IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 40, No. 6, December, 2023, pp. 2683-2700
Journal homepage: http://iieta.org/journals/ts

COPYNet: Unveiling Suspicious Behaviour in Face-to-Face Exams

Dogu Sirt[1*], Ediz Saykol[2]

[1] Ataturk Strategic Studies and Graduate Institute, National Defense University Rectorate, Istanbul 34334, Turkey
[2] Department of Computer Engineering, Beykent University, Ayazaga, Istanbul 34396, Turkey

Corresponding Author Email: sirtd@itu.edu.tr

https://doi.org/10.18280/ts.400629

Received: 12 June 2023
Revised: 25 August 2023
Accepted: 5 September 2023
Available online: 30 December 2023

Keywords:
abnormal behavior detection, exam copy detection, deep learning, transfer learning

ABSTRACT

This research is dedicated to the analysis and detection of anomalies within images captured by digital cameras during face-to-face examinations. The focal point is the development of a novel model designed to identify exam activities with a high degree of precision. Central to this work is the creation of the COPYNet Dataset, a substantial collection of approximately 30,000 images. This dataset is pivotal for the development, verification, and performance evaluation of anomaly detection algorithms. It is meticulously segmented into five distinct groups, each corresponding to a particular behavioral category crucial for anomaly detection. To achieve superior performance in image classification, the transfer learning method is independently hybridized with the Faster R-CNN and YOLOv5 algorithms using the pretrained ResNet model. This leads to the creation of a deep neural network framework, COPYNet, designed to generate an anomaly score by modeling typical behavior. Significantly, the COPYNet framework demonstrates remarkable precision (0.90), recall (0.88), and accuracy (0.88), marking a considerable advancement in anomaly detection compared to existing literature. The results underscore the model's capability to accurately categorize diverse activity classes, making it a promising instrument for addressing the challenge of identifying suspicious behaviors during face-to-face exams. Consequently, when the model identifies an unusual activity, it triggers an alert to be dispatched to the proctor, serving as a decision support mechanism for exam invigilators. Given the obtained success rates, our study proposes a promising solution for detecting suspicious behavior during face-to-face exams, surpassing previous studies in the field.

1. INTRODUCTION

The escalating necessity to safeguard exam integrity in face-to-face settings has triggered a surge of interest among researchers to delve into the application of deep learning techniques for the detection of suspicious behavior. By exploiting the capabilities of computer vision and behavior analysis, these techniques offer an invaluable potential to elevate the efficacy of exam monitoring systems, discourage cheating, and uphold a just and fair assessment milieu.

The realms of computer vision and deep learning have exhibited promising strides in identifying and unmasking suspicious behavior during in-person classroom exams. A myriad of methods and algorithms have been investigated by researchers to pinpoint and categorize such behaviors accurately, all in an endeavor to forge a robust framework that maintains exam integrity. This article seeks to make a seminal contribution to the existing corpus of literature by unraveling the potential of deep learning techniques in bolstering exam integrity and deterring cheating in face-to-face assessments.

There is a burgeoning body of research on exploiting computer vision and deep learning to discern suspicious behavior during classroom exams. These studies not only underscore the potential of computer vision and deep learning in detecting suspicious behavior during classroom exams but also emphasize the urgent need for further exploration to tackle the technical intricacies of these methods, such as preserving privacy and transparency and circumventing bias. Moreover, it is of paramount importance to put these systems to the test using real-world data and juxtapose their performance against traditional methods of detecting cheating.

Object tracking techniques have been the subject of extensive exploration within the realm of computer vision, aimed at tracking and monitoring the movement of objects of interest within video sequences. Comprehensive surveys have been conducted, shedding light on an array of object tracking methods and algorithms, thus illuminating the advances and challenges within this domain [1-5]. This wealth of knowledge lays a solid foundation for harnessing object tracking techniques to identify and track suspicious behavior patterns during face-to-face assessments, paving the way for the development of robust monitoring systems.

In recent years, the utility of deep learning techniques in behavior analysis and anomaly detection within crowded settings has piqued researchers' interest. In this pursuit, an innovative online, real-time crowd behavior detection methodology has been proposed, leveraging video sequences to identify and scrutinize collective behaviors within densely populated environments [6]. Additionally, several techniques have been unveiled for detecting suspicious human activity

during classroom examinations, employing a combination of computer vision techniques and behavior analysis. These studies underscore the vast potential of deep learning in dissecting complex human behaviors and spotting anomalous activities [7-10].

Various deep learning techniques have been investigated to efficaciously detect and recognize anomalous behavior within examination settings. A real-time anomalous behavior detection system has been proposed, which harnesses the power of neural networks and Gaussian distributions to oversee and identify unusual student actions during exams [11]. To bolster the accuracy of human action recognition, Convolutional Neural Networks (CNNs) are employed to exploit spatiotemporal information gleaned from video sequences [12-15]. By capturing the dynamic essence of actions, this technique enhances recognition performance, thereby enabling a more precise identification of student behavior during exams.

Faster R-CNN, a variant of convolutional neural networks based on regions, has been proposed for object detection [16, 17]. This method shares convolutional layers, thus achieving real-time object recognition, which allows for efficient monitoring and tracking of objects within examination rooms. The ability to detect and track objects in real-time is paramount to ensuring the integrity of face-to-face assessments.

- The Darknet framework has been introduced, and the YOLO (You Only Look Once) architecture has been developed as an open-source platform focused on object detection and classification tasks [18, 19]. This framework offers a flexible and efficient platform for training neural networks in surveillance systems. The YOLO architecture represents a promising methodology for accurate and efficient object detection within examination environments. An enhanced version of YOLO has been crafted to incorporate hierarchical classification, facilitating the detection and classification of a vast array of object categories [20]. This progression amplifies the capability of surveillance systems to detect and recognize a wide variety of objects, thereby enabling comprehensive monitoring and assessment in face-to-face examinations.
- Previous studies have probed various methods and techniques for the detection of cheating behavior during face-to-face examinations, such as the use of cameras, microphones, eye trackers, biometric sensors, or software tools. However, these methods have their limitations and challenges. Firstly, they necessitate the use of expensive and complex equipment or software, which may not be feasible or accessible for many educational institutions or exam centers. Secondly, they may instigate ethical and privacy concerns as they entail the collection and processing of sensitive personal data from students, such as their facial expressions, eye movements, voice patterns, or physiological signals. Additionally, these methods might be susceptible to errors or bias as they rely on predefined rules or thresholds to classify behavior as normal or suspicious, which may not encompass the diversity and complexity of human behavior.
- In response to these gaps and to overcome these challenges, this study proposes a novel framework for detecting cheating behavior during face-to-face exams. COPYNet harnesses the prowess of computer vision and deep learning to scrutinize video sequences of exam rooms and identify suspicious behavior patterns among students. COPYNet boasts several advantages over existing methods, such as:
  - It does not necessitate any additional equipment or software, other than a standard camera capable of capturing the examination room from an appropriate angle.
  - It does not collect or store any personal data from students, other than their actions and movements within the exam room. It also ensures transparency and accountability by providing explanations for its decisions.
  - It employs neural networks to learn from data and adapt to different situations, rather than relying on fixed rules or thresholds. It can also handle noise and occlusion within the video sequences.

By employing COPYNet, exam administrators and educators are empowered to monitor and assess face-to-face exams in a more effective and efficient manner, thereby ensuring the integrity and fairness of exams. Further, COPYNet can deter cheating behavior by making students aware that they are under the watchful eye of an intelligent system.

The primary objective of this study is to develop and evaluate a novel framework for detecting cheating behavior during face-to-face examinations, utilizing computer vision and deep learning techniques. The specific research questions guiding this study are:

- How can computer vision and deep learning techniques be applied to analyze video sequences of exam rooms and identify suspicious behavior patterns among students?
- How does the proposed COPYNet framework compare to existing cheating detection methods in terms of accuracy, efficiency, and robustness?
- What are the implications and contributions of the proposed framework in enhancing exam integrity and deterring cheating behavior in face-to-face exams?

By addressing these research questions, this study aims to advance the knowledge and practice of cheating detection in face-to-face exams, as well as to demonstrate the potential of computer vision and deep learning for behavior analysis and anomaly detection.

The structure of the paper is as follows:

In Section 2, a literature review on detecting exam cheating in face-to-face examinations is presented.

Section 3 introduces the COPYNet framework, our proposed solution for detecting suspicious behavior during exams, as well as the dataset used and the model we propose.

Section 4 elucidates the computer vision and object detection techniques employed in our study, including pre-processing steps and feature extraction.

Section 5 presents the performance of the computer vision and deep learning models in detecting cheating. It compares the performance of the models to existing cheating detection methods, and provides visualizations to help explain the results and underscore the insights gleaned from the analysis.

Section 6 begins by summarizing the key findings of the study. It then highlights the significance of the research and the contributions made to the field of cheating detection during classroom exams, providing suggestions for future research

and practical applications of the results. The section concludes by discussing the results in light of the research objectives and questions.

2. LITERATURE STUDY

2.1 Technology types

Apart from object detection, anomaly detection in surveillance videos is a critical aspect of maintaining exam integrity. It is reviewed modeling representations of videos for anomaly detection using deep learning techniques [21] while presenting intelligent video surveillance techniques for crowd analysis using deep learning, focusing on crowd behavior understanding and anomaly detection [22]. By analyzing the collective behavior of individuals, these techniques can identify suspicious or unusual activities in crowded examination environments.

Transfer learning, a technique that leverages knowledge learned from one task or domain to improve performance in another, has also been applied in surveillance systems. A survey is held on transfer learning, discussing different approaches and methods used to transfer knowledge from one domain to another in machine learning tasks [23]. An improved deep learning method for anomaly detection in surveillance videos by leveraging transfer learning techniques is introduced [24]. By transferring knowledge from pretrained models, the performance of anomaly detection systems can be significantly enhanced, leading to more accurate identification of abnormal behaviors during face-to-face assessments.

Specifically, in the context of human activity recognition, transfer learning has shown promising results. There are other works focusing on transfer learning for detecting images and human activity recognition and exploring different techniques to enhance the performance of recognition models [25-29]. By leveraging knowledge from related tasks or datasets, transfer learning enables the effective recognition of specific activities relevant to face-to-face assessments.

To address the challenges of anomaly detection in crowded scenes, methods that utilize density heatmaps and optical flow to detect abnormal behavior in dense crowds are proposed [30-32]. This technique enables the identification of unusual events or activities that might occur in crowded places. Also, it is introduced a hybrid histogram of oriented optical flow for abnormal behavior detection in crowd scenes, leveraging optical flow information to capture motion patterns and identify abnormal activities [33].

2.2 Application domains

However, 2D and 3D deep models for action recognition are combined, incorporating depth information to enhance the performance of action recognition systems [34]. By incorporating both spatial and temporal dimensions, these models capture richer information about human actions, allowing for more precise identification of exam-related behaviors.

Spatial-temporal CNNs for anomaly detection and localization in crowded scenes are presented, leveraging spatio-temporal information to identify anomalies [35] while proposing a deep incremental slow feature analysis network for video anomaly detection that captures and analyzes incremental changes in video streams to detect anomalies [36].

To enhance anomaly detection in crowded scenes, a deep event model for crowd anomaly detection is proposed, utilizing deep learning techniques to model crowd behaviors and identify abnormal events [37]. Deep-Cascade, a cascading 3D deep neural network architecture for fast anomaly detection and localization in crowded scenes, is presented [38]. This architecture enables efficient and accurate anomaly detection by cascading multiple deep neural networks, providing a robust solution for monitoring and identifying abnormal behaviors in environments. Also, it is focused on abnormal behavior detection in videos using deep learning techniques, exploring different deep architectures and training strategies to achieve accurate anomaly detection [39].

Abnormal trajectory and event detection in video surveillance are crucial for maintaining exam integrity. A method that combines trajectory analysis and event detection to identify abnormal behaviors in surveillance videos is proposed [40].

Another work explored transfer learning across human activities using a cascade neural network architecture [41]. The proposed method learns shared representations across different activities, enhancing the performance of activity recognition in surveillance systems for exam monitoring. By leveraging transfer learning, the system can adapt to various activity patterns and improve its anomaly detection capabilities.

In real-world anomaly detection scenarios, spatial and temporal information play crucial roles. It is addressed real-world anomaly detection in surveillance videos by leveraging both spatial and temporal information [42]. However, it is proposed a method that incrementally models normal behaviors and detects anomalous activities based on deviations from the learned models. This incremental approach enables the system to adapt and learn new normal patterns over time, improving its ability to detect abnormal behaviors during exams [43].

Challenges and advancements in deep learning for image recognition are provided in [44, 45], including the difficulties faced in training deep learning models for image recognition tasks. Also, fraud detection is addressed in video recordings of exams using CNNs. The proposed CNN-based method aims to detect fraudulent activities in video recordings, ensuring the integrity of the assessment process during exams [46].

In addition to these, a hybrid deep learning model that combines CNNs and long short-term memory (LSTM) networks to improve human action recognition is introduced. This method can be instrumental in identifying abnormal actions exhibited by students during examinations [47]. To tackle the challenges posed by crowded examination environments, a sparse reconstruction cost-based approach is introduced that aids in identifying abnormal events through the representation of normal behavior [48].

Ensuring fairness in the exam environment extends beyond behavior recognition. A hierarchical system for objectionable video detection, allowing for the identification of inappropriate content at different levels, is presented [49]. Also, there are works dealing with the recognition of suspicious activities associated with cheating during exams, contributing to the maintenance of academic integrity [50, 51]. Machine learning techniques on a different dataset from ours are used in the study [52] and get meaningful results.

Another work reviews anomaly detection techniques with optical flow on the UCSD Anomaly Dataset [53] and with a similar point of view, GAN-based models are compared using

state-of-the-art methods and showcasing GAN-based models' strong performance on the same dataset [54].

Particularly during final exams, the absence of direct teacher monitoring introduces a significant potential for academic dishonesty. To address this issue, the authors [55] proposes a novel approach employing Machine Learning and LSTM (Long Short Term Memory) techniques to identify potential incidents of exam cheating.

An AI-based automated proctoring system, Proctor Net [56], utilizing face recognition, eye-gaze tracking, and mouth opening detection to identify suspicious examinee behavior, achieving an accuracy rate of 91% across diverse datasets and malpractice scenarios

While the aforementioned studies provide insights into various aspects of object tracking, abnormal behavior recognition, and behavior understanding, there is a need to bridge the gap between these domains and the specific context of face-to-face assessments.

In conclusion, this article builds upon the existing body of knowledge on object tracking, abnormal behavior recognition, activity recognition, and human behavior understanding in video surveillance to propose a novel approach for enhancing exam integrity in face-to-face assessments whose algorithmic scope is detailly compared in Table 1.

The integration of these deep learning techniques, transfer learning strategies, and anomaly detection methods in surveillance systems holds significant potential for improving the monitoring and recognition of exam-related behaviors. In the following sections, we will delve into the specific methodologies and applications of these techniques, discussing their implications for enhancing exam integrity in face-to-face assessments.

In the area of modern education, ensuring the authenticity of face-to-face exams and preventing cheating behavior has given rise to the exploration of cutting-edge technologies such as deep learning and computer vision. In this paper, we navigate through a variety of methodologies, spanning from object tracking to transfer learning, aiming to uncover the effectiveness of these technologies in identifying and understanding suspicious behaviors displayed by students during exams.

Also, it is carefully examined the workings of several techniques. A layer of sophistication is added to the detection and recognition of actions suggestive of potential cheating by the strategic deployment of transfer learning techniques, for example, which carefully observe and analyze students' movements throughout exams.

Table 1. Comparison of the suspicious behavior detection method with existing methods

Reference	Suspicious Behaviour Recognition (No: Anomaly or Activity Recognition)	Deep Learning Model (Including Transfer Learning)	Dataset Generation	Feature Engineering	Camera Resolution / Lighting
Pennisi et al. [6]	No	No	No	Yes	Good
Senthilkumar and Narmatha [7]	Yes	No	No	Yes	Good
Soman et al. [8]	Yes	No	No	Yes	Not Bad
Gowsikhaa and Abirami [9]	Yes	No	No	Yes	Good
Debnath et al. [10]	Yes	No	Yes	Yes	Good
Al Ibrahim et al. [11]	Yes	No	Yes	Yes	Not Bad
Ji et al. [12]	No	CNN	No	No	Normal
Simonyan and Zisserman [13]	No	CNN	No	No	Normal
Simonyan and Zisserman [14]	No	CNN	No	No	Normal
Zhou et al. [15]	No	CNN	No	No	N/A
Khaleghi and Moin [24]	Yes	Autoencoder	No	Yes	Not Bad
Al-azzawi et al. [25]	No	Transfer Learning	No	No	Normal
Pang [26]	No	Transfer Learning	No	No	Normal
Keçeli et al. [27]	Yes	Transfer Learning	No	No	Not Bad
Mutegeki and Han [28]	No	Transfer Learning	No	Yes	Normal
Hao et al. [30]	No	No	No	Yes	Not Bad
Lazaridis, L. et al. [31]	No	No	No	Yes	Not Bad
Kratz and Nishino [32]	No	No	No	Yes	Not Bad
Wang et al. [33]	No	No	No	Yes	Normal
Keçeli et al. [34]	No	No	No	Yes	Normal
Medel and Savakis [35]	No	CNN-LSTM	No	No	Normal
Hu et al. [36]	No	No	No	Yes	Normal
Feng et al. [37]	No	Deep GMM	No	No	Normal
Sabokrou et al. [38]	No	CNN	No	No	Normal
Wang and Xia [39]	No	CNN	No	No	Not Bad
Cosar et al. [40]	No	No	Yes	Yes	Good
Du et al. [41]	No	Transfer Learning	No	No	Normal
Sultani et al. [42]	No	Yes	Yes	No	Good
Ouivirach et al. [43]	Yes	No	No	No	Not Bad
Hu [44]	No	CNN	No	No	Good
Jaouedia et al. [47]	No	CNN-LSTM	No	No	Not Bad
Lee et al. [49]	No	No	No	Yes	Normal
Atoum et al. [50]	Yes	No	Yes	Yes	Good
Genemo [51]	Yes	CNN	No	No	Good
Ay and Karabatak [52]	No	Yes	No	No	Good
Nemade and Gohokar [53]	No	Faster R-CNN	No	Yes	Normal
Ours	**Yes**	**Faster R-CNN + Transfer Learning**	**Yes**	**Yes**	**Good**

The addition of new elements that go beyond accepted boundaries is what makes this evaluation unique. As a major breakthrough, the idea of combining multimodal data is introduced. A thorough contextual picture of student behavior is revealed by combining visual clues with aural and physiological inputs. So, it is promoted acknowledging instances of teamwork and knowledge exchange, providing a more unbiased approach to maintaining academic integrity.

In this work, it is explored the complex dynamics of human-computer interaction while acknowledging the wider ethical and human-centric consequences. In order to integrate technology with ethical issues, it is important to build interfaces that promote openness, fairness, and student comfort.

The research also provides ground-breaking ideas that rethink the ways in which cheater detection operates. "Cheating-deterrent AI-assisted assessments" mark a paradigm shift from reactive technology use to proactive monitoring, providing students with real-time feedback and so lessening the incentive to cheat. Additionally, the idea of adaptive models foresees and mitigates prospective student evasion techniques, strengthening the overall robustness of the cheating detection system.

The paper highlights the significance of pre-trained models to solve the shortcomings of depending simply on visual signals. In order to add complexity to our knowledge of behavior patterns, this necessitates including variables like past performance and learning trajectories. The demand for cross-cultural research also acknowledges that cultural norms influence cheating tendencies, necessitating culturally adapted implementations to guarantee correct results among various student populations.

It is promoted the idea of continuous integrity monitoring, extending the scope beyond exam rooms. A culture of academic honesty is promoted throughout a student's educational journey by integrating these tools into the larger educational ecosystem. A further indication of these technologies' potential value in the area of lifelong learning is their incorporation into professional certification tests and occupational exams.

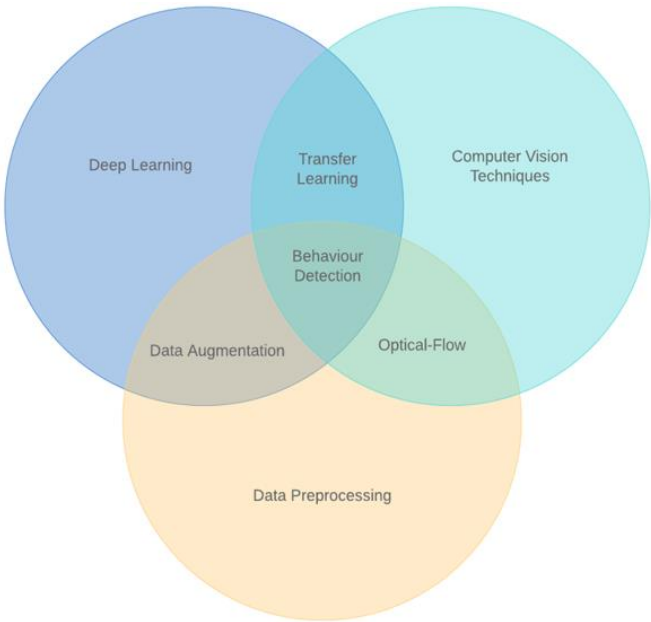

Figure 1. Intersection diagram of subjects

By raising this issue, our work combines the fields of deep learning, computer vision, exam integrity, and cheating prevention. This review plots a course for educators, researchers, and policymakers to harness the potential of technology while upholding the sanctity of academic evaluation by dissecting approaches, introducing innovative elements, and emphasizing ethical and cultural factors.

It's worth noting that while these techniques show promising results, they are still subject to various limitations and challenges. Such as, the accuracy can be affected by various factors such as camera resolution, lighting conditions, and student behavior. Moreover, as mentioned in Figure 1, there are too many intersections among these technologies, and it's important to consider them when deploying these systems. There are many other algorithms that can be used, depending on the specific requirements of the application. Also, in order to increase the accuracy of the detection, multiple algorithms can be combined.

3. THE COPYNET FRAMEWORK

3.1 The dataset

Figure 2. Images from classes B, C and D respectively

The first step is to prepare the dataset. A labeled data set containing abnormal behaviors should be created. This dataset should consist of examples of normal and abnormal behaviors. The dataset should be divided into training, validation and testing subsets. This stage is important for training the model and evaluating its performance. Our dataset (called COPYNet Dataset) is divided into 5 classes to be used in anomaly detection, with approximately 30000 images. Three of these classes are represented in Figure 2 respectively:

- Class A: Replacing Exam Paper: Instances in this class simulate situations where a student attempts to replace their own exam paper with that of another student. (Approximate number of images: 800)
- Class B: Looking at Another's Paper: Actions captured under this class involve students surreptitiously glancing at another student's exam paper. (Approximate number of images: 7000)

- Class C: Cheat Sheet Usage: This class encapsulates behaviors where students engage in cheating by referencing a concealed cheat sheet. (Approximate number of images: 9000)
- Class D: Cell Phone Usage: Instances of students using a cell phone to access online answers during the exam fall under this class. (Approximate number of images: 4200)
- Class E: Normal Exam Behavior: The baseline class portraying students engaging in the normal, non-cheating process of taking an exam. (Approximate number of images: 9000)

Table 2. Benchmark with other datasets used in previous works

OEP DATASET [50]	CUI-EXAM [51]	COPYNet DATASET
Advantages	**Advantages**	**Advantages**
1.Good classified 2.High resolution 3.Very large (11 GB)	1.Good classified 2.High resolution 3.Very large (11 GB) 4.Whole body detection	1.Good classified 2.High resolution 3.Very large (11 GB)
Disadvantages	**Disadvantages**	**Disadvantages**
1.Too large to handle 2.Only face detection	1.Too large to handle 2.Only face detection	1.Too large to handle

In Table 2, pros and cons are discussed and why our dataset in this work is chosen when compared with others.

3.1.1 Labeling and annotation by domain experts

The dataset labeling process is a pivotal aspect of its creation, ensuring that each image is accurately assigned to its respective behavior class. This labeling is meticulously carried out by expert human annotators who manually review each image, regarding the categorization on the observed behavior.

3.1.2 Pre-processing steps for data quality

To enhance the dataset's quality and utility, it undergoes a series of pre-processing steps:

- Image Enhancement: Techniques such as filters and adjustments are applied to enhance image quality and ensure clarity.
- Noise Removal: Unwanted noise, artifacts, or anomalies present in the images are meticulously removed.
- Data Cleaning: Elimination of duplicates, irrelevant images, and instances with ambiguous or inaccurate labeling.
- Normalization: Ensuring uniformity in image attributes, such as size, format, and color channels, for consistency.

3.1.3 Dataset partitioning: Training, validation, and testing

The dataset is thoughtfully partitioned into three subsets: training, validation, and testing. This partitioning facilitates robust model training, effective hyperparameter tuning, and unbiased performance assessment. Each subset plays a distinct role in the model development pipeline, ensuring the model's ability to generalize to unseen data.

In essence, the "COPYNet Dataset" serves as the cornerstone for training and assessing the performance of the proposed model in detecting anomalous behaviors during classroom exams. The meticulous composition, labeling, and pre-processing procedures ensure the dataset's reliability and suitability for meaningful analysis and conclusive results.

3.2 Applying temporal stride to the dataset

Temporal analysis is a kind of technique that can be used to analyze changes in an image or video over time, such as detecting changes in a student's posture or facial expressions.

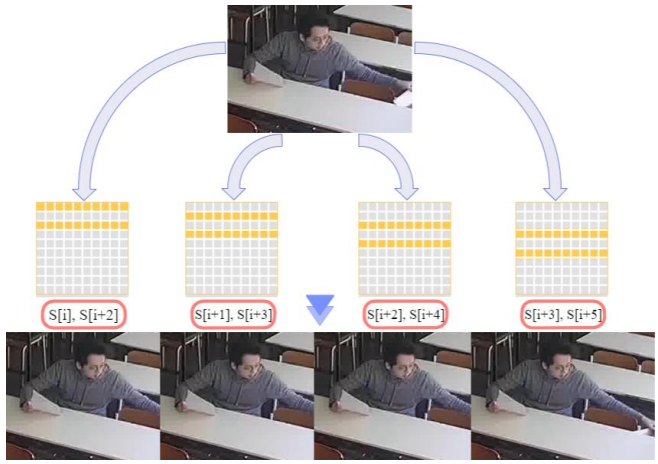

Figure 3. Temporal stride flow

Also, temporal stride is a concept used in video analysis and refers to the number of frames that are skipped when processing a video. In other words, it is the time interval between the frames that is being analyzed. A smaller temporal stride means that more frames are analyzed, resulting in a higher temporal resolution but also requiring more computational resources and time. A larger temporal stride means that fewer frames are analyzed, resulting in a lower temporal resolution but also requiring fewer computational resources and time.

In the context of temporal stride taking place in Figure 3, S[i] represents the current frame or sample in a sequence. When we refer to S[i+2], we are essentially referring to the next frame or sample in the sequence, which is shifted one step forward in time compared to S[i]. The shift between S[i] and S[i+1] allows us to capture temporal dependencies and analyze changes in the sequence over time. By examining the differences between S[i], S[i+1], S[i+2] and S[i+3] consequently, we can detect patterns or trends in the sequence, such as motion or temporal variations. As a result, S[i] serves as a reference point, and the shift to S[i+1], S[i+2], S[i+3] etc. provides insight into the temporal evolution of the sequence.

In activity recognition, the temporal stride is used to control the trade-off between computational resources and the temporal resolution of the analysis. A smaller temporal stride provides more information about the activity, but at the cost of more computational resources, while a larger temporal stride provides less information about the activity, but with fewer computational resources. The choice of the temporal stride will depend on the specific application, the available computational resources, and the desired level of accuracy.

3.3 The model

The selection of suitable models is a pivotal decision in building an effective anomaly detection system for classroom exams. In this study, the YOLOv5 and Faster R-CNN models

were chosen based on their distinct advantages that align well with the application's requirements.

3.3.1 YOLOv5: You only look once

The YOLOv5 model stands out for its unique ability to perform real-time object detection with remarkable speed and accuracy. For the context of classroom exam anomaly detection, YOLOv5's characteristics make it an ideal choice:

- Real-Time Processing: YOLOv5's ability to process images in real-time suits the dynamic nature of a classroom exam setting, enabling immediate detection of suspicious behavior without lag.
- Efficient Architecture: Its single-stage architecture simplifies the object detection process, allowing comprehensive coverage of the entire image in a single pass, which is advantageous for capturing quick and subtle cheating actions.
- Multi-Scale Detection: YOLOv5's multi-scale approach enables the detection of objects of varying sizes, making it suitable for identifying small objects like cell phones or cheat sheets that might be used for cheating.
- Object Detection: YOLOv5 can detect and categorize various objects in an image, aligning with the need to identify specific items such as cell phones or notes during exams.
- Adaptive to Different Resolutions: YOLOv5 can adapt to different input resolutions, accommodating variations in camera quality and student positions during exams.

3.3.2 Faster R-CNN: Region convolutional neural network

Faster R-CNN was also selected due to its unique attributes that align with the exam anomaly detection scenario:

- Accurate Localization: Faster R-CNN's two-stage architecture excels in precise localization of objects within an image, crucial for identifying subtle cheating actions like glancing at another's paper.
- Anchor-Based Proposal Generation: The Region Proposal Network (RPN) in Faster R-CNN generates proposals that facilitate accurate detection of objects, especially small items like cheat sheets or phones.
- Layered Architecture: The two-stage approach enables efficient feature extraction, aiding in the identification of complex cheating behaviors that may require contextual understanding.
- Detection of Multiple Objects: Faster R-CNN can simultaneously detect multiple objects, allowing the model to identify different cheating-related objects or actions occurring simultaneously.

In summary, the selection of YOLOv5 and Faster R-CNN for this application of classroom exam anomaly detection is informed by their respective strengths. YOLOv5's real-time processing and efficiency align well with the dynamic nature of exam settings, while Faster R-CNN's accurate localization and multi-object detection capabilities suit the need for precision in identifying cheating behaviors involving various objects. These models offer a balanced combination of speed, accuracy, and adaptability, making them suitable choices for this specialized application.

Our approach for detecting suspicious behavior during classroom exams uses computer vision and deep learning techniques to analyze live video feeds of the classroom. A deep learning-based object detector is trained to recognize specific actions that are indicative of cheating, such as looking at another student's exam, passing notes, or using a cell phone. The detector is then applied to the images to flag instances of suspicious behavior.

A CNN is a type of neural network that is particularly well-suited for processing data that has a grid-like structure, such as an image. In this context, a CNN could be used to analyze video footage of a classroom during an exam, looking for patterns or features that indicate suspicious behavior. For example, it could be trained to recognize when a student is looking at another student's exam paper or when a student is using a cell phone.

Two-stream CNN was first proposed by Simonyan and Zisserman [13] in which each stream consists of a series of hierarchically arranged convolutional layers for image feature extraction. Specifically, the feature extraction step is achieved through sequential convolution between the kernels at each layer and the feature maps produced in the preceding layer. For the lth layer with M input feature maps and N kernels, the jth output feature map x can be calculated as:

$$x_i^l = f\left(\sum_{i=1}^{M} x_{i^*}^{l-1} k_{ij} + b\frac{l}{j}\right) j = 1, L, N \tag{1}$$

A two-stream convolutional neural network is a type of neural network architecture that can be used to detect suspicious behavior during classroom exams. The two-stream CNN architecture [36] consists of two separate CNNs, one for processing spatial information (i.e., images or video frames) and one for processing temporal information (i.e., sequences of video frames).

The spatial stream CNN processes individual video frames and can be used to detect specific visual cues that indicate suspicious behavior, such as a student looking at another student's exam paper or using a cell phone. It can also be trained to recognize specific objects, such as a cell phone or a book, that may be used for cheating.

The temporal stream CNN processes sequences of video frames and can be used to detect patterns of behavior over time. It can be used to detect more subtle forms of cheating, such as when a student looks at another student's exam paper for an extended period of time or when a student is continuously typing on a device that is hidden from view.

The output of both streams is concatenated and then fed into a final classifier that makes a final decision about whether the behavior is suspicious or not.

Faster R-CNN is a two-stage algorithm that first generates a set of region proposals (i.e., regions of an image that may contain an object or not) In this context, it can be used to analyze video footage of a classroom during an exam to detect specific objects that may be associated with cheating, such as a cell phone or a book. As mentioned in Figure 4, the algorithm starts with detecting normal state without any suspicious activity.

Faster R-CNN works by dividing the detection process into two stages:

- The first stage is a Region Proposal Network (RPN) that generates a set of object proposals by sliding a small network over the convolutional feature maps to predict object bounds.
- The second stage is a Faster R-CNN detector that uses these proposals and classifies them based on the features of the region.

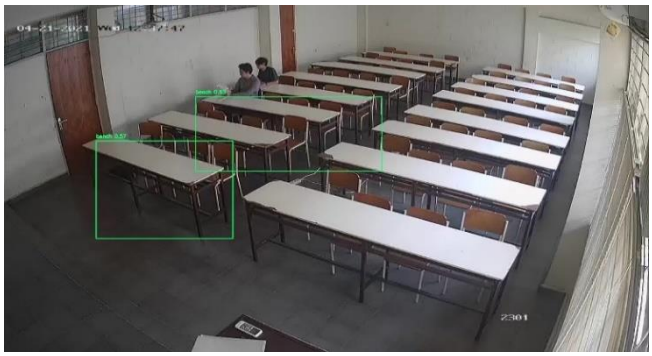

Figure 4. Identifying the normal state without cheating by drawing a bounding box with Faster R-CNN

$$L\left(\{p_i\},\{t_i\}\right)=\frac{1}{N_{cls}}\sum_i L_{cls}\left(p_i,p_i^*\right) +\lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}\left(t_i,t_i^*\right) \quad (2)$$

Here, i is the index of an anchor in a mini-batch, and p_i is the predicted probability of anchor i being an object. The ground-truth label p_i^* is 1 if the anchor is positive and 0 if the anchor is negative. ti is a vector representing the four parameterized coordinates of the predicted bounding box and t_i^* is that of the ground-truth box associated with a positive anchor. The classification loss L_{cls} is the log loss over two classes (object vs. not object). For the regression loss, we use $L_{reg}(t_i,t_i^*)=R(t_i-t_i^*)$ where R is the robust loss function (smooth L1) defined in [2]. The term $p_i^* L_{reg}$ means the regression loss is activated only for positive anchors (p_i^*=1) and is disabled otherwise (p_i^*=0). The outputs of the cls and reg layers consist of {pi} and {ti} respectively. The two terms are normalized by N_{cls} and N_{reg} and weighted by a balancing parameter λ. In our current implementation (as in the released code), the cls term in Eq. (1) is normalized by the mini-batch size (i.e., N_{cls}=256) and the reg term is normalized by the number of anchor locations (i.e., N_{reg}∼2, 400). By default we set λ=10, and thus both cls and reg terms are roughly equally weighted. We show by experiments that the results are insensitive to the values of λ in a wide range. We also note that the normalization as above is not required and could be simplified. For bounding box regression, we adopt the parameterizations of the four coordinates as follows:

$$\begin{aligned} t_x &= (x-x_a)/w_a, t_y=(y-y_a)/h_a,\\ t_w &= \log(w/w_a), t_h=\log(h/h_a),\\ t_x^* &= (x^*-x_a)/w_a, t_y^*=(y^*-y_a)/h_a,\\ t_w^* &= \log(w^*/w_a), t_h^*=\log(h^*/h_a), \end{aligned} \quad (3)$$

where, *x, y, w*, and h denote the box's center coordinates and its width and height. Variables x, x_a, and x^* are for the predicted box, anchor box, and ground truth box respectively (likewise for y; w; h). This can be thought of as bounding-box regression from an anchor box to a nearby ground-truth box.

The Faster R-CNN algorithm is particularly well-suited for detecting small objects within an image, making it well-suited for detecting cheating behaviors that might involve small objects such as phones or notes.

However, just like other techniques, this kind of implementation is still in the research phase and not yet widely used in real-world applications. It also relies on a high-quality dataset of cheating behavior, which could be hard to obtain and generalize well.

YOLOv5 uses a single convolutional neural network (CNN) to simultaneously detect objects and predict their bounding boxes in an image. Unlike Faster R-CNN, which uses a two-stage pipeline, YOLOv5 processes the entire image in one go, and this gives it the ability to process images in real-time. This could be useful in detecting objects such as a cell phone or a book that may be associated with cheating, in real-time.

The algorithm divides the image into a grid of cells, and each cell is responsible for predicting a set of bounding boxes along with their class probabilities. This makes it more efficient, allowing YOLOv5 to work in real-time on videos, which could be useful for real-time monitoring during classroom exams.

Like other techniques, YOLOv5 could be a promising solution for detecting cheating in classroom exams, but it also relies on a high-quality dataset of cheating behavior, which could be hard to obtain and generalize well. Furthermore, it also requires significant computational resources and could be more sensitive to lighting and other environmental factors.

The training procedure simply minimizes the cross-entropy loss. In this study, categorical_crossentropy is used as the loss function. Basically, categorical_crossentropy measures the distance between two probability distributions. We also used Adam Optimizer as the optimization method and accuracy as the performance metric to be tracked.

Usually, performance can be improved with data augmentation, which consists of modifying the training samples with hand-designed random transformations that do not change the semantic content of the image, such as cropping, scaling, mirroring, or color changes.

In the proposed model, suspicious activities in the exam environment can be categorized into five different classes. An architecture based on CNNs is used to solve the problem. Our network is trained by passing the Inception-v3 dataset through transfer learning.

Regardless of the feature extraction for anomaly detection and the type of anomaly detection model, the main flowchart of the anomaly detection framework is shown below. In accordance with the logical flow of the final software to be created, the steps shown in the boxes are realized one by one.

First, physical and temporal segmentation of the video is performed to extract the features of the target region that can be characterized. Then, the normal event is modeled in the training phase. In the testing phase, the abnormality of the test feature is calculated for the learned normal event model to determine whether the behavior is abnormal according to the abnormal threshold in the specified feature. Feature extraction and abnormal behavior detection modeling and classification are the two elements that have the greatest impact on the detection of abnormal behavior.

However, the Keras [57] high-level neural network API was used to develop the model with deep learning mechanisms for this thesis. The Keras library provides the flexibility to use the API in a faster and more modular way. It also supports convolutional networks to process the images/video clips that we use in our application to identify anomalous events in videos. Moreover, it guarantees that our application works with both CPUs and GPUs.

3.3.3 Residual networks

Standard CNNs that follow the architecture of the LeNet family are not easily extended to deep architectures and suffer from the vanishing gradient problem. The residual networks, or ResNets, address the issue of the vanishing gradient with residual connections, that allow hundreds of layers. They have become standard architectures for computer vision applications, and exist in multiple versions depending on the number of layers. In our work, we use the architecture of the ResNet-50 for classification as well as transfer learning.

$$x_{i+1} = \sigma\left(x_i + F\left(x_i; W_i\right)\right) \tag{4}$$

where, x_i and x_{i+1} are the input and output of the ith layer of the network, respectively. $F(x_i; W_i)$ is the non-linear residual mapping of the weight of CNN filters.

ResNet-50 starts with a 7×7 convolutional layer that converts the three-channel input image to a 64-channel image of half the size, followed by four sections of residual blocks. The output of the last residual block is 2048×7×7, which is converted to a vector of dimension 2048 by an average pooling of kernel size 7×7, and then processed through a fully-connected layer to get the final logits, here for 5 classes.

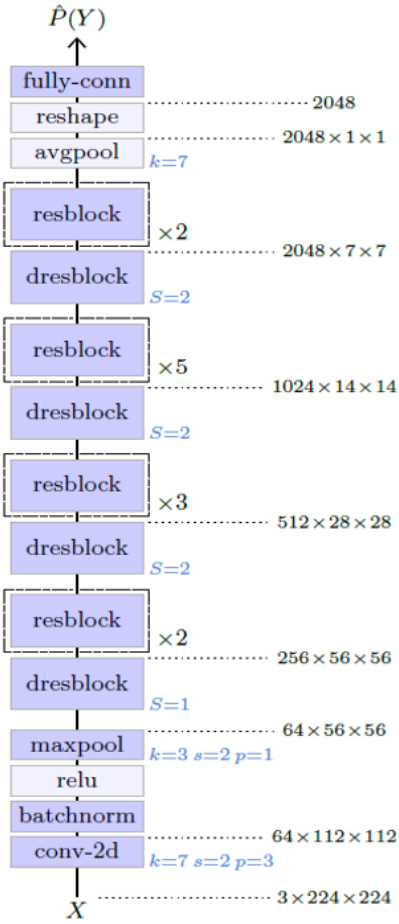

Figure 5. ResNet pretrained network

Training sets for object detection are costly to create, since the labeling with bounding boxes requires slow human intervention. To mitigate this issue, the standard approach is to start with a convolutional model that has been pre-trained on a large classification dataset such as ResNet which is shown in Figure 5 for the original SSD, and to replace its final fully connected layers with additional convolutional ones. Surprisingly, models trained for classification only have learned feature representations that can be repurposed for object detection, even though that task involves the regression of geometric quantities. During training, every ground truth bounding box is associated with its axes, and induces a loss term composed of a cross-entropy loss for the logits, and a regression loss such as Mean Square Error for the bounding box coordinates. Every other axis free of bounding-box matches induces a cross-entropy only penalty to predict the class "no object".

3.4 Model specification and training details

The chosen models, YOLOv5 and Faster R-CNN, were set up with certain settings and put through a rigorous training process in order to maximize their performance for the anomaly detection process in the classroom exam.

3.4.1 YOLOv5 configuration and training

The model was set up with the following hyperparameters for YOLOv5:

- Input Resolution: 416×416 pixels
- Batch Size: 16
- Learning Rate: 0.001
- Number of Epochs: 100
- Optimizer: Adam
- Loss Function: Mean Squared Error (MSE) for bounding box regression, Cross-Entropy for classification

A labeled dataset with about 30,000 images divided into five different groups that represent different cheating behaviors was used to start the training process. To achieve model robustness, preprocessing comprised picture normalization and augmentation.

A learning rate scheduler with exponential decay was used to modify the learning rate during training. We monitored model convergence using parameters like loss and validation accuracy. Overfitting was reduced by adding a dropout layer. To optimize convergence, the model was trained using Google Colab [58] that uses GPU acceleration.

3.4.2 Faster R-CNN configuration and training

The configuration of Faster R-CNN included the following elements:

- Backbone: ResNet-50
- Input Image Size: 600×600 pixels
- Anchor Ratios: [0.5, 1, 2]
- Batch Size: 8
- Learning Rate: 0.001
- Number of Epochs: 50
- Optimizer: Adam
- Loss Function: Cross-Entropy for classification, Mean Squared Error (MSE) for bounding box regression

The dataset received preprocessing, similar to YOLOv5, which included data augmentation and normalization. A two-stage pipeline was used throughout the training process: first, the Region Proposal Network (RPN) created candidate bounding boxes, and then the Fast R-CNN module refined the candidate bounding boxes.

During training, the learning rate was adaptively adjusted using a scheduler for learning rates. To improve model generalization, batch normalization and dropout layers were added. In order to optimize convergence, training was carried out on GPU infrastructure of Google Colab.

3.4.3 Evaluation protocol

A rigorous evaluation process was developed to determine the effectiveness of the model:

- Metrics: Common measures including precision, recall, and F1-score were used to assess the performance of the model. In order to assess detection precision across several item categories, Average Precision (mAP) and Average Intersection over Union (IoU) were computed.
- Test Dataset: To gauge how well a model generalizes to previously unreported data, a test dataset unique from the training and validation sets was employed.
- Non-Maximum Suppression: To reduce unnecessary bounding boxes and improve localization accuracy, a non-maximum suppression technique was used.
- Threshold Tuning: To balance detection sensitivity and specificity, an ideal confidence threshold was found.
- Comparison with Baselines: In order to show that the model was superior, performance was compared to baseline techniques that were widely used in the literature.

The effectiveness of the models in spotting and categorizing suspicious activities during classroom exams was measured by strictly following this evaluation process. This thorough evaluation guarantees the applicability and reliability of the suggested models in actual exam conditions.

3.5 The framework

The software would first collect data on students' normal behavior during exams, such as their facial expressions and head movements. This data would be collected using bullet cameras.

Next, the software would use machine learning techniques to train a model on the collected data. The model would learn to recognize patterns of normal behavior and be able to distinguish them from abnormal behavior.

During the exam, the software would use the trained model to monitor students by recording the exam, analyzing their behavior, and flagging any suspicious activity.

If abnormal behavior is detected, the software will generate an alarm, alerting the instructor or proctor to investigate further. This will provide feedback for the instructor. After the first two mistakes, the system only shows a yellow card, which means “keep him/her under control”. Action is realized after the third mistake, and then the student’s name and video footage are shared with the instructor to be able to make a decision on whether the student is going to continue the exam or not.

Based on the investigation, the software would generate a report summarizing the abnormal behavior, and the instructor or proctor would decide if cheating occurred or not.

The flowchart in Figure 6 describes the general technologies that the software would use in order to detect abnormal behavior in classroom examinations. The software starts by collecting data on students' normal behavior during exams. This data is then used to train a model that can recognize patterns of normal behavior and distinguish it from abnormal behavior. During the exam, the software uses the trained model to monitor students, flagging any suspicious activity. If abnormal behavior is detected, an alarm is generated, and the instructor or proctor can investigate further. Based on the investigation, a decision is made and a report is generated.

When a framework is designed, first of all, the data collection step should be defined. This step involves collecting video footage of classroom exams, either through cameras or other means. The footage may also be augmented with additional data sources such as audio or text data.

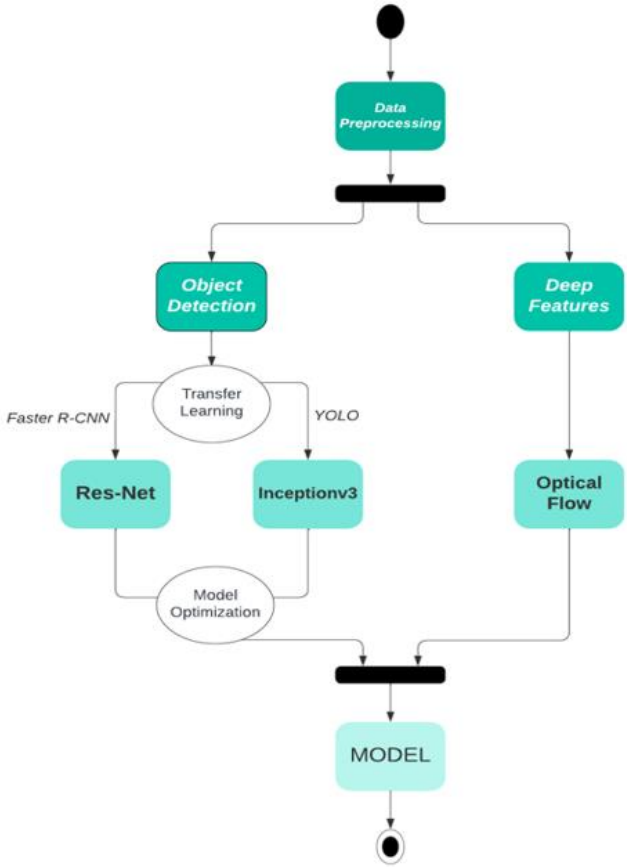

Figure 6. The COPYNet framework flowchart

Then, the preprocessing step involves preparing the data for analysis by performing tasks such as image enhancement, noise removal, and data cleaning.

After that, feature extraction’s turn comes. This step involves extracting relevant features from the data, such as color histograms, edge detection, motion history images, or optical flow. These features are then used as input for the next step.

The model training step involves training a machine learning model, such as a deep neural network, using the extracted features as input and labeled data as output. The goal of this step is to create a model that can recognize specific events or situations, such as a student cheating or using a prohibited resource.

The model evaluation step involves evaluating the trained model's performance by testing it on unseen data. The performance of the model is measured by metrics such as accuracy, precision, and recall.

The model deployment step involves deploying the trained model in a real-world setting, such as a classroom exam. The model is used to analyze live video footage in real-time and detect suspicious behavior.

The post-processing step involves performing additional tasks such as data visualization, event classification, and alert generation to notify the instructors of any suspicious behavior.

It is important to note that this is not a general framework, and the specific steps and techniques used may vary depending on the application and the available resources. Additionally, this framework shown in Figure 7 works best when it is used with the related datasets and algorithms mentioned above.

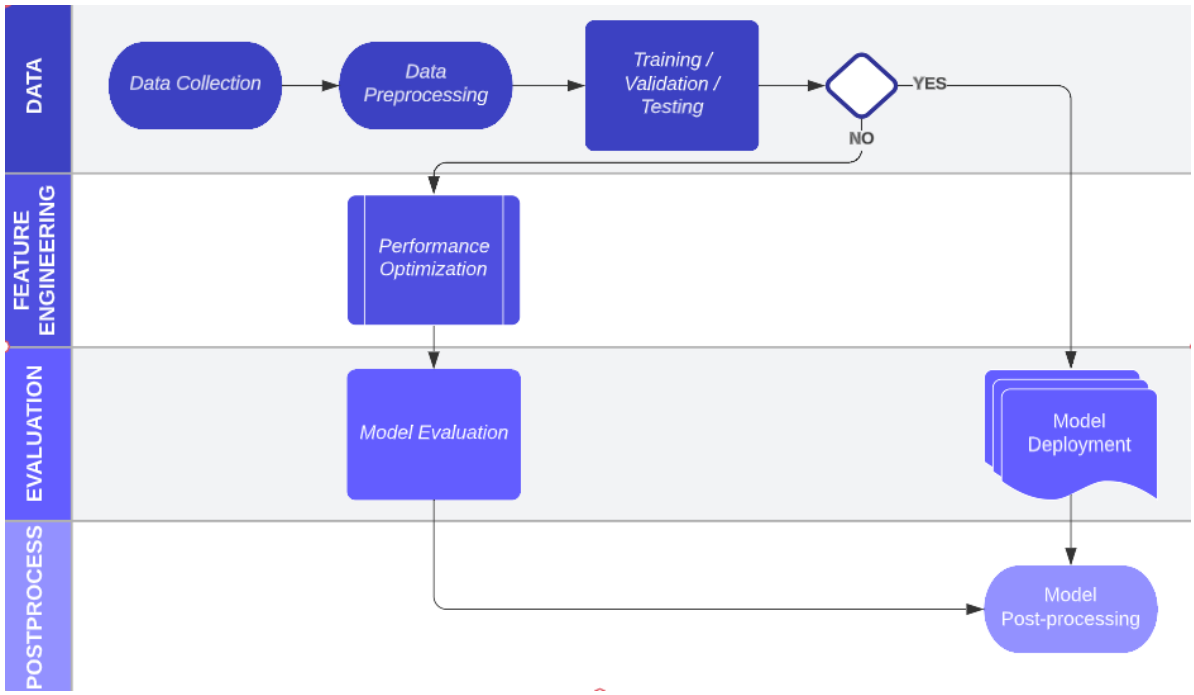

Figure 7. The COPYNet framework

4. OBJECT DETECTION TECHNIQUES

4.1 Optical flow analysis

This technique involves tracking the movement of objects or people in a video to detect changes in motion or direction. This can be useful for detecting suspicious behavior, such as a student looking away from their screen for an extended period of time. In Figure 8, a ready-made setup for Lucas-Kanade optical flow is shown.

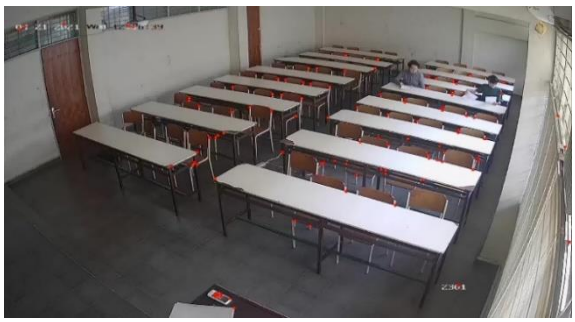

Figure 8. Classroom environment with Lucas-Kanade Optical Flow technique applied

In our work, depending on the number of features used, such as color information, edges, texture, etc., many parameters also need to be fine-tuned manually by the programmer. The Lucas-Kanade method [52] of the optical flow algorithm assumes that the pixel under study is essentially stationary in a local neighborhood. It also solves the basic optical flow equations for all pixels in that neighborhood using the least squares method.

$$I_x u + I_y u + I_t \tag{5}$$

Eq. (2) is a formula for calculating optical flow, where (u, v) is unknown and depends on the image gradients f_x and f_y as well as the time gradient ft. Two unknown elements make it difficult to solve a single equation. This issue can be solved in a number of ways. The Lucas-Kanade (LK) approach is one of them. The LK technique takes into account 3×3 chunks surrounding the locations with similar motion based on the second supposition:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \sum_i f_{x_i}^2 & \sum_i f_{x_i}^2 f_{y_i} \\ \sum_i f_{x_i} f_{y_i} & \sum_i f_{y_i}^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_i f_{x_i t_i} \\ -\sum_i f_{y_i} f_{t_i} \end{bmatrix} \tag{6}$$

4.2 Feature engineering techniques

There are several object detection techniques that can be used to detect suspicious behavior during classroom exams, including:

- **Scale-Invariant Feature Transform (SIFT):** SIFT is a feature extraction method that can be used to detect and match objects in images despite changes in scale or viewpoint.
- **Speeded Up Robust Feature (SURF):** SURF is a feature extraction method that is similar to SIFT but is faster and more robust. A version used in our study is shown in Figure 9.
- **Features from Accelerated Segment Test (FAST):** FAST is a feature extraction method that can be used to quickly detect corners in an image.
- **Multi-Scale Oriented Gradient (MSOG):** MSOG is a feature extraction method that can be used to detect edges in an image at multiple scales.
- **Single Shot MultiBox Detector (SSD):** SSD is a real-time object detection method that can be used to detect objects in an image or video. A version used in our study is shown in Figure 10.

Figure 9. Object detection with SURF

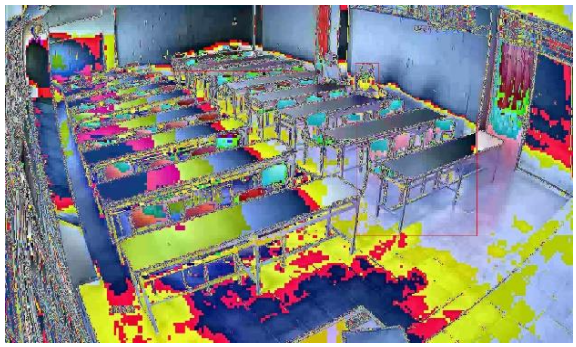

Figure 10. Object detection with SSD

Table 3. Model scores of different neural nets

Entity Number	YOLOv5				Faster R-CNN			
	Class Number	Class Name	Model Score	Confidence Interval	Class Number	Class Name	Model Score	Confidence Interval
0	0	person	0.734	(0.720, 0.750)	0	person	0.812	(0.800, 0.824)
1	13	bench	0.702	(0.680, 0.720)	13	bench	0.762	(0.748, 0.776)
2	0	person	0.622	(0.600, 0.640)	0	person	0.718	(0.704, 0.732)
3	13	bench	0.598	(0.580, 0.620)	13	bench	0.696	(0.682, 0.710)
4	13	bench	0.553	(0.530, 0.570)	13	bench	0.648	(0.634, 0.662)
5	56	chair	0.464	(0.440, 0.480)	56	chair	0.572	(0.558, 0.586)
6	13	bench	0.462	(0.440, 0.480)	13	bench	0.568	(0.554, 0.582)
7	56	chair	0.393	(0.370, 0.410)	56	chair	0.498	(0.484, 0.512)

5. PERFORMANCE STUDY

During anomalous behavior detection, it is important to evaluate the performance of the model. This can be done using metrics such as accuracy, precision, recall and F1 score. However, it is important to consider how successfully the model detects the targeted abnormal behaviors and how low the false alarm rate is.

Anomalous behavior detection using the YOLOv5 framework and transfer learning produced results within the confidence interval in Table 3 and was able to accurately bounding box people or objects in the image specified in Figure 11 and Figure 12 and identify the anomalous behavior. This approach allows the model to learn deeper and more general features and speeds up the training process.

COPYNet has success rates at the levels shown in Table 3. Compared to previous studies in the literature, the model has an important place in terms of usability. In addition, the fact that the model is both lightweight and promising in terms of success rate shows that it is open to improvement in future studies.

The use of deep features and the results obtained from their use in combination with data augmentation methods have strengthened the COPYNet framework and created the most important catalyst effect in increasing the final performance.

The bar chart represents the occurrences of different class names in the dataset. Each class name is associated with a specific class number. The x-axis of the chart shows the class names, while the y-axis represents the count of occurrences for each class name.

The chart provides a visual representation of the distribution of class names in the dataset. It allows us to quickly identify the most frequent class names and observe any imbalances or biases present in the data. By examining the heights of the bars, we can compare the occurrence of different classes and gain insights into the data composition.

This bar chart helps in understanding the relative frequencies of different classes in the dataset, which can be valuable for tasks such as object detection or classification. It provides a concise summary of the class distribution, aiding in data analysis and decision-making during model development.

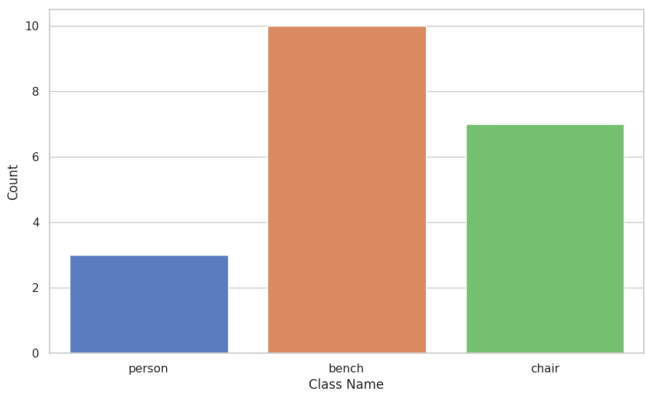

Figure 11. Occurrences of class names

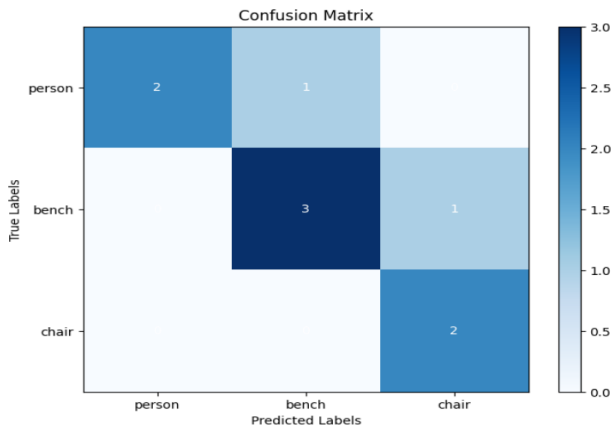

Figure 12. Confusion matrix of class names

Table 4. Computational performance of different models

MODELS	VALUES							
	mAP	Average IoU	Recall	Precision	F1-score	Loss	Accuracy	Model Inference Time (ms)
Optical Flow (YOLOv5)	0.74	0.65	0.78	0.73	0.75	3.56%	0.72	23
Feature Engineering and Transfer Learning (YOLOv5)	0.76	0.68	0.80	0.76	0.78	2.52%	0.75	25
Faster R-CNN and Transfer Learning (YOLOv5)	0.72	0.63	0.76	0.70	0.73	1.48%	0.78	33
COPYNet (YOLOv5)	**0.78**	**0.70**	**0.82**	**0.78**	**0.80**	**0.62%**	**0.80**	**35**
Optical Flow (Faster R-CNN)	0.83	0.76	0.84	0.82	0.83	3.04%	0.82	24
Feature Engineering and Transfer Learning (Faster R-CNN)	0.85	0.78	0.86	0.88	0.87	2.04%	0.84	27
Faster R-CNN and Transfer Learning (Faster R-CNN)	0.87	0.80	0.88	0.89	0.88	1.6%	0.86	29
COPYNet (Faster R-CNN)	**0.89**	**0.82**	**0.90**	**0.88**	**0.89**	**0.8%**	**0.88**	**36**

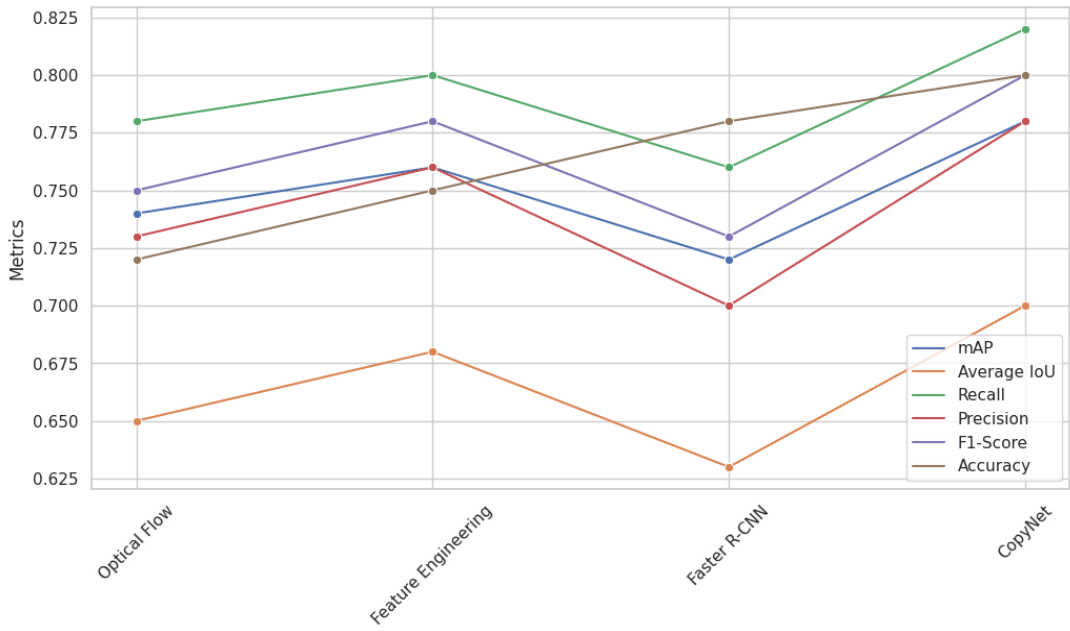

Figure 13. Benchmark graphic for evaluation metrics (YOLOv5)

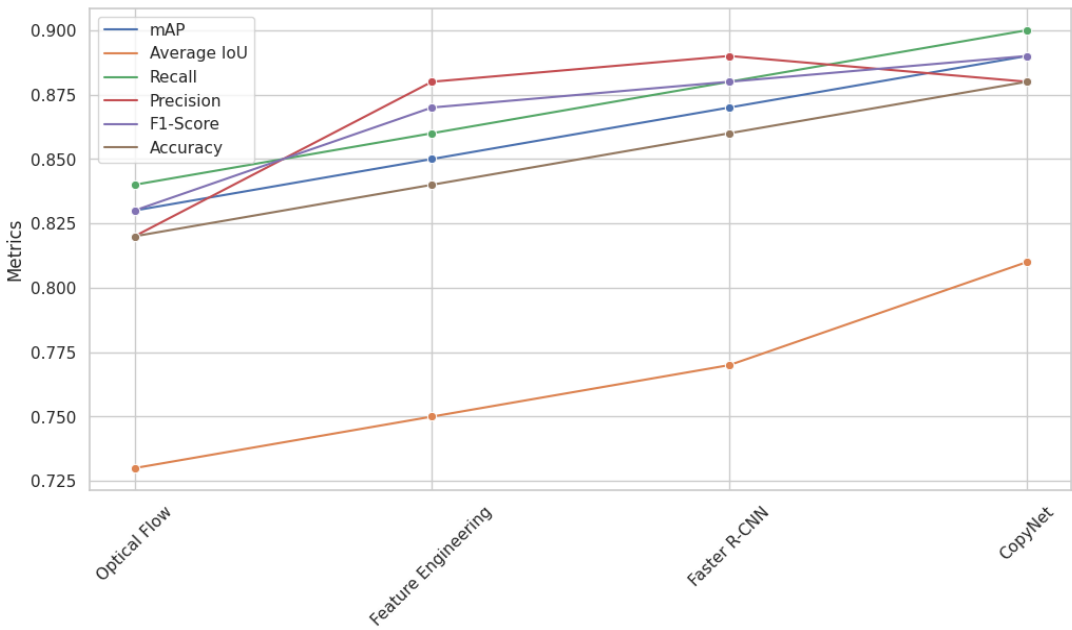

Figure 14. Benchmark graphic for evaluation metrics (Faster R-CNN)

As evaluating the performance of different models, we use mean Average Precision (mAP), Average IoU, Recall, Precision, F1-Score, Loss, and Accuracy as shown in Table 4.

The first model examined is YOLOv5 with Optical Flow. It demonstrates satisfactory performance in object detection, as measured by the mAP metric. The model utilizes Optical Flow, which aids in capturing object motion information. The results indicate a balance between recall and precision, as reflected by the F1-score. The model's accuracy and loss values provide an overall measure of its performance.

Next, we explore YOLOv5 with feature engineering and transfer learning. This configuration leverages additional techniques, such as feature engineering and transfer learning, to improve object detection performance. The model aims to enhance precision and recall values, resulting in an improved F1-score. The accuracy and loss values provide insights into the model's overall performance compared to other configurations.

Another configuration evaluated is YOLOv5 with Faster R-CNN and Transfer Learning. This combination utilizes the strengths of both the YOLOv5 and Faster R-CNN models, along with transfer learning techniques. The goal is to achieve better object detection accuracy by leveraging pre-trained models and sharing knowledge across domains. The evaluation metrics which could be seen on Figure 13 provide an understanding of the model's performance, including recall, precision, F1-Score, accuracy, and loss.

We also examine YOLOv5 with COPYNet, which incorporates the COPYNet architecture into YOLOv5. COPYNet is designed to improve object detection performance by better handling object occlusions and complex scenes. The model's performance is measured using various metrics, including mAP, average IoU, recall, precision, F1-Score, accuracy, and loss as shown in Figure 13.

Moving on, we consider Faster R-CNN with optical flow. This configuration combines the strengths of the Faster R-CNN model with the integration of optical flow. By utilizing optical flow, the model can capture object motion and improve object detection performance. The evaluation metrics offer insights into the model's performance in terms of recall, precision, F1-Score, accuracy, and loss.

We also explore Faster R-CNN with Feature Engineering and Transfer Learning. This configuration incorporates additional feature engineering techniques and transfer learning to enhance object detection performance. By leveraging pre-trained models and incorporating domain-specific knowledge, the model aims to achieve improved accuracy and precision. The evaluation metrics provide a comprehensive understanding of its performance.

Lastly, we examine COPYNet. This configuration utilizes the Faster R-CNN model in conjunction with transfer learning techniques, aiming to achieve superior object detection performance. The model leverages knowledge transfer from pre-trained models to enhance accuracy, precision, and recall. The evaluation metrics offer insights into its performance, including F1-Score, accuracy, and loss as shown in Table 4.

Overall, this performance study provides an overview of various models and configurations, showcasing their respective strengths and limitations in the field of object detection.

Figure 14 represents the performance of the Faster R-CNN model across various evaluation metrics. The graphic provides a visual overview of the model's capabilities and allows for easy comparison of its performance across different metrics.

Each line in the graphic represents a specific evaluation metric, showcasing how the metric value changes across different scenarios or experiments. The x-axis of the graphic typically represents the different scenarios or iterations, while the y-axis represents the values of the evaluation metrics.

By examining the lines in the benchmark graphic, one can observe the relative performance of the Faster R-CNN model across different metrics. Changes in the slope or trend of a line indicate improvements or variations in the model's performance, while the overall height of the lines reflects the absolute values of the evaluation metrics.

Figure 14 serves as a visual aid for evaluating and comparing the effectiveness of the Faster R-CNN model. It allows researchers and practitioners to quickly assess the strengths and weaknesses of the model in different scenarios and make informed decisions based on the performance metrics presented.

Overall, it offers a concise and visually appealing representation of the performance of the Faster R-CNN model, aiding in the understanding and interpretation of its evaluation metrics.

5.1 Model failure cases and limitations

While examining our system, it is faced with model failure cases and limitations that demonstrate the need for careful evaluation.

First of all, in object detection using YOLOv5, the model might occasionally classify non-cheating behaviors as cheating actions. For instance, a student shifting their position could be misinterpreted as cheating behavior due to similarities in motion patterns. This indicates that the model's motion-based features might not be robust enough to distinguish between such actions accurately.

During transfer learning with Feature Engineering and Transfer Learning using Faster R-CNN, the model might show exceptional performance on the training dataset, but it may struggle when exposed to unseen data from a real exam scenario. Overfitting to the training data might lead to poor generalization and suboptimal performance in real-world situations.

COPYNet using YOLOv5 could excel in detecting specific cheating actions seen during training, such as looking at another student's paper. However, if a student invents a new method of cheating that was not present in the training data, the model may fail to recognize it due to a lack of representative samples. This highlights the challenge of designing a dataset that encompasses all possible cheating scenarios.

Optical Flow based models might struggle in environments with poor lighting conditions or complex backgrounds. In dimly lit classrooms, the model's accuracy might decrease, leading to an increased number of false positives or false negatives.

Faster R-CNN models might struggle to detect very small objects, such as notes written on tiny pieces of paper. Due to the size of the objects and limited resolution, these objects might not meet the model's detection threshold, resulting in missed instances of cheating.

All models may struggle to understand the broader context of an exam environment. For instance, if a student is speaking to themselves while trying to remember something, the model might mistakenly flag it as cheating. This showcases the models' inability to comprehend the nuances of human

behavior.

Also the models' performance might be affected by the lack of diversity in the training dataset. For instance, if the dataset predominantly features cheating instances involving male students, the model's performance might degrade when applied to female students, revealing a gender bias in the model's predictions.

As ethical considerations, the models might flag behaviors that are not cheating but are rather related to personal habits or medical conditions. For example, a student's frequent head movements might be due to a health issue, leading to ethical concerns regarding privacy and discrimination. It is evaluated that having ethical constraints does not mean that our work is not applicable, but it needs to be applied carefully.

By considering these failure cases and limitations, it becomes evident that the models' performance cannot be solely relied upon. Thorough evaluation, continuous monitoring, and human oversight are essential to ensure accurate and ethical detection of cheating behaviors during exams.

6. CONCLUSION AND FUTURE DISCUSSION

Computer vision and deep learning techniques have the potential to be powerful tools for detecting suspicious behavior during classroom exams. However, it's important to consider the limitations of these technologies and their ethical implications, such as privacy concerns, when implementing such systems. Additionally, human oversight is still necessary to review flagged instances and make the final determination.

While these systems offer potential benefits, they also raise several ethical concerns that need to be carefully addressed to ensure fairness, privacy, and the well-being of individuals involved.

Monitoring students' behaviors during exams raises questions about their right to privacy. Recording video footage and analyzing students' actions can intrude on their personal space, leading to discomfort or unease. Implementing such systems without obtaining proper consent from students can infringe on their privacy rights.

Storing and processing sensitive data, such as video recordings of students, requires robust data security measures to prevent unauthorized access, hacking, or data breaches. Any compromise in data security could lead to the leakage of personal information, potentially causing harm to individuals.

AI models trained on biased or limited datasets can lead to biased outcomes. If a model disproportionately misidentifies certain groups or behaviors, it can result in unfair treatment or discrimination. Ensuring diversity and inclusivity in the training data is crucial to prevent bias in detection outcomes.

Automated detection systems may inadvertently flag innocent behaviors or misinterpret actions due to limited contextual understanding. Punishing students based on false positives can lead to unjust consequences and undermine trust in the education system.

Students and educators have the right to understand how the AI system arrives at its conclusions. Black-box models that lack transparency can be challenging to interpret and challenge, potentially resulting in frustration and mistrust.

Relying solely on AI systems without human oversight can lead to errors going unnoticed. Combining human judgment and AI detection can help mitigate false positives and ensure fair decisions.

Introducing a surveillance system in the classroom might negatively impact the learning environment by creating an atmosphere of distrust and suspicion. Students might feel anxious, leading to stress and discomfort during exams.

Educators and administrators must be well-informed about the system's capabilities, limitations, and potential biases before implementing it. Informed decisions should be made considering the educational value and potential harm.

Continuous monitoring and suspicion can have psychological effects on students. Constant surveillance might lead to feelings of intrusion, impacting their mental well-being and attitude toward education.

Implementing clear policies, transparent communication, and grievance mechanisms can help address ethical concerns. Educators and institutions should be prepared to handle cases where the system's outputs are contested. AI systems should be used as tools to support educators rather than replace their roles. Human judgment, empathy, and understanding are crucial components of effective education.

For addressing these ethical considerations, it's essential to engage stakeholders, including students, educators, parents, and experts in ethics and privacy. Implementing safeguards, transparent guidelines, and a robust feedback mechanism can help strike a balance between leveraging AI's benefits and safeguarding ethical principles in the educational context.

Computer vision and deep learning have the potential to revolutionize online proctoring by automating the detection of suspicious behavior during online exams. However, there are several challenges that need to be addressed, including the need for high-quality training data and the difficulty of distinguishing between suspicious and normal behavior. These challenges need to be addressed to ensure that this technology can be effectively and ethically used in the education sector. For instance, examinations can be photographed in different lighting conditions and resolutions, which can affect the quality of the images.

Developing methods to normalize image quality and eliminate variations is an open research problem. However, in order to have a more robust system, it could be useful to combine multiple modalities like computer vision and audio analysis, this would allow us to detect cheating not only in written exams but also in oral exams.

The techniques mentioned in our study also shed light on some future work. First of all, conducting more empirical studies to evaluate the performance and effectiveness of computer vision and deep learning techniques for detecting cheating behavior in face-to-face exams, using real-world data and scenarios. Also comparing the performance and efficiency of computer vision and deep learning techniques with traditional methods of cheating detection, such as cameras, microphones, eye trackers, biometric sensors, or software tools.

Investigating the ethical, legal, social, and educational implications of applying computer vision and deep learning techniques to face-to-face assessments, such as ensuring privacy and consent, avoiding discrimination and bias, and enhancing trust and transparency.

By synthesizing the existing literature on computer vision and deep learning techniques for detecting cheating behavior in face-to-face exams, our work aims to contribute to the advancement of knowledge and practice in this field. It also demonstrates the potential of computer vision and deep learning for behavior analysis and anomaly detection in general.

REFERENCES

[1] Yilmaz, A., Javed, O., Shah, M. (2006). Object tracking: A survey. ACM Computing Surveys (CSUR), 38(4): 13-es. https://doi.org/10.1145/1177352.1177355

[2] Popoola, O.P., Wang, K. (2012). Video-based abnormal human behavior recognition—A review. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(6): 865-878. http://doi.org/10.1109/TSMCC.2011.2178594

[3] Vishwakarma, S., Agrawal, A. (2013). A survey on activity recognition and behavior understanding in video surveillance. The Visual Computer, 29: 983-1009. http://doi.org/10.1007/s00371-012-0752-6

[4] Borges, P.V.K., Conci, N., Cavallaro, A. (2013). Video-based human behavior understanding: A survey. IEEE Transactions on Circuits and Systems for Video Technology, 23(11): 1993-2008. http://doi.org/10.1109/TCSVT.2013.2270402

[5] Tripathi, R.K., Jalal, A.S., Agrawal, S.C. (2018). Suspicious human activity recognition: A review. Artificial Intelligence Review, 50: 283-339. http://doi.org/10.1007/s10462-017-9545-7

[6] Pennisi, A., Bloisi, D.D., Iocchi, L. (2016). Online real-time crowd behavior detection in video sequences. Computer Vision and Image Understanding, 144: 166-176. http://doi.org/10.1016/j.cviu.2015.09.010

[7] Senthilkumar, T., Narmatha, G. (2016). Suspicious human activity detection in classroom examination. In Computational Intelligence, Cyber Security and Computational Models: Proceedings of ICC3 2015, Springer Singapore, pp. 99-108. http://doi.org/10.1007/978-981-10-0251-9_11

[8] Soman, N., Devi, M.R., Srinivasa, G. (2017). Detection of anomalous behavior in an examination hall towards automated proctoring. In 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), Coimbatore, India, pp. 1-6. http://doi.org/10.1109/ICECCT.2017.8117908

[9] Gowsikhaa, D., Abirami, S. (2012). Suspicious human activity detection from surveillance videos. International Journal on Internet & Distributed Computing Systems, 2(2): 141-148. https://doi.org/10.5454/JPSv1i220161014

[10] Debnath, P.P., Rashed, M.G., Das, D. (2018). Detection and controlling of suspecious behaviour in the examination hall. International Journal of Scientific & Engineering Research, 9(7): 1227-1232. https://doi.org/10.12755/ijser.2018.10.14

[11] Al Ibrahim, A., Abosamra, G., Dahab, M. (2018). Real-time anomalous behavior detection of students in examination rooms using neural networks and Gaussian distribution. International Journal of Scientific and Engineering Research, 9(10): 1716-1724. http://doi.org/10.14299/ijser.2018.10.15

[12] Ji, S., Xu, W., Yang, M., Yu, K. (2012). 3D convolutional neural networks for human action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(1): 221-231. http://doi.org/10.1109/TPAMI.2012.59

[13] Simonyan, K., Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. Advances in Neural Information Processing Systems, 27. 568-576. https://doi.org/10.48550/arXiv.1406.2199

[14] Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. https://doi.org/10.48550/arXiv.1409.1556

[15] Zhou, S., Shen, W., Zeng, D., Fang, M., Wei, Y., Zhang, Z. (2016). Spatial–temporal convolutional neural networks for anomaly detection and localization in crowded scenes. Signal Processing: Image Communication, 47: 358-368. http://doi.org/10.1016/j.image.2016.06.007

[16] Girshick, R. (2015). Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440-1448. https://doi.org/10.48550/arXiv.1504.08083

[17] Ren, S., He, K., Girshick, R., Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in Neural Information Processing Systems, 28. http://doi.org/10.1109/TPAMI.2016.2577031

[18] Redmon, J. (2016). Darknet: Open source neural networks in c. http://pjreddie.com/darknet.

[19] Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, pp. 779-788. http://doi.org/10.1109/CVPR.2016.91

[20] Redmon, J., Farhadi, A. (2017). YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, pp. 7263-7271. http://doi.org/10.1109/CVPR.2017.690

[21] Chong, Y.S., Tay, Y.H. (2015). Modeling representation of videos for anomaly detection using deep learning: A review. arXiv preprint arXiv:1505.00523. https://doi.org/10.48550/arXiv.1505.00523

[22] Sreenu, G., Durai, S. (2019). Intelligent video surveillance: A review through deep learning techniques for crowd analysis. Journal of Big Data, 6(1): 1-27. http://doi.org/10.1186/s40537-019-0212-5

[23] Pan, S.J., Yang, Q. (2009). A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 22(10): 1345-1359. http://doi.org/10.1109/TKDE.2009.191

[24] Khaleghi, A., Moin, M.S. (2018). Improved anomaly detection in surveillance videos based on a deep learning method. In 2018 8th Conference of AI & Robotics and 10th RoboCup Iranopen International Symposium (IRANOPEN), Qazvin, Iran, pp. 73-81. http://doi.org/10.1109/RIOS.2018.8406634

[25] Al-azzawi, A., Al-jumaili, S., Duru, A.D., Duru, D.G., Uçan, O.N. (2023). Evaluation of deep transfer learning methodologies on the COVID-19 radiographic chest images. Traitement du Signal, 40(2): 407-420. https://doi.org/10.18280/ts.400201

[26] Pang, J. (2018) Human activity recognition based on transfer learning, graduate theses and dissertations. https://scholarcommons.usf.edu/etd/7558.

[27] Keçeli, A.S., Kaya, A.Y.D.I.N. (2017). Violent activity detection with transfer learning method. Electronics Letters, 53(15): 1047-1048. https://doi.org/10.1049/el.2017.0970

[28] Mutegeki, R., Han, D.S. (2019). Feature-representation transfer learning for human activity recognition. In 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju,

Korea (South), IEEE, pp. 18-20. http://doi.org/10.1109/ICTC46691.2019.8939979
[29] Cook, D., Feuz, K.D., Krishnan, N.C. (2013). Transfer learning for activity recognition: A survey. Knowledge and Information Systems, 36: 537-556. http://doi.org/10.1007/s10115-013-0665-3
[30] Hao, Y., Liu, Y., Fan, J., Xu, Z. (2021). Group abnormal behaviour detection algorithm based on global optical flow. Complexity, Article ID: 5543204. https://doi.org/10.1155/2021/5543204
[31] Lazaridis, L., Dimou, A., Daras, P. (2018). Abnormal behavior detection in crowded scenes using density heatmaps and optical flow. In 2018 26th European Signal Processing Conference (EUSIPCO), Rome, Italy, pp. 2060-2064. http://doi.org/10.23919/EUSIPCO.2018.8553620
[32] Kratz, L., Nishino, K. (2009). Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In 2009 IEEE conference on computer vision and pattern recognition, Miami, FL, USA, pp. 1446-1453. http://doi.org/10.1109/CVPR.2009.5206771
[33] Wang, Q., Ma, Q., Luo, C.H., Liu, H.Y., Zhang, C.L. (2016). Hybrid histogram of oriented optical flow for abnormal behavior detection in crowd scenes. International Journal of Pattern Recognition and Artificial Intelligence, 30(2): 1655007. http://doi.org/10.1142/S0218001416550077
[34] Keceli, A.S., Kaya, A., Can, A.B. (2018). Combining 2D and 3D deep models for action recognition with depth information. Signal, Image and Video Processing, 12: 1197-1205. https://doi.org/10.1007/s11760-018-1271-3
[35] Medel, J.R., Savakis, A. (2016). Anomaly detection in video using predictive convolutional long short-term memory networks. arXiv preprint arXiv:1612.00390. https://doi.org/10.48550/arXiv.1612.00390
[36] Hu, X., Hu, S., Huang, Y., Zhang, H., Wu, H. (2016). Video anomaly detection using deep incremental slow feature analysis network. IET Computer Vision, 10(4): 258-267. http://doi.org/10.1049/iet-cvi.2015.0271
[37] Feng, Y., Yuan, Y., Lu, X. (2017). Learning deep event models for crowd anomaly detection. Neurocomputing, 219: 548-556. http://doi.org/10.1016/j.neucom.2016.09.063
[38] Sabokrou, M., Fayyaz, M., Fathy, M., Klette, R. (2017). Deep-cascade: Cascading 3D deep neural networks for fast anomaly detection and localization in crowded scenes. IEEE Transactions on Image Processing, 26(4): 1992-2004. http://doi.org/10.1109/TIP.2017.2670780
[39] Wang, J., Xia, L. (2019). Abnormal behavior detection in videos using deep learning. Cluster Computing, 22(Suppl 4): 9229-9239. http://doi.org/10.1007/s10586-018-2114-2
[40] Coşar, S., Donatiello, G., Bogorny, V., Garate, C., Alvares, L.O., Brémond, F. (2016). Toward abnormal trajectory and event detection in video surveillance. IEEE Transactions on Circuits and Systems for Video Technology, 27(3): 683-695. http://doi.org/10.1109/TCSVT.2016.2589859
[41] Du, X., Farrahi, K., Niranjan, M. (2019). Transfer learning across human activities using a cascade neural network architecture. In Proceedings of the 2019 ACM International Symposium on Wearable Computers, London United Kingdom, pp. 35-44. http://doi.org/10.1145/3341163.3347730
[42] Sultani, W., Chen, C., Shah, M. (2018). Real-world anomaly detection in surveillance videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, pp. 6479-6488. http://doi.org/10.1109/CVPR.2018.00678
[43] Ouivirach, K., Gharti, S., Dailey, M.N. (2013). Incremental behavior modeling and suspicious activity detection. Pattern Recognition, 46(3): 671-680. http://doi.org/10.1016/j.patcog.2012.10.008
[44] Hu, Y. (2020). Design and implementation of abnormal behavior detection based on deep intelligent analysis algorithms in massive video surveillance. Journal of Grid Computing, 18: 227-237. http://doi.org/10.1007/s10723-020-09506-2
[45] Thida, M., Yong, Y.L., Climent-Pérez, P., Eng, H.L., Remagnino, P. (2013). A literature review on video analytics of crowded scenes. Intelligent Multimedia Surveillance: Current Trends and Research, 17-36. http://doi.org/10.1007/978-3-642-41512-8_2
[46] Kuin, A. (2018). Fraud detection in video recordings of exams using Convolutional Neural Networks. https://5dok.net/document/lzgjr5nz-fraud-detection-video-recordings-exams-convolutional-neural-networks.html.
[47] Jaouedi, N., Boujnah, N., Bouhlel, M.S. (2020). A new hybrid deep learning model for human action recognition. Journal of King Saud University-Computer and Information Sciences, 32(4): 447-453. http://doi.org/10.1016/j.jksuci.2019.09.004
[48] Cong, Y., Yuan, J., Liu, J. (2011). Sparse reconstruction cost for abnormal event detection. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, pp. 3449-3456. http://doi.org/10.1109/CVPR.2011.5995434
[49] Lee, S., Shim, W., Kim, S. (2009). Hierarchical system for objectionable video detection. IEEE Transactions on Consumer Electronics, 55(2): 677-684. http://doi.org/10.1109/TCE.2009.5174439
[50] Atoum, Y., Chen, L., Liu, A.X., Hsu, S.D.H., Liu, X. (2017). Automated online exam proctoring. IEEE Transactions on Multimedia, 19(7): 1609-1624. http://doi.org/10.1109/TMM.2017.2656064
[51] Genemo, M.D. (2022). Suspicious activity recognition for monitoring cheating in exams. Proceedings of the Indian National Science Academy, 88(1): 1-10. http://doi.org/10.1007/s43538-022-00069-2
[52] Ay, S., Karabatak, M. (2023). A new automatic vehicle tracking and detection algorithm for multi-traffic video cameras. Traitement du Signal, 40(2): 457-468. https://doi.org/10.18280/ts.400205
[53] Nemade, N., Gohokar, V.V. (2019). Comparative performance analysis of optical flow algorithms for anomaly detection. In Proceedings of International Conference on Communication and Information Processing (ICCIP). http://doi.org/10.2139/ssrn.3419775
[54] Roka, S., Diwakar, M., Singh, P., Singh, P. (2023). Anomaly behavior detection analysis in video surveillance: A critical review. Journal of Electronic Imaging, 32(4): 042106. http://doi.org/10.1117/1.JEI.32.4.042106
[55] Alsabhan, W. (2023). Student cheating detection in higher education by implementing machine learning and LSTM techniques. Sensors, 23(8): 4149.

http://doi.org/10.3390/s23084149
[56] Tejaswi, P., Venkatramaphanikumar, S., Kishore, K.V. K. (2023). Proctor net: An AI framework for suspicious activity detection in online proctored examinations. Measurement, 206: 112266. http://doi.org/10.1016/j.measurement.2022.112266
[57] Chollet, F. (2015). keras. https://github.com/fchollet/keras.
[58] Google Colaboratory, colab.research.google.com.

NOMENCLATURE

X	input
P(Y)	output