improving recognition rates under varying poses.

While the feature-based method is powerful, its effectiveness relies on the quality of extracted features and classifier robustness. Thus, ongoing research focuses on developing new techniques and algorithms to enhance its performance in challenging conditions.

3) Template Matching

Template matching is a popular technique for face detection that involves the use of predefined or parameterized face templates. The correlation between these templates and input images is used to locate or detect faces. One approach involves dividing a human face into different parts such as the eyes, face delineation, nose, and mouth. By using the edge detection method, a prototype face can be created. However, this approach is not sufficient for accurate face detection in complex situations.

To address this limitation, deformable templates have been proposed to deal with problems such as occlusion and variations in pose and expression. These templates allow the shape of the template to vary to better match the input image. Although template matching is relatively easy to implement, its effectiveness is limited by the quality of the templates used and the degree of variability in the target faces. Despite these challenges, template matching remains a valuable technique for face detection, and researchers continue to explore ways to improve its accuracy and robustness.

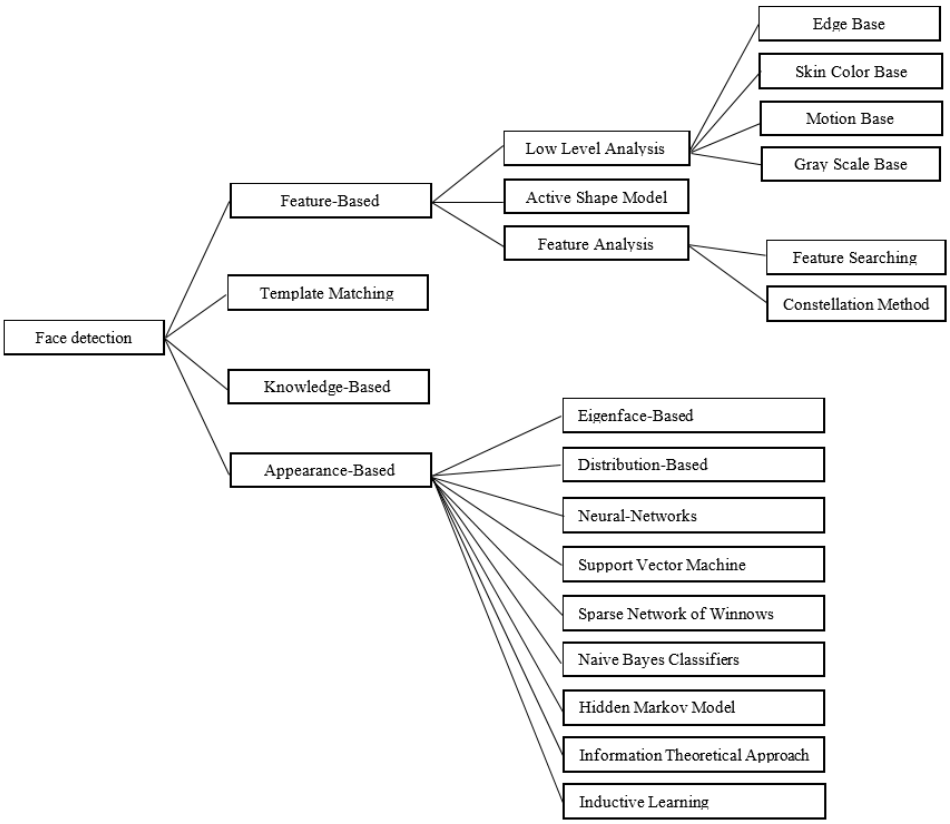

Figure 2 primary categories of face detection methods

4) Appearance-based methods

The appearance-based method uses statistical information in pixel values of facial images to extract features for face recognition [106]. It leverages machine learning and statistical techniques to identify facial characteristics, delivering superior performance. This method integrates various algorithms, including Support Vector Machines, Neural Networks, Eigenface-Based methods, Distribution-Based

methods, Hidden Markov Models, Naive Bayes Classifiers, Information Theoretical Approaches, and Inductive Learning [14],[16],[17]. These sub-methods offer diverse approaches for extracting and analyzing facial features, making the appearance-based method versatile and effective.

Convolutional neural networks (CNN), such as Multi-task Cascaded Convolutional Networks (MTCNN), have significantly advanced tasks like object and face detection and emotion recognition [5]. Rowley et al. [83] developed an early successful neural network-based face detection system in 1998, using a two-stage approach with coarse scanning and refinement stages. This method, evaluated on the MIT-CBCL [107] and CMU PIE [108], [109] datasets became a common paradigm in face detection systems.

Researchers have developed algorithms categorized into two types: two-stage and one-stage object detection algorithms [110]. Two-stage detectors, like Faster RCNN, RCNN, Fast RCNN, Mask R-CNN, SPPNet, Pyramid Networks, and G-RCNN, first identify regions of interest and then classify them. One-stage detectors, such as RetinaNet, YOLO, YOLOR, SSD, YOLOv3, and YOLOv4, directly predict bounding boxes without a separate region proposal step.

One-stage algorithms like YOLO [16] and SSD [14], use dense grids of bounding boxes with various aspect ratios and scales, making them computationally efficient and faster. However, they are less accurate in detecting small or complex-shaped objects and often suffer from false positives.

Two-stage algorithms, such as Faster R-CNN [15] and Mask R-CNN [111], use a region proposal network (RPN) to generate candidate object regions, which are then classified and refined by a separate network. These are more accurate and robust in detecting small and complex-shaped objects but are computationally more expensive and slower. Two-stage methods typically achieve the highest detection accuracy but are slower, while one-stage detectors are faster but struggle with irregularly shaped objects or small groups of objects [110], [112].

In summary, the exploration of face detection and recognition methodologies reveals a landscape rich with diversity and innovation. Knowledge-based methods, although simple, showcase computational efficiency, particularly in tasks such as facial landmark detection. Feature-based approaches, overcoming the limitations of rule-based methods, provide high accuracy in recognizing faces even in images with multiple subjects. Template Matching, utilizing predefined templates, remains a valuable technique, while Appearance-Based methods, integrating statistical information, emerge as powerful tools with superior performance.

V. FACE DETECTION FEATURE EXTRACTION AND FACE RECOGNITION ALGORITHMS

Face detection and recognition algorithms can be broadly categorized into knowledge-based, feature-based, template-matching, and appearance-based approaches. Knowledge-based methods rely on predefined rules and expert knowledge of facial anatomy to identify features. Feature-based algorithms extract specific facial components for representation and analysis. Template matching involves comparing a facial template with the target image for similarity. Appearance-based approaches consider overall facial appearance, utilizing statistical models or machine learning to capture holistic representations. These categories encompass diverse techniques, each with its strengths, ranging from rule-based systems to advanced machine learning methods, collectively contributing to the evolution of face detection and recognition technologies for a variety of applications. Here, we present literature relevant to each category.

1) knowledge-Based

Zhang et al. [78] developed a knowledge-based eye detection algorithm for human faces, combining image processing techniques with knowledge-based methods. The algorithm comprises two stages: first, locating the face and eye regions using histogram thresholding and smoothing procedures. Then, the eye region coordinates are identified, and the eye region is extracted. In the second stage, edge enhancement via the Laplacian operator and a knowledge-guided eye contour searching method are employed. This method leverages knowledge of the eye's shape and location to improve detection accuracy. Evaluation on the Yale Face Database using 320*243 face images and 200*102 eye region images demonstrated effective face and eye region localization, with successful eye contour extraction. The approach offers a balance of accuracy and computational efficiency, making it suitable for eye contour searching applications.

Yang G et al. [113] proposed a hierarchical knowledge-based approach for detecting human faces in complex backgrounds. The algorithm includes three levels of rules based on mosaic images of varying resolutions. Facial features are identified using an improved edge detection method at lower levels. At the highest level, a window scans the input image to identify potential face candidates, applying rules at each location. Higher-level rules define general facial appearance, while lower-level rules define specific features. The algorithm proceeds by dividing the image into small regions and processing each with an edge detector to detect edge pixels. These pixels are grouped into connected segments, which are then clustered into larger ones. These segments are analyzed to determine if they represent facial features, and if so, the algorithm combines them to form a complete face. Testing on images with complex backgrounds yielded promising results.

Kotropoulos et al. [79] introduced a rule-based method for frontal face detection using heuristic rules derived from domain knowledge. The algorithm begins with pre-processing steps like noise reduction and image segmentation, followed by applying rules to identify potential face regions. By averaging pixel intensities, the algorithm obtains horizontal and vertical profiles, detecting local minima to locate facial features such as hair, eyebrows, eyes, mouth, and chin. It

employs nostril, nose, eyebrow, eye, and mouth detection rules to validate these features. Using edge detection and the Hough transform, the algorithm identifies line segments corresponding to facial features, generating candidate facial feature combinations that meet geometric constraints. These candidates are assessed against heuristic rules to select potential face regions, verified by template-based classifiers. Evaluated on 90 frontal face images, the algorithm achieved a 97.77% detection rate with a 2.2% false positive rate. However, it was limited to detecting single faces against a uniform background from the European ACTS M2VTS dataset. Despite this, the method demonstrated robustness to occlusion, illumination, and expression variations. Primarily designed for frontal face detection, the method may not effectively detect non-frontal or profile views. Nevertheless, it showed high accuracy across different lighting conditions and outperformed existing methods at the time of publication.

2) Feature-Based

Chan et al. [114] present a face detection system using feature-based chrominance color information from a single indoor face image with a non-uniform background. The method employs the Modified Golden Ratio (MGR), Adapted Chain Code (ACC), and eye detection for accuracy. The algorithm starts with skin color segmentation to identify potential facial regions, then uses ACC to estimate the face boundary and predict eye candidates. To reduce computational complexity, estimated eye candidates and face boundary images are downsampled to 128x128 pixels using wavelet transform. The algorithm approximates eye positions, performs eye detection, and extracts key facial features such as eyes, brows, mouth, and nose based on the detected eyes and refined boundary. The method focuses on estimating eye candidates and analyzing chrominance in skin color segmentation. Chrominance, representing color purity and saturation, consists of Cr (red difference chroma) and Cb (blue difference chroma), capturing color information independent of brightness. Each pixel in a color image has YCbCr values containing luminance (Y) and chrominance (Cr and Cb). The method empirically determines Cr and Cb ranges from 16 skin regions for segmentation. For eye candidate estimation, an equation reduces the computational complexity of the eye map used by Hsu et al. [115]. To avoid misclassifying noise as eyes, the method introduces multilevel thresholding with 3-level priority. Evaluated 80 face images, including faces with and without spectacles and headscarves,

Table 2 Summary of the literature, sorted according to the Face detection– Knowledge-Based

Year	Authors	Detections	Key methods	Dataset	Accuracy and Advantages	Drawback
2000	Zhang et al. [78]	Locating both the face and eye regions	• Histogram thresholding • Histogram smoothing • Automatic thresholding • Laplacian operator • Knowledge-guided eye contour searching	Yale Face Database	Low computational cost	• Accurate for frontal-view face images with a plain background • Not suitable for complex images and multi faces images.
1997	Kotropoulos et al. [79]	Detect frontal faces	• Heuristic rules • Noise reduction • Image segmentation • Horizontal and vertical profiles • Eyebrow, eye, nostril, nose, and mouth detection rules • Edge detection • Hough transform • Template-based classifiers	90 frontal face images from the European ACTS M2VTS dataset	The overall detection rate of 97.77% with a false positive rate of 2.2%	• Limited to detecting only one face in a uniform background • Not suitable for detecting faces in profile or other non-frontal views
1994	Yang G et al. [113]	Locate human faces	• Improved edge detection method, • Edge detector, • Rule-based connected edge segments • Clustering algorithm	set of images with complex backgrounds	Accurate for complex background	• Accurate only for black-and-white pictures

the algorithm achieved 91.25% accuracy for detecting near-frontal faces, indicating its potential for practical applications.

Viola and Jones et al. [116] introduced the Viola-Jones algorithm for object detection using a boosted cascade of simple Haar-like features. The algorithm involves training a cascade of weak classifiers to efficiently classify image regions as containing or not containing an object of interest, using the AdaBoost machine learning algorithm to iteratively select the most informative features. Haar-like features, which are rectangular regions with varying intensities (e.g., edges or contrasts), represent the object of interest. The algorithm applies a sliding window technique, moving a fixed-size rectangular window across the image and computing Haar-like features at each location. AdaBoost evaluates these features, selecting the most informative ones to build a strong classifier for accurate object detection. The algorithm uses a cascade of classifiers, where each stage comprises multiple weak classifiers. Each stage quickly rejects regions that do not contain the object, reducing false positives and improving efficiency. The algorithm was trained and tested on a manually labeled dataset of face and non-face images, with 4,000 positive (face) and 8,000 negative (non-face) images in the training set, and 5,000 positive and 10,000 negative images in the testing set. The Viola-Jones algorithm employs Haar-like features and AdaBoost to train a cascade of weak classifiers for efficient object detection. By selecting informative features and using a classifier cascade, the algorithm achieves high accuracy and fast detection times. It demonstrated high accuracy on both training and testing datasets, making it a powerful tool for object detection widely used in various applications.

Fasel et al. [117] present a novel real-time eye detection method combining generative and discriminative models using the AdaBoost algorithm. They train a cascade of weak classifiers with features selected by AdaBoost, supplementing the training data with synthetic examples from a generative model, addressing limited training data, and enhancing discriminative model performance. A new feature representation for eye detection incorporates the spatial relationship between the eyes and other facial features, forming a feature vector used as input for the weak classifiers in the boosting cascade. This enables accurate, efficient eye tracking in real-time video streams. The system includes two types of eye detectors: one for general illumination and background conditions, and another for higher accuracy by leveraging contextual information. The algorithm first identifies potential face regions using a likelihood-ratio model trained on web images from Compaq Research Laboratories, detecting faces as small as 24x24 pixels. Efficiency is improved by using a sequence of smaller classifiers to evaluate wavelets and make early decisions, reducing unnecessary processing. Inspired by Viola and Jones [116], the system scales larger image patches to 24x24 pixels and applies the likelihood ratio model. It scans the entire image plane for patches with high likelihood ratios, identifying probable eye locations, and forwards these patches to a blink detection module for further analysis. Treating each frame independently, the system works for both static images and videos, encoding eye location and behavior for multiple faces appearing and disappearing randomly. By utilizing both discriminative and generative models, the algorithm achieves high accuracy and robustness in real-time eye coding applications. The approach is demonstrated to be effective on a dataset of video sequences, outperforming other state-of-the-art methods.

Vukadinovic et al. [80] introduced an automated technique to detect 20 facial feature points in expressionless face images using boosted classifiers based on Gabor features. Their approach improves on the original Viola-Jones face detector by Fasel et al. [117], offering a fast and reliable face detection process. The detected face region is divided into 20 regions of interest, where feature points are predicted using "GentleBoost templates" derived from gray-level intensities and Gabor wavelet features. The method employs an enhanced Viola-Jones face detection algorithm using GentleBoost instead of AdaBoost to detect the face region initially. It then refines feature selection with new filters, training on 5,000 faces and millions of non-face patches, achieving a 100% detection rate on 422 images. The approach automates the detection of the medial point of the mouth and irises to determine regions of interest, dividing the face into upper (eyes) and lower (mouth) sections. It uses horizontal and vertical histogram analysis to locate the irises and calculates the angle between the irises and the horizontal plane, rotating the image if needed. The algorithm identifies the medial point of the mouth within a defined region based on the distance between the irises (ED). It positions the region top at 0.85×ED and height at 0.65×ED, determining the vertical position by analyzing the mouth region's vertical histogram. This method achieves a 100% detection rate for both the irises and the medial point of the mouth. The algorithm demonstrated a 93% average recognition rate on the Cohn-Kanade database for facial expression analysis. It was trained on 5,000 faces and numerous non-face patches from about 8,000 web images and evaluated on 422 Cohn-Kanade images, achieving a 100% detection rate.

Cox et al. [118] presented a feature-based face recognition approach using mixture-distance. They modeled the training data as a combination of normal densities, projecting local second-order statistics onto it. They used 35 manually identified facial features from each face to create a 30-dimensional feature vector. Testing on a database of 685 individuals, they achieved a 95% recognition rate. Duplicate images from 95 individuals were used as queries to measure performance. The approach involved mixture-distance functions to measure distance, encountering two model selection challenges: determining the number of mixture elements and selecting between first and second-order statistics for each Gaussian component. They addressed these with a flat prior approach, yielding results comparable to the

best individual model. The experimental results confirmed the method's effectiveness, achieving the highest recognition rate among feature-based systems for a database of this size.

Manjunath et al. [119] proposed a face recognition approach divided into three stages: feature detection and localization, graph representation of the face, and recognition. Feature detection is based on the end-habilitation property model, using local scale interactions between oriented features. It involves extracting oriented feature information at different scales using the Gabor wavelet transform and then interacting these features across scales to achieve the end-habilitation effect. The method can be executed in parallel, making it suitable for real-time applications. It employs topological graphs to represent feature relationships and uses a deterministic graph marching schema to recognize faces from a database. This approach represents an early use of graph-based methods for face recognition, which has since become popular in the field.

3) Template Matching

Najat et al. [120] proposed a human face detection method in crowded images using template-matching techniques. The method starts by converting target and template images to grayscale. The template image is divided into a grid, and each cell is matched against the corresponding target image cell using Two-Dimensional Normalized Cross-Correlation (2D-NCC) to measure similarity and find maximum correlation. If the correlation between a template grid cell and the corresponding target cell exceeds a threshold, it is considered a match, and the location is recorded. The process involves comparing each template cell to face and non-face templates, repeating for all cells to identify matches, which are then combined for the final detection results. The algorithm can detect faces in low-resolution images with varying lighting conditions and expressions but may struggle with occlusion and rotation. This straightforward approach is promising for real-time applications.

Bose et al. [121] proposed a technique for detecting facial parts using the normalized cross-correlation (NCC) template matching method. They created a template database with images of different facial features (nose, eyes, mouth, and face), rotated at various angles to handle pose variations and improve the matching likelihood. To detect facial parts, the NCC algorithm calculates a correlation map between the input and template images at each position, identifying the best match by the highest correlation value. The method uses NCC values to detect the face and its parts. If the NCC value exceeds a threshold, the corresponding facial part is considered detected. The authors claimed their method can accurately detect facial parts in real-time with high accuracy but noted potential limitations due to facial expressions, occlusion, lighting variations, and rotation.

Chai et al. [81] presented a face detection method using a skin color model based on the r, g color space. The model was created using the equations: $r = R / (R + G + B)$ and $g = G / (R + G + B)$, where R, G, and B are the red, green, and blue color components of each pixel. Pixels representing skin color were selected if their color values, $v = (r, g)$, were above a specific threshold. The mean and covariance matrix of these selected pixels were used to create a Gaussian distribution model with the probability density function. After identifying the face region using the skin color model, the image was converted to grayscale, and a grayscale closing operation was applied. Morphological operations such as dilation and erosion were performed to refine the face region by filling gaps and removing small holes, smoothing the edges for easier feature extraction. For iris detection, the authors applied image processing techniques including illumination normalization and light spot deletion to enhance iris features. They reported a 93.06% accuracy rate for iris detection. The mouth region was detected using the location of the irises, applying a color space method, and the SUSAN [122] corner detector to refine the mouth corners and exact location, achieving a 95.83% accuracy rate. After feature extraction and refinement, template matching was used to classify the extracted features and recognize faces. Experiments on the AR face database, comprising 72 color images of 12 male and 12 female subjects with variations in pose, head orientation, facial expression, and other factors, yielded a recognition rate of 86.11%.

Bose et al. [121] proposed a technique for detecting facial parts using the normalized cross-correlation (NCC) template matching method. They created a template database with images of different facial features (nose, eyes, mouth, and face), rotated at various angles to handle pose variations and improve the matching likelihood. To detect facial parts, the NCC algorithm calculates a correlation map between the input and template images at each position, identifying the best match by the highest correlation value. The method uses NCC values to detect the face and its parts. If the NCC value exceeds a threshold, the corresponding facial part is considered detected. The authors claimed their method can accurately detect facial parts in real-time with high accuracy but noted potential limitations due to facial expressions, occlusion, lighting variations, and rotation.

Chai et al. [81] presented a face detection method using a skin color model based on the r, g color space. The model was created using the equations: $r = R / (R + G + B)$ and $g = G / (R + G + B)$, where R, G, and B are the red, green, and blue color components of each pixel. Pixels representing skin color were selected if their color values, $v = (r, g)$, were above a specific threshold. The mean and covariance matrix of these selected pixels were used to create a Gaussian distribution model with the probability density function. After identifying the face region using the skin color model, the image was converted to grayscale, and a grayscale closing operation was applied. Morphological operations such as dilation and erosion were performed to refine the face region by filling gaps and removing small holes, smoothing the edges for easier feature

Table 3 Summary of the literature, sorted according to the Face detection– Feature-Based

Year	Authors	Detections	Key methods	Dataset	Accuracy and Advantages	Drawbacks
2005	Fasel et al. [117]	location of a person's eyes in a video stream, real-time eye coding	• AdaBoost algorithm • sequence of smaller classifiers feature vector • likelihood-ratio model	dataset of web images provided by Compaq Research Laboratories	• address the problem of insufficient training data • simultaneous encoding of eye position and behavior in multiple faces • achieve high accuracy and robustness in real-time eye-coding applications	• Detectors that provide high accuracy within the face may exhibit high false-alarm rates outside the face
2005	Vukadinovic et al. [80]	detect 20 facial feature points in expressionless face images	• Gabor feature-based boosted classifiers • GentleBoost templates • individual feature patch templates • vertical and horizontal histogram • horizontal and vertical thresholded edges	Cohn-Kanade database	• fast and robust face detection algorithm • achieved a detection rate of 100% for the irises and the medial point of the mouth • average recognition rate of 93%	• Limited Training Data Tested on Grayscale Images • Limited Variation in Inter-ocular Distance and Sensitivity to Inter-ocular Distance
2004	Chan et al. [114]	one face in an indoor environment with a non-uniform background	• Adapted Chain Code (ACC) • Modified Golden Ratio (MGR) • chrominance color information • skin color segmentation • face boundary estimation • eyes candidate estimation	80 face images consisting of faces with and without spectacles, wearing a headscarf and without wearing a headscarf	achieved 91.25% accuracy in detecting a near-frontal face	Accurate for one face in an indoor environment.

			• wavelet transform • Multilevel thresholding with 3-level priority			
2001	Viola and Jones et al. [116]	object detection, face detection	• Haar-like features • AdaBoost algorithm • machine learning algorithm • sliding window technique • classifiers	manually labeled face and non-face images	achieving high accuracy and fast detection times	• Deeper classifiers in the cascade exhibit higher false positive rates.
1996	Cox et al. [118]	face recognition	• mixture-distance • 30-dimensional feature vector • mixture-distance functions • first and second-order statistics	a database of 685 individuals	• recognition rate of 95%	• Challenges in Gaussian mixture model selection impact recognizer performance due to the difficulty in determining the optimal number of mixtures. • Challenge in optimal model selection
1992	Manjunath et al. [119]	face recognition	• Gabor wavelet • topological graphs • simple deterministic graph marching schema	Face images of 86 persons with two or four images per person, taken with different facial expressions	• approach can be implemented in parallel. • Ability to use real-time face recognition applications. • The recognition accuracy 86%	• Not suitable for complex images and multi faces • Sensitivity to variations in lighting conditions, facial expressions, pose, scale, and occlusions

extraction. For iris detection, the authors applied image processing techniques including illumination normalization and light spot deletion to enhance iris features. They reported a 93.06% accuracy rate for iris detection. The mouth region was detected using the location of the irises, applying a color space method, and the SUSAN [122] corner detector to refine the mouth corners and exact location, achieving a 95.83% accuracy rate. After feature extraction and refinement, template matching was used to classify the extracted features and recognize faces. Experiments on the AR face database, comprising 72 color images of 12 male and 12 female subjects with variations in pose, head orientation, facial expression, and other factors, yielded a recognition rate of 86.11%.

4) *Appearance-Based*

Yang et al. [5] introduced a method for detecting heterogeneous facial features using a Multi-Task Cascaded Convolutional Neural Network (MTCNN), consisting of three sub-networks: Proposal Network (P-Net), Refine Network (R-Net), and Output Network (O-Net). The first step scales the image to various scales to construct an image pyramid. Candidate face windows are obtained using the P-Net, a fully convolutional attention network. Highly overlapping windows are adjusted by a bounding box regression vector and combined using non-maximum suppression (NMS). In the second step, all candidate windows are processed by the R-Net, which refines initial candidate windows and eliminates most non-face candidates using border regression and facial keypoint localization. NMS is applied again to combine intersecting windows. Finally, the O-Net recognizes the facial area, outputting coordinates for the upper left and lower right corners of the face and locations of five facial landmarks (two eyes, nose, and mouth corners). The MTCNN algorithm effectively detects faces with significant variations in pose, scale, and occlusion by leveraging the three networks. It achieved an average accuracy of 96.95% on the CASIA NIR-VIS 2.0, CUFS, and CUFSF datasets, outperforming traditional face detection algorithms and achieving state-of-the-art performance on various benchmarks.

Rowley et al. [83] introduced a face detection system using neural networks, tested with 130 images, achieving a detection rate between 78.9% and 90.5%. The approach has two main stages: coarse scanning and refinement. In the coarse scanning stage, neural network-based filters scan the image at various scales and locations to generate candidate face regions. In the refinement stage, a detailed neural network evaluates these candidate regions to confirm face presence, merging detections and removing overlaps to reduce errors. The method was evaluated on the MIT-CBCL and CMU PIE datasets, comparing its performance with traditional feature-based and other neural network-based methods. The evaluation metric was the detection rate at a false positive rate of 10%. Results showed superior performance, with detection rates of 93.5% on the MIT-CBCL dataset and 96.7% on the CMU PIE dataset, outperforming state-of-the-art methods at the time.

Almabdy et al. [68] conducted a study on face recognition using three CNN-based methods, evaluated on databases including GTAV face, LFW, YouTube face, FEI faces, ORL, F_LFW, and Georgia Tech face. In the first method, they used the pre-trained AlexNet model combined with an SVM classifier for classification. In the second method, they employed a pre-trained ResNet-50 model, also using an SVM for classification. In the third method, they modified AlexNet by removing the last three layers and adding a new fully connected layer for fine-tuning. The accuracy of these models ranged between 94% and 100\%, with the third method achieving 100% accuracy on the GTAV face dataset and outperforming the other methods in accuracy and computational efficiency. Their methods achieved comparable or superior results to state-of-the-art methods on most datasets. They found that increasing the number of training samples improved accuracy. LDA-based dimensionality reduction improved model accuracy compared to PCA, and higher-resolution images generally led to better performance.

Sun et al. [69] proposed a face verification method in unconstrained conditions using a hybrid model combining convolutional networks (ConvNets) with Restricted Boltzmann Machines (RBMs). They used multiple ConvNets to capture high-level and global facial characteristics. The framework integrates deep belief networks, ConvNets, and deep Boltzmann machines in three stages: pre-training, fine-tuning, and feature extraction. During pre-training, deep Boltzmann machines and deep belief networks were trained layer-by-layer. In fine-tuning, the pre-trained models were refined using labeled face verification data. Finally, in feature extraction, the learned representations were used for face verification. The method was evaluated on the LFW [67] and PubFig [123] datasets, using the receiver operating characteristic curve (ROC) and verification rate at a false positive rate of 0.1% as metrics. The hybrid approach outperformed state-of-the-art methods, achieving a verification rate of 99.52% on LFW and 91.54% on PubFig. Experiments showed that each component of the approach contributed to its overall performance.

Schroff et al. [77] proposed FaceNet, a face recognition system that performs face verification, recognition, and clustering using a deep convolutional network. The system learns a 128-D Euclidean embedding per image, representing key facial features with L2 normalization. They employed triplet loss to train the network, ensuring faces with similar features are close in the embedding space, while dissimilar faces are farther apart. Triplet loss minimizes the distance between an anchor image and a positive image (same person) and maximizes the distance to a negative image (different person). A triplet dataset of anchor, positive, and negative images was used for training, with stochastic gradient descent

Table 4 Summary of the literature, sorted according to the Face detection– Template Matching

Year	Authors	Detections	Key methods	Dataset	Accuracy and Advantages	Drawbacks
2022	Najat et al. [120]	Face recognition	• Two-Dimensional Normalized Cross-Correlation (2D-NCC) technique	not mentioned	• detect faces in low-resolution images with varying lighting conditions and facial expressions	• may face limitations when dealing with occlusion and rotation • certain image processing techniques may still require
2020	Bose et al. [121]	detecting facial parts of the human face	• normalized cross-correlation template matching • Template database of facial parts images • correlation maps	more than 100 images were used to verify the algorithm	• detect facial parts in real-time and with high accuracy	• May suffer from limitations such as variations in lighting and facial expressions, occlusion, and rotation
2009	Chai et al. [81]	face detection, iris detection, mouth detection	• skin color model • Gaussian distribution model • probability density function • grayscale closing method • image processing techniques like illumination normalization, • light spot deletion • SUSAN corner detector • color space method • template matching	AR face database	• accuracy rate of 93.06% for iris detection • accuracy rate of 95.83% for mouth detection • accuracy rate of 86.11% for template matching	• Accuracy limited to head-shoulder images with plain backgrounds • Lower intensity differences in iris and facial expressions like frowning can cause iris detection failures. • Failure of the projection method in images with thick facial hair or 'mouth-opened' conditions.

minimizing the triplet loss. In testing, the network produced embeddings compared using a distance metric to verify identity. The model was evaluated on YouTube Faces DB and Labeled Faces in the Wild (LFW), achieving 99.63% accuracy on LFW and 95.12% on YouTube Faces DB, outperforming existing methods and demonstrating the effectiveness of deep learning for face recognition.

Jiang H et al. [54] proposed using the Faster R-CNN framework for face detection, combining a Region-based Convolutional Neural Network (R-CNN) and a Region Proposal Network (RPN). They trained the model using the WIDER face dataset, containing 12,880 images and 159,424 faces. VGG16, pre-trained on ImageNet, was used as the face detection model, and its performance was evaluated on FDDB and IJB-A benchmark datasets. For optimization, they resized input images to 600 pixels on the longer side while maintaining the aspect ratio. They adjusted anchor scales to [16, 32, 64, 128, 256] pixels and reduced the number of proposed regions in the RPN from 3,000 to 600 per image for improved speed and accuracy. A specialized loss function enhanced face detection accuracy. The RPN was the most effective module in the Faster R-CNN model. Their approach achieved state-of-the-art performance on the WIDER face detection dataset, with a mean Average Precision (mAP) score of 94.9%, making the Faster R-CNN framework a popular choice for face detection due to its high accuracy and efficiency.

Sun X et al. [53] proposed an enhanced Faster RCNN approach for face detection using deep learning techniques. They improved the Faster RCNN framework with several strategies: feature concatenation (combining features from different layers), hard negative mining (identifying and adding false negatives to the training set), and multi-scale training (assigning random scales to images for improved scale invariance). The CNN network was trained on the WIDER FACE dataset and tested on the same dataset to generate hard negatives, which were then incorporated into the training process. The model was fine-tuned using the FDDB. They also converted the rectangular detection bounding boxes into ellipses to better match the human face shape. The proposed algorithm demonstrated state-of-the-art results and top rankings among published methods, attributed to the combination of multi-scale training, feature concatenation, model pre-training, hard negative mining, and proper calibration of key parameters.

Guo G et al. [82] proposed a fast face detection algorithm using discriminative complete features (DCFs) from a deep convolutional neural network. The method includes two main components: sparse discriminative features and a nonlinear mapping function. The nonlinear mapping function uses convolutional and max-pooling layers in the CNN to project raw data onto the Euclidean space. Sparseness is achieved using rectified linear units (ReLU), generating a sparse feature space. Once the sparse feature space is created, a nearest neighbor interpolation technique resizes the multi-scale features, enhancing feature extraction at various scales to detect faces of different sizes. The authors propose a fast method to obtain features based on complete feature maps for each window in an input image, extracting desired features before the fully connected layer. The proposed model was trained on the CAS-PEAL-R1 [124] and VOC2012 [125] datasets, and evaluated on the AFW AFW [126] and FDDB [51] datasets. The DCFs-based face detection approach was compared to several state-of-the-art methods, including RCNN, fast RCNN, faster RCNN, DeepIR, and YOLO. A sliding window approach was also used to detect small faces. The proposed method demonstrated significant improvements in face detection performance on benchmark datasets, achieving high accuracy and fast detection speed, making it suitable for real-time applications like surveillance systems and facial recognition technology.

Zhang et al. [101] introduced AInnoFace, a face detection method based on the RetinaNet [127] approach, incorporating modern techniques like Selective Two-step Regression (STR), Intersection over Union (IoU) loss function, Two-step Classification (STC), data augmentation, and max-out operation to reduce false positives. They utilized a multi-scale testing strategy to enhance detection accuracy. AInnoFace employs ResNet-152 with a 6-level feature pyramid structure as the backbone network, enabling feature extraction at multiple scales. It achieves state-of-the-art performance on the WIDER FACE dataset, attributed to the combination of modern techniques and the feature pyramid structure of ResNet-152, which collectively contribute to its high accuracy and fast speed. Experiments on the WIDER FACE dataset compared AInnoFace to YOLO, Faster R-CNN, and SSD detectors using average precision (AP) and average recall (AR) at different IoU thresholds. AInnoFace achieved an AP of 95.8% and an AR of 95.6% at an IoU threshold of 0.5, significantly outperforming other detectors. AInnoFace demonstrated high detection accuracy across different scales of faces, crucial for real-world applications. Its multi-scale testing strategy further contributed to its performance. Overall, AInnoFace exhibits state-of-the-art performance on the challenging WIDER FACE dataset.

Wang J et al. [56] proposed the Face Attention Network (FAN) for detecting partially occluded faces using a deep convolutional neural network with a novel face attention module. This module filters out irrelevant regions to focus on facial features. They introduced an anchor-level attention mechanism to enhance facial parts and improve detection accuracy. The method uses a feature pyramid network to handle faces of different scales, similar to RetinaNet. The FAN method consists of five detector layers, each linked to a specific scale anchor with aspect ratios of 1 and 1.5, covering areas from 162 to 4,062 on pyramid levels. Data augmentation techniques, such as random cropping, generated a large

Table 5 Summary of the literature, sorted according to the Face detection– Appearance-Based

Year	Authors	Detections	Key methods	Dataset	Accuracy and Advantages
2023	Sun et al. [69]	face verification	• hybrid convolutional network (ConvNet) • Restricted Boltzmann Machine (RBM) • multiple groups of ConvNets • Deep Boltzmann machines • Deep belief networks • convolutional neural networks	• LFW • PubFig face verification datasets	• achieve strong characterization of face similarities from different features • verification rate of 99.52% on LFW and 91.54% on PubFig, which outperformed the state-of-the-art methods
2022	Yang et al. [5]	upper left corner coordinates and lower right corner coordinates of the face area and positions of the five feature points, including two eyes, a nose, and two corners of the mouth	• MTCNN • P-Net • R-Net • O-Net • image pyramid • estimated bounding box regression vector • non-maximum suppression (NMS)	• CUFS • CUFSF • CASIA NIR-VIS 2.0	• capable of detecting faces with large variations in pose, scale, and occlusion • achieved an average accuracy of 96.95 %
2022	Cao et al. [103]	detecting face mask-wearing	• YoloMask model • convolutional neural network (CNN) • YOLOX model • CSP-DarkNet • feature pyramid network (FPN) • path aggregation network (PAN) • "Decoupled Head" technique • alpha-CIoU loss function	• Diverse Masked Faces	• authors introduce a new dataset called Diverse Masked Faces • higher detection performance

			• data augmentation techniques such as Mosaic and MixUp • multi-positives trick • OTA • dynamic top-k strategy		
2021	Sanchez et al. [70]	face recognition	• YOLO-Face • image processing techniques • FaceNet+SVM • FaceNet+KNN • FaceNet+RF	• LFW dataset	• FaceNet+SVM model achieves an accuracy of 99.7% • FaceNet+KNN model achieves an accuracy of 99.5% • FaceNet+RF model achieves an accuracy of 85.1% • recognition accuracy of 99.1% and operates in 49 ms
2021	Ali-Gombe et al. [102]	face detection	• 10m-YOLO model • 5m-YOLO model • 2m-YOLO model	• WIDER face • FDDB	• three models were significantly smaller than the original 33m-YOLO model • reduce the model's size • suitable for deployment in a resource-limited environment
2019	Almabdy et al. [68]	face recognition	• Deep Convolutional Neural Network • pre-trained AlexNet + SVM • pre-trained ResNet-50 + SVM • modified AlexNet	• GTAV face • YouTube face • labeled faces in the wild (LFW) • ORL • Frontalized labeled faces in the wild (F_LFW) • Georgia	• 100% accuracy achieved on the GTAV face dataset with modified AlexNet. • modified AlexNet outperformed the other two methods in terms of accuracy and computational efficiency • higher resolution images generally led to better performance

				Tech face • FEI faces	
2019	Zhang et al. [101]	face detection	• AInnoFace detector which is based on the RetinaNet approach • Intersection over Union (IoU) loss function • Two-step Classification (STC) • Selective Two-step Regression (STR) • data augmentation • max-out operation • multi-scale testing strategy • ResNet-152 with a 6-level feature pyramid structure	• WIDER face	• high-performance face detector • achieved high detection accuracy across different scales of faces • achieves state-of-the-art average precision performance results with an AP of 95.8% and an AR of 95.6% • efficiency in detecting faces in complex scenes • high accuracy and fast speed
2019	Zeng et al. [128]	face detection	• cascade face detector • CNN • multi-task learning • network acceleration techniques • multi-scale face proposals • bounding box and facial landmark regression • NMS • multi-layer merging • knowledge distilling • down-sampling • batch normalization (BN) • data augmentations • online and offline hard sample mining • novel multi-layer merging technique	• FDDB	• fast and accurate multi-scale face detection • focus on computational efficiency and accuracy performance • achieving comparable results with state-of-the-art methods at a speed of 165 frames per second on Titan GPU
2018	Sun X et al. [53]	face detection	• improved Faster RCNN	• WIDER face	• achieved state-of-the-art results and ranked the best among all the published

			• feature concatenation • hard negative mining • multi-scale training • converted the detection bounding boxes into ellipses	• FDDB	methods
2018	Guo G et al. [82]	face detection	• discriminative complete features (DCFs) • Deep convolutional network • nonlinear mapping function • sparse discriminative features • Euclidean space • sparse feature space • nearest neighbor interpolation method • sliding window approach	• CAS-PEAL-R1 • VOC2012 • FDDB • AFW	• better feature extraction at different scales • detecting faces of varying sizes • detect small-sized faces • achieves high accuracy and fast detection speed • suitable for real-time applications
2018	Garg et al. [104]	face detection	• YOLO • NMS technique • gradient descent optimizer algorithm	• FDDB	• The proposed approach achieved a high accuracy of 92.2% while maintaining real-time performance
2017	Jiang H et al. [54]	face detection	• Faster R-CNN • Region Proposal Network (RPN) • Region-based Convolutional Neural Network (R-CNN) • pre-trained ImageNet model- VGG16	• WIDER face • FDDB • IJB-A	• mean Average Precision (mAP) score of 94.9% • become a popular choice for face detection due to its high accuracy and efficiency
2017	Wang J et al. [56]	detecting partially occluded faces	• Face Attention Network (FAN) • Deep convolutional neural network with a novel face attention module • new anchor-level attention mechanism • feature pyramid network • data augmentation techniques	• WIDER face • MAFA	• achieved an average precision (AP) of 94.6% (easy) and 88.5% (hard) on the WIDER FACE dataset • achieved an accuracy of 88.3 % on the MAFA dataset • effective solution for detecting partially occluded faces in images

2015	Schroff et al. [77]	face verification, recognition, and clustering	• deep convolutional network called FaceNet • Euclidean embedding • 128-D embedding • L2 normalization • triplet-based loss function • Large Margin Nearest Neighbor (LMNN)	• Labeled Faces in the Wild (LFW) • YouTube Faces DB	• high accuracy of 99.63% for the LFW dataset and 95.12% for the YouTube Faces DB
2015	Ranjan et al. [58]	face detection	• DP2MFD algorithm • Deformable Part Models (DPM) • Deep convolutional neural network • deep pyramidal features • a seven-level normalized deep feature pyramid • sliding window approach • SVM • z-score normalization • 10-fold cross-validation approach • non-maximum suppression and bounding box regression	• AFW • FDDB • MALF • IJB-A	• designed to detect faces of various sizes and poses in unconstrained conditions • achieved new state-of-the-art detection performances • able to detect profile faces as well as different size faces in images with a cluttered background
1998	Rowley et al. [83]	face detection	• neural network-based filters	• MIT-CBCL • CMU PIE	• achieved detection rate between 78.9% and 90.5% for face detection • one of the earliest successful approaches using deep learning techniques in 1998 • The coarse scanning stage and a refinement stage, have become a common paradigm in many faces detection systems

number of occluded faces for training. The authors trained and evaluated FAN on datasets including WIDER FACE and MAFA [129]. FAN achieved an average precision (AP) of 94.6% (easy) and 88.5% (hard) on WIDER FACE, and 88.3% accuracy on MAFA, demonstrating its effectiveness. The FAN method outperformed YOLO, Faster R-CNN, and SSD in accuracy and robustness to occlusion. This research highlights the potential of attention-based methods for face detection and provides an effective solution for detecting partially occluded faces.

Cao et al. [103] propose a novel method for detecting face mask-wearing using the YOLOX [130] model, a state-of-the-art CNN for object detection. They introduce the Diverse Masked Faces dataset, containing five mask-wearing classes: normal, irregular, chin, nose-only, and spoofing. The YoloMask architecture comprises four main components: input, neck, backbone, and predictor. The input is a fixed-size 416x416 RGB image. The backbone is CSP-DarkNet, an improved version of DarkNet inspired by CSPNet. The neck uses Path Aggregation Network (PAN) and Feature Pyramid Network (FPN) techniques to fuse features at different scales. During prediction, the "Decoupled Head" technique splits localization and classification into parallel branches, enhancing accuracy and speed. Experimental results show that YoloMask outperforms state-of-the-art models on popular face detection datasets. A novel composite loss function, alpha-CIoU, is introduced, merging CIoU and alpha-IoU losses to improve performance. The CIoU loss considers the distance between midpoints, width-to-height ratio, and IoU, while the alpha-IoU loss considers the confidence score of the predicted box. The YoloMask model is trained using data augmentation techniques like Mosaic [131] and MixUp [132], and it uses the multi-positives trick, designating the center 3x3 area as positives. The Optimal Transport Assignment (OTA) technique is employed for label assignment, approximating solutions effectively. Evaluated on the Diverse Masked Faces dataset, YoloMask outperforms other state-of-the-art methods. It effectively detects different types of mask-wearing, making it suitable for monitoring proper mask usage in public places, especially important during the COVID-19 pandemic.

Sanchez et al. [70] presented an efficient facial recognition system for real-time operation in unconstrained environments. They combined deep learning techniques, such as FaceNet, with traditional classifiers like K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Random Forest (RF). The system includes face detection, preprocessing, feature extraction, and comparison stages. Using YOLO-Face, a real-time face detector based on YOLOv3, the system detects faces with over 89.6% accuracy on the Honda/UCSD dataset [133], handling partial occlusion, pose variations, and small faces. The preprocessing stage enhances image quality, while the feature extraction stage employs FaceNet and a supervised learning algorithm. FaceNet+SVM achieves 99.7% accuracy on the LFW dataset, with FaceNet+RF and FaceNet+KNN achieving 85.1% and 99.5% respectively. The system reaches 99.1% person identification accuracy by comparing extracted features to enrolled users' known features. The combined face detection and classification stages operate in 49 milliseconds, demonstrating high performance on unconstrained datasets.

Ali-Gombe et al. [102] proposed a technique for face detection using YOLO on edge devices. YOLO, a well-known object detection algorithm, divides an image into grids of varying sizes, such as 52×52, 26×26, and 13×13, and outputs bounding box coordinates for detected objects. YOLO employs a convolutional neural network backbone with multiple convolution layers and a fully connected layer for prediction. The YOLO v3-tiny model, a simplified version of YOLO v3, uses smaller grid sizes (26×26 and 13×13) and only nine convolution layers for faster processing but lower accuracy. The 33m-YOLO model, based on YOLO v3-tiny, includes ten convolution layers, resulting in a 34.8 MB model with 8,676,244 parameters and 5.448 BFLOPS. The 10m-YOLO model, a smaller version, removes layers seven to nine and the final max-pooling layer, resulting in a 10.1 MB model with 2,508,692 parameters and 3.365 BFLOPS. The 5m-YOLO model further reduces the sixth layer's filters from 512 to 256, resulting in a 5.7 MB model with 1,388,948 parameters and 2.987 BFLOPS. The 2m-YOLO model uses 1x1 filters in all convolutions in the second part, creating a 2.7 MB model with 602,516 parameters and 1.924 BFLOPS. Experiments tested the 10m-YOLO, 5m-YOLO, and 2m-YOLO models. While smaller than the 33m-YOLO model, they had a slight reduction in performance. On the WiderFaces dataset, the 2m-YOLO model, 16 times smaller than 33m-YOLO, had a mean Average Precision (mAP) of only 4 points lower. On the FDDB dataset, the smaller models had higher F1 scores than the 33m-YOLO model but also a higher false positive rate, attributed to the dataset. Regarding speed, the 2m-YOLO model was the fastest, achieving 25.9 FPS when tested on a core i5 MacBook Pro-2019 running a pre-recorded one-minute video.

Based on the YOLO architecture, Garg et al. [104] suggested a deep-learning technique for face detection. The study utilized the FDDB dataset, comprising 2,845 images and 5,171 faces, to train and test the proposed face detection model. The model architecture includes seven convolutional layers, a 2x2 max pooling layer, and three fully connected layers. The output layer uses the Non-Maximum Suppression (NMS) technique to predict bounding box coordinates and class probabilities. After tuning various performance parameters, the model was optimized by selecting the best values. The training process was carried out for 25 epochs using a gradient descent optimizer with a learning rate of 0.0001. The suggested approach achieved a high accuracy of 92.2% on the dataset called the FDDB dataset.

Ranjan et al. [58] introduced DP2MFD, a face detection algorithm combining deep pyramidal features and Deformable

Part Models (DPM) to address unconstrained conditions. DP2MFD includes a normalization layer in its deep convolutional neural network (CNN) to mitigate bias towards specific face sizes in deep feature representations. DP2MFD comprises two modules. The first module generates a normalized deep feature pyramid with seven levels for any input image size using a sliding window approach. Fixed-length features are extracted from each pyramid location. The second module employs a linear support vector machine (SVM) to classify pyramid locations as face or non-face based on their scores. Training DP2MFD involved using the FDDB dataset and Caffe framework to train both 1-component (DP2MFD-1c) and 2-component (DP2MFD-2c) DPMs. Positive and negative training samples were collected directly from the deep feature pyramid, and z-score normalization was applied to max5 features at each level to reduce size bias. DP2MFD was evaluated on Face Detection Dataset and Benchmark (FDDB), IARPA Janus Benchmark A (IJB-A), Annotated Face in-the-Wild (AFW), and the Multi-Attribute Labelled Faces (MALF) datasets, achieving new state-of-the-art performance in unconstrained face detection. It detects profile faces and faces of various sizes in cluttered backgrounds. Non-maximum suppression and bounding box regression techniques were employed to improve bounding box accuracy, contributing to DP2MFD's overall high performance.

Zeng et al. [128] proposed a fast and accurate multi-scale face detection method using multi-task learning and network acceleration techniques in a CNN cascade face detector. The method consists of three stages: the first stage is a fully convolutional network with a pyramid architecture that generates multi-scale face proposals with minimal image resizing, detecting faces in a 12x12 window and processing larger windows in subsequent branches. The second and third stages refine face proposals with bounding box and facial landmark regression, using non-maximum suppression (NMS) to eliminate overlaps. To enhance the method, extensive data augmentations, online and offline hard sample mining, and a novel multi-layer merging technique were employed. Network compression and acceleration techniques, including knowledge distilling and merging batch normalization layers with neighboring convolutions, improved inference speed. The network was designed to balance computational efficiency and accuracy, using increased convolutional strides instead of pooling layers for down-sampling. The face detector was tested on the FDDB benchmark, achieving competitive results with state-of-the-art methods and operating at 165 frames per second on a Titan GPU. Key contributions include the novel pyramid network, hard sample mining techniques, and performance improvements via knowledge distilling and multi-layer merging.

VI. RESULTS

This study categorizes the tasks of face detection, facial feature detection, and face recognition into four primary approaches: knowledge-based, template-matching, feature-based, and appearance-based methods. Through a systematic review of 28 research papers spanning from 1991 to 2023, key findings and advancements in face detection and recognition were identified, along with insights into their limitations and applications.

1. Knowledge-Based Methods:

These methods rely on predefined rules and simple image processing techniques such as histogram thresholding, edge detection, and noise reduction. They exhibit high accuracy (over 95%) for detecting frontal faces in plain backgrounds but lack robustness in complex images or multi-face detection scenarios. Computationally efficient, these methods are unsuitable for real-time or multi-face detection tasks due to their limited adaptability.

2. Feature-Based Methods:

Widely used in real-time applications, these methods employ techniques like skin color segmentation, Haar-like features, AdaBoost, and thresholding. Feature-based approaches achieved high accuracy (over 90%) in face detection and recognition, with the ability to detect multiple faces and facial features efficiently. While effective, these methods are less accurate than appearance-based approaches in handling pose variations and occlusions.

3. Template-Matching Methods:

Template-based methods compare predefined templates with detected regions in the image, offering limited flexibility. Although effective in controlled environments, their computational cost and inability to adapt to variations in scale, pose, and lighting conditions make them less suitable for modern real-world applications.

4. Appearance-Based Methods:

The most widely used category, leveraging advanced machine learning and deep learning techniques such as CNN, Faster R-CNN, YOLO, and hybrid convolutional networks. Achieved significant advancements in accuracy (over 99%) and robustness, effectively handling challenges like pose, scale, and occlusions. Techniques such as data augmentation, multi-task learning, and batch normalization have improved detection speed and performance, making these methods highly suitable for real-time applications.

5. Deep Learning Algorithms:

Algorithms like MTCNN, DCNN, and YOLO-Face models dominate the field due to their scalability, accuracy, and efficiency. The integration of traditional classifiers (e.g., SVM, KNN, Random Forest) with deep learning models has further enhanced their performance. These approaches have been applied extensively in applications like biometric systems, driver drowsiness detection, and surveillance.

6. Trends Over Time:

From 1991 to 2010, research focused primarily on knowledge-based and feature-based methods with limited capabilities.

After 2010, deep learning techniques revolutionized the field, enabling significant improvements in accuracy, adaptability, and speed.

Key Outcomes:

Accuracy: Deep learning methods consistently outperform traditional techniques, achieving over 99% accuracy in face detection and recognition tasks.

Applications: Modern algorithms are robust against variations in environmental conditions, enabling their use in real-world scenarios such as surveillance, healthcare, and automotive safety.

Challenges: Despite advancements, further research is needed to address issues such as computational overhead, dataset bias, and performance in extreme environmental conditions.

This systematic review highlights the evolution of face detection and recognition methods and underscores the dominance of deep learning approaches in addressing real-world challenges while identifying areas for future research.

VII. DISCUSSION AND CONCLUSION

The literature discussed in this paper categorizes the tasks of face detection, facial feature detection, and face recognition into four primary categories: knowledge-based, template matching, feature-based, and appearance-based. As shown in

Figure 3, the majority of research has focused on face detection. Moreover, most of the algorithms for face detection and recognition have been developed using the Appearance-Based technique, as depicted in

Figure 4.

Figure 4 provides an overview of the research conducted in face detection and recognition using knowledge-based and

feature-based approaches, indicating that there have been no recent studies in these categories. However, three research studies were identified from 1991 to the present.

In recent studies, deep learning algorithms have been frequently employed to address the challenges of face detection and recognition. The literature highlights the use of machine learning algorithms such as Haar Cascade Classifiers, AdaBoost, CNN, DCNN, and Faster RCNN in various face detection and recognition approaches. Among these algorithms, CNN, DCNN, and Faster RCNN have gained popularity among researchers for their effectiveness in face detection and recognition tasks. Additionally, SVM classifiers are commonly utilized for classification tasks in this domain.

According to Table 2, Knowledge-based methods are used for detecting a human face and facial features like eyes, nose, and mouth. There are several image processing techniques like histogram thresholding, noise reduction, edge detection methods, and edge segments were used to enhance the accuracy of these algorithms. In general, these methods require low computational power, and some algorithms have achieved over 95% accuracy in face detection. According to the literature discussed in this paper, most of the algorithms can only detect one face and are not suitable for complex images and multi-face detection. Also, these algorithms are only accurate for frontal view with plain background images.

In **Error! Reference source not found.**, Feature-based m ethods find extensive application in real-time scenarios involving face detection, facial feature detection, and face recognition. Various techniques including skin color segmentation, face boundary estimation, eyes candidate estimation, thresholding, Haar-like features, classifiers, feature patch templates, and the AdaBoost algorithm, are employed to implement these algorithms. The literature shows that these algorithms achieved high accuracy and successfully detected multiple faces and eye coordinates in real-time. Moreover, these algorithms achieved over 90% accuracy in face recognition and fast detection time.

According to the literature, these algorithms exhibit high accuracy in detecting faces with significant variations in pose, scale, and occlusion. They can achieve accurate results even in complex backgrounds and images containing multiple faces.

Table 5 provides a summary of research conducted on Appearance-Based methods for face detection and recognition, emphasizing their notable success in these areas as well as facial feature detection. Researchers have developed various one-stage and two-stage algorithms, which are extensively utilized in real-time applications.

Two-stage object detectors such as CNN, DCNN, Fast RCNN, and Faster RCNN, as well as one-stage object detectors like YOLO-Face, 10m-YOLO model, 5m-YOLO model, and 2m-YOLO model, have been employed to achieve accurate and efficient face detection and recognition. The literature reveals that deep learning techniques, including MTCNN, DCNN, YOLO, and hybrid convolutional networks like ConvNet, have been combined with traditional classifiers such as K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Random Forest (RF).

Table 5 provides a summary of research conducted on Appearance-Based methods for face detection and recognition, emphasizing their notable success in these areas as well as facial feature detection. Researchers have developed various one-stage and two-stage algorithms, which are extensively utilized in real-time applications.

Two-stage object detectors such as CNN, DCNN, Fast RCNN, and Faster RCNN, as well as one-stage object detectors like YOLO-Face, 10m-YOLO model, 5m-YOLO model, and 2m-YOLO model, have been employed to achieve accurate and

efficient face detection and recognition. The literature reveals that deep learning techniques, including MTCNN, DCNN, YOLO, and hybrid convolutional networks like ConvNet, have been combined with traditional classifiers such as K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Random Forest (RF).

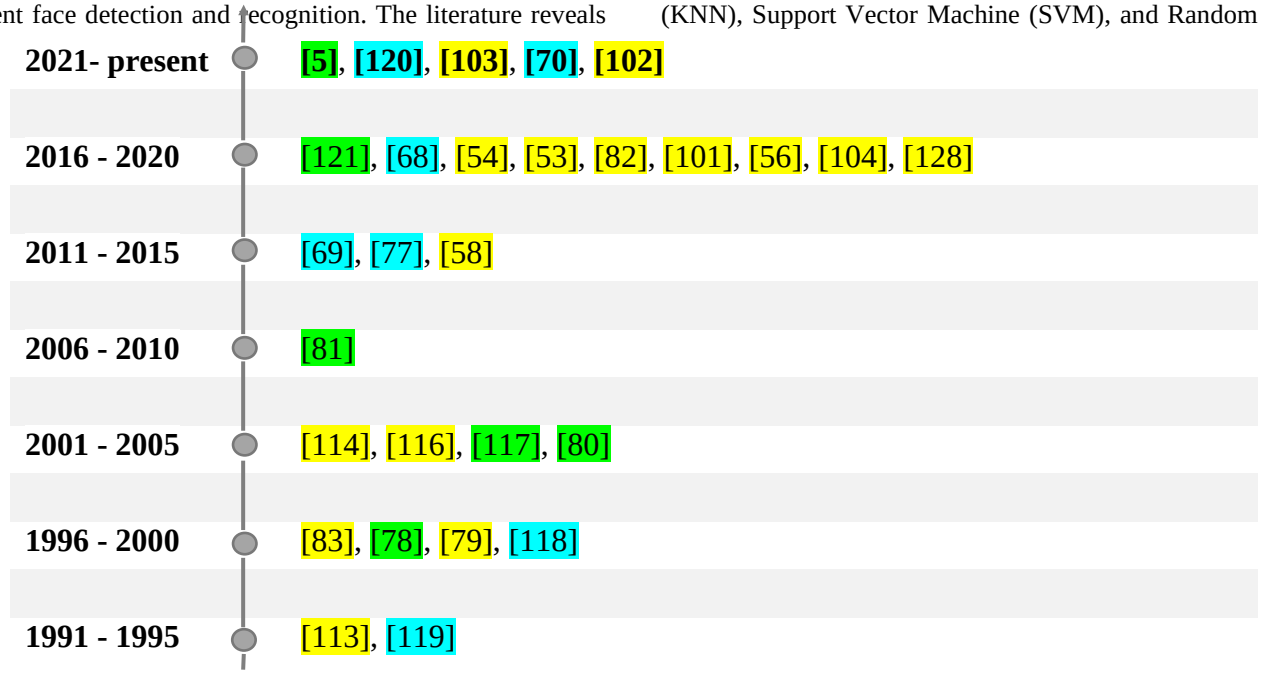

Figure 3 Timeline of research where face detection, facial feature detection, and face recognition method were used; Background colors represent different tasks of problems: face detection, facial feature detection, face recognitio

To enhance the accuracy of face detection and address challenges like pose, scale, and occlusion variations, researchers have employed a variety of training strategies, including data augmentation techniques, multi-task learning, network acceleration techniques, multi-layer merging, down-sampling, and batch normalization. These techniques have enabled the algorithms to achieve accuracy rates of over 99\% in face detection. This demonstrates that these algorithms surpass previous categories in terms of accuracy, making them highly capable of face detection and recognition.

Researchers commonly employed Deep Learning algorithms, highlighting their significant impact on face detection,

recognition, and feature extraction. This observation indicates the remarkable position attained by Deep Learning algorithms in these fields. Due to the increasing complexity of real-world problems, fast and accurate face detection, recognition, and feature extraction have become widely discussed topics in computer vision and object detection. Consequently, significant amounts of research have been conducted to find effective algorithms for these tasks. This paper aims to provide an overview of state-of-the-art research and various mechanisms used in face detection, recognition, and feature extraction.

A total of twenty-eight papers were selected for this study. Beginning with a comprehensive introduction to the topic, we discuss modern algorithms utilized in face detection, recognition, and feature extraction. Additionally, we explore different problem-generation mechanisms employed in various research studies related to these tasks. The content of the paper highlights the fact that face detection, recognition, and feature extraction are considered crucial aspects of object detection, and numerous solutions have been proposed. However, there is still ample room for further research to develop mechanisms that enhance face detection, recognition, and feature extraction, thus making significant contributions to the field of computer vision.

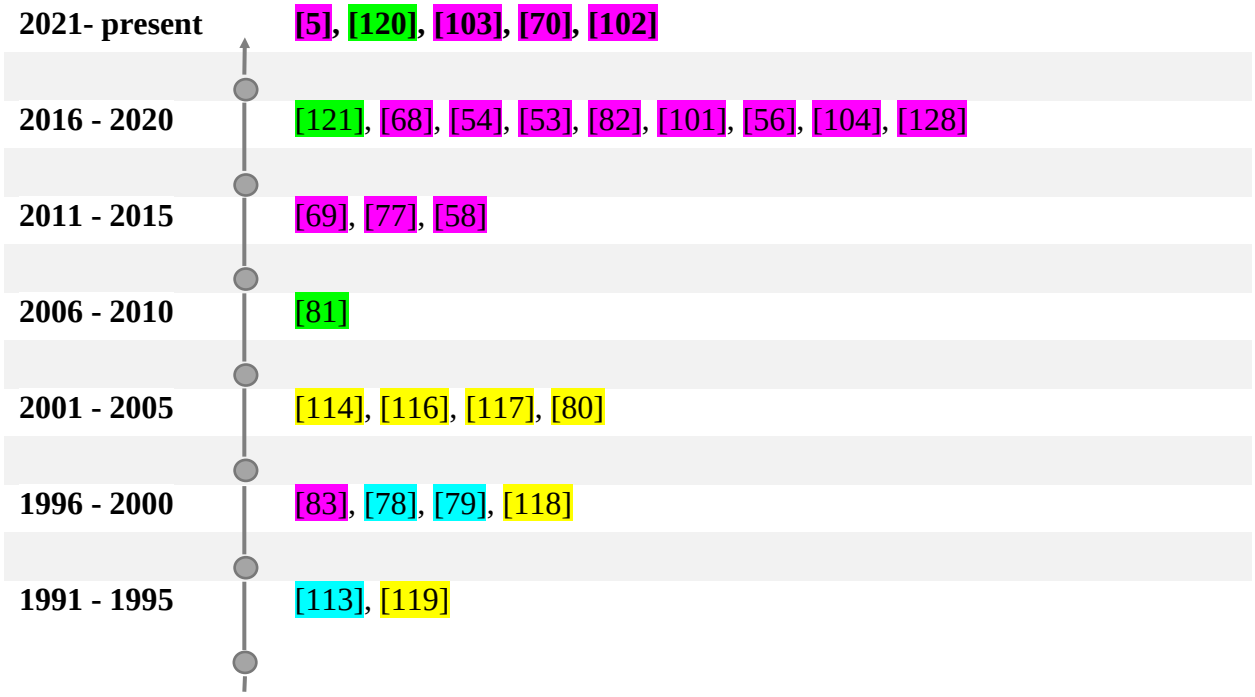

Figure 4 Timeline of research where face detection and recognition method were used; Background colors represent different categories of problems: Knowledge-Based, Feature-Based, Template Matching, Appearance-Based

REFERENCES

[1] Y.-Q. Wang, "An Analysis of the Viola-Jones Face Detection Algorithm," Image Process. Line, vol. 4, pp. 128–148, Jun. 2014, doi: 10.5201/ipol.2014.104.

[2] G. Lowe, "Sift-the scale invariant feature transform," Int J, vol. 2, no. 91–110, p. 2, 2004.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 1, pp. 142–158, 2015.

[4] S. E. B. D. P. Ltd.23/4736, A. Road, and D. Delhi 110 002, "Multimedia Image and Video Processing," Routledge & CRC Press. Accessed: Mar. 16, 2023. [Online]. Available: https://www.routledge.com/Multimedia-Image-and-Video-Processing/Guan-He-Kung/p/book/9781138072534

[5] X. Yang and W. Zhang, "Heterogeneous face detection based on multi-task cascaded convolutional neural network," IET Image Process., vol. 16, Jan. 2022, doi: 10.1049/ipr2.12344.

[6] T. S. Srinivas, T. Goutham, and D. M. S. Kumaran, “Face Recognition based Smart Attendance System Using IoT,” vol. 09, no. 03, 2022.

[7] A. A. Alsanabani, M. A. Ahmed, and A. M. Al Smadi, “Vehicle Counting Using Detecting-Tracking Combinations: A Comparative Analysis,” in Proceedings of the 2020 4th International Conference on Video and Image Processing, in ICVIP ’20. New York, NY, USA: Association for Computing Machinery, Apr. 2021, pp. 48–54. doi: 10.1145/3447450.3447458.

[8] D. A. A. Deepal and T. G. I. Fernando, “Convolutional Neural Network Approach for the Detection of Lung Cancers in Chest X-Ray Images,” in Deep Learning for Cancer Diagnosis, U. Kose and J. Alzubi, Eds., in Studies in Computational Intelligence. , Singapore: Springer, 2021, pp. 203–226. doi: 10.1007/978-981-15-6321-8_12.

[9] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. Rehg, “A Scalable Approach to Activity Recognition based on Object Use,” Nov. 2007, pp. 1–8. doi: 10.1109/ICCV.2007.4408865.

[10] M. A. Berbar, H. M. Kelash, and A. A. Kandeel, “Faces and Facial Features Detection in Color Images,” in Geometric Modeling and Imaging–New Trends (GMAI’06), Jul. 2006, pp. 209–214. doi: 10.1109/GMAI.2006.18.

[11] H. Hatem, Z. Beiji, and R. Majeed, “A Survey of Feature Base Methods for Human Face Detection,” Int. J. Control Autom., vol. 8, no. 5, pp. 61–78, May 2015, doi: 10.14257/ijca.2015.8.5.07.

[12] J. Chatrath, P. Gupta, P. Ahuja, A. Goel, and S. M. Arora, “Real time human face detection and tracking,” 2014 Int. Conf. Signal Process. Integr. Netw. SPIN, pp. 705–710, Feb. 2014, doi: 10.1109/SPIN.2014.6777046.

[13] I. R. Tsang, J. P. Magalhaes, and G. D. C. Cavalcanti, “Combined AdaBoost and gradientfaces for face detection under illumination problems,” in 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Oct. 2012, pp. 2354–2358. doi: 10.1109/ICSMC.2012.6378094.

[14] W. Liu et al., “SSD: Single Shot MultiBox Detector,” vol. 9905, 2016, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.

[15] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” Jan. 06, 2016, arXiv: arXiv:1506.01497. Accessed: Mar. 16, 2023. [Online]. Available: http://arxiv.org/abs/1506.01497

[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[17] D. Q. Rizvi, “A Review on Face Detection Methods,” J. Manag. Dev. Inf. Technol., vol. 11, Feb. 2011.

[18] M. C. P. Archana, C. K. Nitish, and S. Harikumar, “Real time Face Detection and Optimal Face Mapping for Online Classes,” J. Phys. Conf. Ser., vol. 2161, no. 1, p. 012063, Jan. 2022, doi: 10.1088/1742-6596/2161/1/012063.

[19] Sciforce, “Face Detection Explained: State-of-the-Art Methods and Best Tools,” Sciforce. Accessed: Mar. 16, 2023. [Online]. Available: https://medium.com/sciforce/face-detection-explained-state-of-the-art-methods-and-best-tools-f730fca16294

[20] D. Dwivedi, “Face Detection For Beginners,” Medium. Accessed: Mar. 16, 2023. [Online]. Available: https://towardsdatascience.com/face-detection-for-beginners-e58e8f21aad9

[21] S. Harikumar and R. Ramachandran, “Hybridized fragmentation of very large databases using clustering,” in 2015 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES), Feb. 2015, pp. 1–5. doi: 10.1109/SPICES.2015.7091488.

[22] R. Kaur and S. Singh, “A comprehensive review of object detection with deep learning,” Digit. Signal Process., vol. 132, p. 103812, Jan. 2023, doi: 10.1016/j.dsp.2022.103812.

[23] Y. Xiao et al., “A review of object detection based on deep learning,” Multimed. Tools Appl., vol. 79, no. 33, pp. 23729–23791, Sep. 2020, doi: 10.1007/s11042-020-08976-6.

[24] “A Detailed Review on Object Detection Algorithms | IEEE Conference Publication | IEEE Xplore.” Accessed: Dec. 30, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10125764

[25] Z.-Q. Zhao, P. Zheng, S. Xu, and X. Wu, “Object Detection with Deep Learning: A Review,” Apr. 16, 2019, arXiv: arXiv:1807.05511. Accessed: Dec. 30, 2023. [Online]. Available: http://arxiv.org/abs/1807.05511

[26] “A Survey of Face Recognition Techniques.” Accessed: Jun. 30, 2023. [Online]. Available: https://www.researchgate.net/publication/220635738_A_Survey_of_Face_Recognition_Techniques

[27] M. Lal, K. Kumar, R. Hussain, A. Maitlo, S. Ali, and H. Shaikh, “Study of Face Recognition Techniques: A Survey,” Int. J. Adv. Comput. Sci. Appl., vol. 9, no. 6, 2018, doi: 10.14569/IJACSA.2018.090606.

[28] D. Moher, A. Liberati, J. Tetzlaff, and D. G. Altman, “Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement,” BMJ, vol. 339, p. b2535, Jul. 2009, doi: 10.1136/bmj.b2535.

[29] “Don’t Take Facial Recognition at Face Value: Application Examples Across Different Industries,” AnyConnect. Accessed: Jan. 16, 2024. [Online]. Available: https://anyconnect.com/blog/facial-recognition-applications/

[30] “The Uses of Facial Recognition Across Industries | Mindy Support Outsourcing.” Accessed: Jan. 16, 2024. [Online]. Available: https://mindy-support.com/news-post/the-uses-of-facial-recognition-across-industries/

[31] “Facial Recognition (Updated with Examples).” Accessed: Jan. 16, 2024. [Online]. Available: https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/biometrics/facial-recognition

[32] “Sensors | Free Full-Text | Face Recognition Systems: A Survey.” Accessed: Jan. 16, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/20/2/342

[33] “About Face ID advanced technology,” Apple Support. Accessed: Jan. 16, 2024. [Online]. Available: https://support.apple.com/en-us/102381

[34] “What is biometric unlock technology? Face recognition history | Blackview Blog.” Accessed: Jan. 16, 2024. [Online]. Available: https://www.blackview.hk/blog/tech-news/what-is-biometric-unlock

[35] E. Czerwonka, “5 Best Attendance Systems with Face Recognition.” Accessed: Jan. 16, 2024. [Online]. Available: https://buddypunch.com/blog/attendance-system-face-recognition/

[36] “Top 10 Photo Manager Software with Facial Recognition: A Comprehensive Guide | Daminion Blog.” Accessed: Jan. 16, 2024. [Online]. Available: https://daminion.net/articles/tools/photo-management-software-with-facial-recognition/

[37] E. Fatekhov, “Top 12 Photo Managers with Face Recognition.” Accessed: Jan. 16, 2024. [Online]. Available: https://tonfotos.com/articles/best-face-recognition-software/

[38] “Make Your Own Face Filters in PictoBlox Using the Face Detection,” STEMpedia Education. Accessed: Jan. 16, 2024. [Online]. Available: https://ai.thestempedia.com/project/make-your-own-face-filters-in-pictoblox-using-the-face-detection/

[39] M. Ilves, Y. Gizatdinova, V. Surakka, and E. Vankka, “Head movement and facial expressions as game input,” Entertain. Comput., vol. 5, no. 3, pp. 147–156, Aug. 2014, doi: 10.1016/j.entcom.2014.04.005.

[40] C. Zhan, W. Li, P. Ogunbona, and F. Safaei, “A Real-Time Facial Expression Recognition System for Online Games,” Int. J. Comput. Games Technol., vol. 2008, p. e542918, Mar. 2008, doi: 10.1155/2008/542918.

[41] “Emotional response evoked by viewing facial expression pictures leads to higher temporal resolution - Misa Kobayashi, Makoto Ichikawa, 2023.” Accessed: Jan. 16, 2024. [Online]. Available: https://journals.sagepub.com/doi/10.1177/20416695231152144

[42] “How Emotion-Detection Technology Will Change Marketing.” Accessed: Jan. 16, 2024. [Online]. Available: https://blog.hubspot.com/marketing/emotion-detection-technology-marketing

[43] X. Kong, Z. Wang, J. Sun, X. Qi, and Q. Qiu, “Facial Recognition for Disease Diagnosis Using a Deep Learning Convolutional Neural Network: A systematic Review and Meta-Analysis”.

[44] J. Qiang, D. Wu, H. Du, H. Zhu, S. Chen, and H. Pan, “Review on Facial-Recognition-Based Applications in Disease Diagnosis,” Bioengineering, vol. 9, no. 7, p. 273, Jun. 2022, doi: 10.3390/bioengineering9070273.

[45] B. Abirami, T. S. Subashini, and V. Mahavaishnavi, “Gender and age prediction from real time facial images using CNN,” Mater. Today Proc., vol. 33, pp. 4708–4712, Jan. 2020, doi: 10.1016/j.matpr.2020.08.350.

[46] S. Haseena, S. Saroja, R. Madavan, A. Karthick, B. Pant, and M. Kifetew, “Prediction of the Age and Gender Based on Human Face Images Based on Deep Learning Algorithm,” Comput. Math. Methods Med., vol. 2022, p. e1413597, Aug. 2022, doi: 10.1155/2022/1413597.

[47] W. S. M. Sanjaya, D. Anggraeni, K. Zakaria, A. Juwardi, and M. Munawwaroh, “The design of face recognition and tracking for human-robot interaction,” Nov. 2017, pp. 315–320. doi: 10.1109/ICITISEE.2017.8285519.

[48] Y. Wang, J. Shen, S. Petridis, and M. Pantic, “A real-time and unsupervised face re-identification system for human-robot interaction,” Pattern Recognit. Lett., vol. 128, pp. 559–568, Dec. 2019, doi: 10.1016/j.patrec.2018.04.009.

[49] “AI Facial Recognition with Temperature Measurement | Solution - GIGABYTE Global,” GIGABYTE. Accessed: Jan. 16, 2024. [Online]. Available: https://www.gigabyte.com/Solutions/facialntemp

[50] G. Bae et al., “DigiFace-1M: 1 Million Digital Face Images for Face Recognition,” presented at the Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023, pp. 3526–3535. Accessed: Jan. 16, 2024. [Online]. Available: https://openaccess.thecvf.com/content/WACV2023/html/Bae_DigiFace-1M_1_Million_Digital_Face_Images_for_Face_Recognition_WACV_2023_paper.html

[51] “FDDB : Main.” Accessed: Apr. 17, 2023. [Online]. Available: http://vis-www.cs.umass.edu/fddb/index.html#explore

[52] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks,” IEEE Signal Process. Lett., vol. 23, no. 10, pp. 1499–1503, Oct. 2016, doi: 10.1109/LSP.2016.2603342.

[53] X. Sun, P. Wu, and S. C. H. Hoi, “Face detection using deep learning: An improved faster RCNN approach,” Neurocomputing, vol. 299, pp. 42–50, Jul. 2018, doi: 10.1016/j.neucom.2018.03.030.

[54] H. Jiang and E. Learned-Miller, “Face Detection with the Faster R-CNN,” in 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), May 2017, pp. 650–657. doi: 10.1109/FG.2017.82.

[55] “WIDER FACE: A Face Detection Benchmark.” Accessed: Apr. 17, 2023. [Online]. Available: http://shuoyang1213.me/WIDERFACE/

[56] J. Wang, Y. Yuan, and G. Yu, “Face Attention Network: An Effective Face Detector for the Occluded Faces,” Nov. 22, 2017, arXiv: arXiv:1711.07246. doi: 10.48550/arXiv.1711.07246.

[57] “CelebA Dataset.” Accessed: Apr. 17, 2023. [Online]. Available: https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

[58] R. Ranjan, V. M. Patel, and R. Chellappa, “A Deep Pyramid Deformable Part Model for Face Detection,” Aug. 18, 2015, arXiv: arXiv:1508.04389. doi: 10.48550/arXiv.1508.04389.

[59] S. Yang, P. Luo, C. C. Loy, and X. Tang, “Faceness-Net: Face Detection through Deep Facial Part Responses,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 8, pp. 1845–1859, Aug. 2018, doi: 10.1109/TPAMI.2017.2738644.

[60] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, “VGGFace2: A dataset for recognising faces across pose and age,”

May 13, 2018, arXiv: arXiv:1710.08092. Accessed: Apr. 17, 2023. [Online]. Available: http://arxiv.org/abs/1710.08092

[61] O. A. Aghdam, B. Bozorgtabar, H. K. Ekenel, and J.-P. Thiran, "Exploring Factors for Improving Low Resolution Face Recognition," in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Jun. 2019, pp. 2363–2370. doi: 10.1109/CVPRW.2019.00290.

[62] NVlabs/ffhq-dataset. (Apr. 17, 2023). Python. NVIDIA Research Projects. Accessed: Apr. 17, 2023. [Online]. Available: https://github.com/NVlabs/ffhq-dataset

[63] T. Karras, S. Laine, and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks," Mar. 29, 2019, arXiv: arXiv:1812.04948. doi: 10.48550/arXiv.1812.04948.

[64] C. E. Bencheriet, H. Abdelmoumène, A. Sebbagh, A. Yahiyaoui, and Z. Taba, "Fake face detection based on a multi discriminator deep CNN architecture (MDD-CNN)," 2023, doi: 10.14311/AP.2023.63.0305.

[65] "Tufts Face Database." Accessed: Apr. 17, 2023. [Online]. Available: https://www.kaggle.com/datasets/kpvisionlab/tufts-face-database

[66] P. Martins, J. Silva, and A. Bernardino, "Multispectral Facial Recognition in the Wild," Sensors, vol. 22, p. 4219, Jun. 2022, doi: 10.3390/s22114219.

[67] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments".

[68] S. Almabdy and L. Elrefaei, "Deep Convolutional Neural Network-Based Approaches for Face Recognition," Appl. Sci., vol. 9, no. 20, Art. no. 20, Jan. 2019, doi: 10.3390/app9204397.

[69] Y. Sun, X. Wang, and X. Tang, "Hybrid Deep Learning for Face Verification," presented at the Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1489–1496. Accessed: Mar. 18, 2023. [Online]. Available: https://openaccess.thecvf.com/content_iccv_2013/html/Sun_Hybrid_Deep_Learning_2013_ICCV_paper.html

[70] A. S. Sanchez-Moreno, J. Olivares-Mercado, A. Hernandez-Suarez, K. Toscano-Medina, G. Sanchez-Perez, and G. Benitez-Garcia, "Efficient Face Recognition System for Operating in Unconstrained Environments," J. Imaging, vol. 7, no. 9, Art. no. 9, Sep. 2021, doi: 10.3390/jimaging7090161.

[71] "UTKFace," UTKFace. Accessed: Apr. 17, 2023. [Online]. Available: https://susanqq.github.io/UTKFace/

[72] H. A. Nugroho, R. D. Goratama, and E. L. Frannita, "Face recognition in four types of colour space: a performance analysis," IOP Conf. Ser. Mater. Sci. Eng., vol. 1088, no. 1, p. 012010, Feb. 2021, doi: 10.1088/1757-899X/1088/1/012010.

[73] S. J. Devaraj, R. Catherine Joy, I. Santhosh, and I. C. Kevin, "Deep Learning Based Facial Feature Detection for Ethnicity Recognition," in Smart Computing Techniques and Applications, S. C. Satapathy, V. Bhateja, M. N. Favorskaya, and T. Adilakshmi, Eds., in Smart Innovation, Systems and Technologies. Singapore: Springer, 2021, pp. 527–534. doi: 10.1007/978-981-16-1502-3_52.

[74] "Google facial expression comparison dataset," Google Research. Accessed: Apr. 18, 2023. [Online]. Available: https://research.google/resources/datasets/google-facial-expression/

[75] R. Vemulapalli and A. Agarwala, "A Compact Embedding for Facial Expression Similarity," in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA: IEEE, Jun. 2019, pp. 5676–5685. doi: 10.1109/CVPR.2019.00583.

[76] "YouTube Faces With Facial Keypoints." Accessed: Apr. 18, 2023. [Online]. Available: https://www.kaggle.com/datasets/selfishgene/youtube-faces-with-facial-keypoints

[77] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2015, pp. 815–823. doi: 10.1109/CVPR.2015.7298682.

[78] L. Zhang and P. Lenders, "Knowledge-based eye detection for human face recognition," in KES'2000. Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies. Proceedings (Cat. No. 00TH8516), IEEE, 2000, pp. 117–120.

[79] C. Kotropoulos and I. Pitas, "Rule-based face detection in frontal views," 1997 IEEE Int. Conf. Acoust. Speech Signal Process., vol. 4, pp. 2537–2540, 1997, doi: 10.1109/ICASSP.1997.595305.

[80] D. Vukadinovic and M. Pantic, "Fully automatic facial feature point detection using gabor feature based boosted classifiers," in 2005 IEEE International Conference on Systems, Man and Cybernetics, IEEE, 2005, pp. 1692–1698.

[81] T.-Y. Chai, R. M, W. San, and T. Seong, "Facial Features for Template Matching Based Face Recognition," Am. J. Appl. Sci., vol. 6, Nov. 2009, doi: 10.3844/ajassp.2009.1897.1901.

[82] G. Guo, H. Wang, Y. Yan, J. Zheng, and B. Li, "A Fast Face Detection Method via Convolutional Neural Network," Mar. 27, 2018, arXiv: arXiv:1803.10103. doi: 10.48550/arXiv.1803.10103.

[83] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 1, pp. 23–38, Jan. 1998, doi: 10.1109/34.655647.

[84] "TensorFlow.js | Machine Learning for JavaScript Developers," TensorFlow. Accessed: May 01, 2023. [Online]. Available: https://www.tensorflow.org/js

[85] "face-api.js." Accessed: May 13, 2023. [Online]. Available: https://justadudewhohacks.github.io/face-api.js/docs/index.html

[86] "@tensorflow-models/blazeface," npm. Accessed: Feb. 05, 2024. [Online]. Available: https://www.npmjs.com/package/@tensorflow-models/blazeface

[87] "Overview — OpenVINOTM documentation." Accessed: May 13, 2023. [Online]. Available: https://docs.openvino.ai/latest/home.html

[88] H. Wang and J. Hu, "Intelligent lecture recording system based on coordination of face-detection and pedestrian dead reckoning," PeerJ Comput. Sci., vol. 8, p. e971, May 2022, doi: 10.7717/peerj-cs.971.

[89] D. Brown, Mobile Attendance based on Face Detection and Recognition using OpenVINO. 2021, p. 1157. doi: 10.1109/ICAIS50930.2021.9395836.

[90] M. Basurah, W. Swastika, and O. H. Kelana, "IMPLEMENTATION OF FACE RECOGNITION AND LIVENESS DETECTION SYSTEM USING TENSORFLOW.JS," J. Inform. Polinema, vol. 9, no. 4, Art. no. 4, Aug. 2023, doi: 10.33795/jip.v9i4.1332.

[91] D. Yadav, S. Maniar, K. Sukhani, and K. Devadkar, "In-Browser Attendance System using Face Recognition and Serverless Edge Computing," in 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Jul. 2021, pp. 01–06. doi: 10.1109/ICCCNT51525.2021.9580042.

[92] "GitHub - opencv/opencv at master," GitHub. Accessed: May 13, 2023. [Online]. Available: https://github.com/opencv/opencv

[93] R. T. Hasan and A. B. Sallow, "Face Detection and Recognition Using OpenCV," J. Soft Comput. Data Min., vol. 2, no. 2, Art. no. 2, Oct. 2021.

[94] M. A. Hoque, T. Islam, T. Ahmed, and A. Amin, "Autonomous Face Detection System from Real-time Video Streaming for Ensuring the Intelligence Security System," 2020 6th Int. Conf. Adv. Comput. Commun. Syst. ICACCS, pp. 261–265, Mar. 2020, doi: 10.1109/ICACCS48705.2020.9074260.

[95] A. Das, M. Wasif Ansari, and R. Basak, "Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV," in 2020 IEEE 17th India Council International Conference (INDICON), Dec. 2020, pp. 1–5. doi: 10.1109/INDICON49873.2020.9342585.

[96] J. Mehariya, C. Gupta, N. Pai, S. Koul, and P. Gadakh, "Counting Students using OpenCV and Integration with Firebase for Classroom Allocation," Jul. 2020, pp. 624–629. doi: 10.1109/ICESC48915.2020.9155825.

[97] Z. Soomro, T. Memon, F. Naz, and A. Ali, "FPGA Based Real-Time Face Authorization System for Electronic Voting System," Jan. 2020, pp. 1–6. doi: 10.1109/iCoMET48670.2020.9073880.

[98] "#dotnet – Detecting Faces using DNN from the camera feed in a WinForm using #OpenCV and #net5 – El Bruno." Accessed: May 13, 2023. [Online]. Available: https://elbruno.com/2020/11/18/dotnet-detecting-faces-using-dnn-from-the-%F0%9F%8E%A6-camera-feed-in-a-winform-using-opencv-and-net5/

[99] "EmguCV #62: Face Landmark Detection from Images - YouTube." Accessed: May 13, 2023. [Online]. Available: https://www.youtube.com/watch?v=ZOt-A7-Ehq0

[100] Q. Xu, Z. Zhu, H. Ge, Z. Zhang, and X. Zang, "Effective Face Detector Based on YOLOv5 and Superresolution Reconstruction," Comput. Math. Methods Med., vol. 2021, p. e7748350, Nov. 2021, doi: 10.1155/2021/7748350.

[101] F. Zhang, X. Fan, G. Ai, J. Song, Y. Qin, and J. Wu, "Accurate face detection for high performance," ArXiv Prepr. ArXiv190501585, 2019.

[102] A. Ali-Gombe, E. Elyan, and J. Zwiegelaar, "Face detection with YOLO on edge.," Jul. 2021, doi: 10.1007/978-3-030-80568-5_24.

[103] Z. Cao, W. Li, H. Zhao, and L. Pang, "YoloMask: An Enhanced YOLO Model for Detection of Face Mask Wearing Normality, Irregularity and Spoofing," Nov. 2022, pp. 205–213. doi: 10.1007/978-3-031-20233-9_21.

[104] D. Garg, P. Goel, S. Pandya, A. Ganatra, and K. Kotecha, "A Deep Learning Approach for Face Detection using YOLO," in 2018 IEEE Punecon, Nov. 2018, pp. 1–4. doi: 10.1109/PUNECON.2018.8745376.

[105] S. Ding, L. Lin, G. Wang, and H. Chao, "Deep feature learning with relative distance comparison for person re-identification," Pattern Recognit., vol. 48, no. 10, Art. no. 10, Oct. 2015, doi: 10.1016/j.patcog.2015.04.005.

[106] L. Wolf, "Face Recognition, Geometric vs. Appearance-Based," in Encyclopedia of Biometrics, S. Z. Li and A. Jain, Eds., Boston, MA: Springer US, 2009, pp. 347–352. doi: 10.1007/978-0-387-73003-5_92.

[107] "CBCL FACE RECOGNITION DATABASE." Accessed: Mar. 15, 2024. [Online]. Available: http://cbcl.mit.edu/software-datasets/heisele/facerecognition-database.html

[108] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-PIE," in 2008 8th IEEE International Conference on Automatic Face & Gesture Recognition, Sep. 2008, pp. 1–8. doi: 10.1109/AFGR.2008.4813399.

[109] "The CMU Multi-PIE Face Database." Accessed: Mar. 15, 2024. [Online]. Available: https://www.cs.cmu.edu/afs/cs/project/PIE/MultiPie/Multi-Pie/Home.html

[110] G. Boesch, "Object Detection in 2023: The Definitive Guide," viso.ai. Accessed: Mar. 16, 2023. [Online]. Available: https://viso.ai/deep-learning/object-detection/

[111] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in 2017 IEEE International Conference on Computer Vision (ICCV), Oct. 2017, pp. 2980–2988. doi: 10.1109/ICCV.2017.322.

[112] "Object Detection: Models, Architectures & Tutorial [2023]." Accessed: Mar. 16, 2023. [Online]. Available: https://www.v7labs.com/blog/object-detection-guide#h2

[113] G. Yang and T. S. Huang, "Human face detection in a complex background," Pattern Recognit., vol. 27, no. 1, pp. 53–63, Jan. 1994, doi: 10.1016/0031-3203(94)90017-5.

[114] Y. H. Chan and S. A. R. Abu-Bakar, "Face detection system based on feature-based chrominance colour information," in Proceedings. International Conference on Computer Graphics, Imaging and Visualization, 2004. CGIV 2004., IEEE, 2004, pp. 153–158.

[115] R.-L. Hsu, M. Abdel-Mottaleb, and A. K. Jain, "Face detection in color images," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 5, pp. 696–706, May 2002, doi: 10.1109/34.1000242.

[116] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern

Recognition. CVPR 2001, Kauai, HI, USA: IEEE Comput. Soc, 2001, p. I-511-I–518. doi: 10.1109/CVPR.2001.990517.

[117] I. R. Fasel, B. Fortenberry, and J. R. Movellan, “GBoost: A generative framework for boosting with applications to realtime eye coding,” Comput. Vis. Image Underst., vol. 98, no. 1, pp. 182–210, 2005.

[118] I. J. Cox, J. Ghosn, and P. N. Yianilos, “Feature-based face recognition using mixture-distance,” in Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA: IEEE, 1996, pp. 209–216. doi: 10.1109/CVPR.1996.517076.

[119] B. S. Manjunath, R. Chellappa, and C. von der Malsburg, “A feature based approach to face recognition,” in Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Jun. 1992, pp. 373–378. doi: 10.1109/CVPR.1992.223162.

[120] N. Abdulsada and S. Ali, “Human face detection in a crowd image based on template matching technique,” presented at the AIP Conference Proceedings, Aug. 2022, p. 020033. doi: 10.1063/5.0093156.

[121] P. Bose and S. Bandyopadhyay, “Human Face and Facial Parts Detection using Template Matching Technique,” Int. J. Eng. Adv. Technol., vol. 9, pp. 2249–8958, May 2020, doi: 10.35940/ijeat.D6689.049420.

[122] S. M. Smith and J. M. Brady, “SUSAN—A New Approach to Low Level Image Processing,” Int. J. Comput. Vis., vol. 23, no. 1, pp. 45–78, May 1997, doi: 10.1023/A:1007963824710.

[123] “Pubfig: Public Figures Face Database.” Accessed: Mar. 15, 2024. [Online]. Available: https://www.cs.columbia.edu/CAVE/databases/pubfig/

[124] “The PEAL Face Database.” Accessed: Mar. 15, 2024. [Online]. Available: http://www.jdl.link/peal/index.html

[125] “The PASCAL Visual Object Classes Challenge 2012 (VOC2012).” Accessed: Mar. 15, 2024. [Online]. Available: http://host.robots.ox.ac.uk/pascal/VOC/voc2012/

[126] X. Zhu and D. Ramanan, “Face detection, pose estimation, and landmark localization in the wild,” in 2012 IEEE Conference on Computer Vision and Pattern Recognition, Jun. 2012, pp. 2879–2886. doi: 10.1109/CVPR.2012.6248014.

[127] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal Loss for Dense Object Detection,” in 2017 IEEE International Conference on Computer Vision (ICCV), Oct. 2017, pp. 2999–3007. doi: 10.1109/ICCV.2017.324.

[128] D. Zeng, F. Zhao, S. Ge, and W. Shen, “Fast cascade face detection with pyramid network,” Pattern Recognit. Lett., vol. 119, pp. 180–186, Mar. 2019, doi: 10.1016/j.patrec.2018.05.024.

[129] “Masked Face Analysis.” Accessed: Mar. 15, 2024. [Online]. Available: https://imsg.ac.cn/research/maskedface.html

[130] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, “YOLOX: Exceeding YOLO Series in 2021,” Aug. 05, 2021, arXiv: arXiv:2107.08430. doi: 10.48550/arXiv.2107.08430.

[131] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “Yolov4: Optimal speed and accuracy of object detection,” ArXiv Prepr. ArXiv200410934, 2020.

[132] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “mixup: Beyond Empirical Risk Minimization,” Apr. 27, 2018, arXiv: arXiv:1710.09412. doi: 10.48550/arXiv.1710.09412.

[133] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman, “Visual tracking and recognition using probabilistic appearance manifolds,” Comput. Vis. Image Underst., vol. 99, no. 3, pp. 303–331, Sep. 2005, doi: 10.1016/j.cviu.2005.02.002.