KARADENİZ TECHNICAL UNIVERSITY
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DEPARTMENT OF COMPUTER ENGINEERING

A DEEP LEARNING-BASED FEATURE EXTRACTION AND CLASSIFICATION FOR STUDENTS ACTIVITIES IN EXAM

DOCTORATE THESIS

Musa GENEMO

NOVEMBER 2021
TRABZON

KARADENİZ TECHNICAL UNIVERSITY
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DEPARTMENT OF COMPUTER ENGINEERING

A DEEP LEARNING-BASED FEATURE EXTRACTION AND CLASSIFICATION FOR STUDENTS ACTIVITIES IN EXAM

Musa GENEMO

This thesis is accepted to give the degree of
DOCTOR OF PHILOSOPHY
By
The Graduate School of Natural and Applied Sciences at
Karadeniz Technical University

The Date of Submission : 30 /09 /2021
The Date of Examination : 05 / 11 /2021

Thesis Supervisor : Prof.Dr.Murat Ekinci

Trabzon 2021

PREFACE

This thesis is a study conducted at Karadeniz Technical University, Institute of Natural Science, Department of Computer Engineering, Ph.D. Program. In the study title " Deep Learning-based Feature Extraction and Classification for Students Activities in Exam", A framework for predicting cheating activities has been proposed in this study. This framework monitors the physical activities performed by the students during an examination. The framework detects any abnormal attempts and labels to six different classes like, back watching, front watching, Normal, side watching, showing gestures, and suspicious which is considered as illegal and relevant to cheating. Resizing the images in the whole dataset and their conversion to grayscale images, feature extraction, feature selection, and classification is integrated into this work. The study was tested on actual datasets collected from students taking an exam through the surveillance camera. The framework tested on these data and showed a promising result with an accuracy of 91.99%.

This Ph.D. has been a truly life-changing experience for me, and I would not have been able to complete it without the assistance and direction of many individuals. First and foremost, I would Like to express my gratitude to my advisor Prof.Dr.Murat Ekinci. My gratitude also extends to my family and friends for their unwavering support and encouragement during my education. My Mother, Dr.Tola Bariso Gada, Husen Abdo, and Dayma Muhammed, as well as Bayan Tamam, deserve special thanks for their continuing support.

Musa GENEMO

Trabzon 2021

THESIS ETHICS STATEMENT

This work, titled " Deep Learning-based Feature Extraction and Classification for Students Activities in Exam," which I submitted as a Ph.D. dissertation, to Karadeniz Technical University Faculty of Engineering and Natural Science. I declare that I completed the work under the supervision of my advisor, Prof. Dr.Murat EKINCI. I have fully disclosed all information obtained from other sources in the reference section, that I have acted under scientific research and ethical rules throughout the work process, and that I have accepted all legal consequences in the event of a violation. 5/11/2021

Musa GENEMO

TABLE OF CONTENTS

Page No

Ph.D. Thesis

ABSTRACT

A DEEP LEARNİNG-BASED FEATURE EXTRACTION AND CLASSIFICATION FOR STUDENTS ACTIVITIES IN EXAM

Musa GENEMO

Karadeniz Technical University
Faculty of Natural Science
Department of Computer Engineering
Advisor: Prof.Dr. Murat EKINCI
2021, 100 Page

Visual monitoring study findings have improved considerably as a result of research on video description and human activity detection. Exam cheating detection is a fundamental part of any level education program. This work focuses on students' activities labeling in the exam. The framework is developed for labeling students' activities into six different classes, back- watching, front-watching, side-watching, normal, showing gestures, and suspicious. A methodology for predicting cheating activities is proposed in this study. To extract features, feature descriptors such as local binary patterns and texture features are used. The entropy and ant colony optimization (ACO) based feature selection methods are utilized separately on the acquired feature subsets having qualities of both filter and wrapper-based approaches. The features are then combined to form a powerful features subset. Those selected features are trained on several different models. SVM-based and KNN classifiers are showed promising results on the datasets. The classification system accurately labels the student activities into abnormal and normal classifications using the exam activities detection dataset. The findings show that the proposed framework for activity recognition in exams is quite effective and accurate. 92% for SVM and 94% for KNN achieved on the dataset.

Keywords: Feature fusion, Exam onitoring, Computer vision, Activity recognition, Deep Leaning.

Doktora Tezi

ÖZETI

SINAVDA ÖĞRENCİ ETKİNLİKLERİNİN ETİKETLENMESİ İÇİN DERİN ÖĞRENME TEMELLİ ÖZNİTELİKLERİN ÇIKARILMASI VE SINIFLANDIRILMASI

Musa GENEMO

Karadeniz Teknik Üniversitesi
Fen Bilmilar Enistit üsü
Bilgisayar Mühendisliği Anabilim Dali
Danisman: Prof.Dr. Murat EKINCI
2021, 100 Page Sayfa

Video açıklaması ve insan etkinliği tanıma üzerine yapılan araştırmalar, görsel izleme konusundaki araştırma bulgularını önemli ölçüde iyileştirmektedir. Sınavda izleme etkinlikleri, öğrencilerin bir sınav odasında çeşitli etkinlikler gerçekleştirebilecekleri henüz çözülmemiş bir sorundur. Bu tür faaliyetler, otomatik bir gözetim sistemi aracılığıyla otomatik olarak izlenebilir. Derin özellik çıkarma için derin öğrenme yapıları olarak sıkma ağı ve VGG16 kullandık. Bu özellikler daha sonra tek bir özellik seti oluşturmak için seri olarak birleştirilir. Entropi ve karınca kolonisi optimizasyonu (ACO) tabanlı öznitelik seçimi yaklaşımları, hem filtre hem de sarmalayıcı tabanlı yaklaşımların niteliklerine sahip edinilmiş öznitelik alt kümelerine ayrı ayrı uygulanır. Ayrı olarak seçilen özellikler daha sonra güçlü bir özellik alt kümesi elde etmek için birleştirilir. Son olarak tahmin için SVM tabanlı sınıflandırıcılar uygulanır. Sınıflandırma algoritması, sınav etkinlikleri algılama veri kümesinden öğrenci etkinliklerini tam olarak anormal ve normal sınıflar olarak etiketler. Sonuçlar, sınavda aktivite tanıma için önerilen çerçevenin kabul edilebilir doğrulukla (%92) çok etkili olduğunu göstermektedir. Çerçeve, sınav sistemini iyileştirmek için sınavlardaki öğrenci etkinliğini analiz etmeye yardımcı olacaktır.

Anahtar Kelimelari: Özellik birleştirme, Sınav izleme, Bilgisayarla görme, Etkinlik tanıma, Derin Özellikler.

LIST OF TABLES

Page No

LIST OF FIGURES

Page No

LIST OF ABBREVIATIONS

ACC	:	Accuracy
AUC	:	Area Under Curve
C-HOG	:	Circular Histogram Oriented Gradient
CNN	:	Convolution Neural Network
CV	:	Cross Validation
HAR		Human Action Recognition
LBPHF	:	Local Binary Pattern Histogram Feature
LIFT	:	Learned Invariant Feature Transform
MKL-SVM	:	Multi kernel Support Vector Machine
PCA	:	Principal Component Analysis
PRE	:	Precision
R-HOG	:	Rectangular Histogram Oriented Gradient
RGB-D	:	Red, Green Blue Dimension
SDTH	:	Simultaneous Detection and Tracking of Humans
SEN	:	Sensitivity
SFTA	:	Segmentation based Fractal Texture Analysis
SIFT	:	Scale Invariant Feature Transformation
SVM	:	Support Vector Machine
SPE	:	Specificity
TILDE	:	Temporally Invariant Learned Detector

1.1.INTRODUCTION

1.2. Background of the Research

The quantity of photos and videos on the internet nowadays is pushing the development of search applications and algorithms that can investigate the semantic analysis of images and videos to provide better search content and summarization to users [1]. Breakthroughs have been reported by several researchers throughout the world in image labeling, object identification, and scene classification [2] [3].

This enables the formulation of approaches to problems involving object recognition and scene classification. Since artificial neural networks, particularly convolutional neural networks (CNN) [4] [5] [6], have shown a performance breakthrough in the domain of object detection and scene classification, our work focuses on determining the framework that classifies student activities into different six classes[back-watching, front-watching, side-watching, normal, showing -gestures, and suspicious].

A crucial stage in such algorithms is feature extraction. Feature extraction from photos entails extracting a small number of features from low-level image pixel values that include a lot of item or scene information, thereby capturing the differences between the object categories. Scale-invariant feature transform (SIFT) [7], histogram of oriented gradients (HOG) [8], local binary patterns (LBP) [10], Content-Based Image Retrieval (CBIR) [11], and others are some of the traditional feature extraction approaches employed on images. Once features have been extracted, they are classified depending on the items contained in the image.

An examination is a fundamental part of any education program. Monitoring exam activities in higher institutions are becoming one of the greatest challenges everywhere despite countries' development level. In recent years study on video description and human activity recognition has been constantly improving their support for visual tracking with different kind of study finds. Among this human activity detection, medical image processing, fluid detection, and security area are amongst where deep learning has strong promising results [6].

Cheating in the examination is an important issue where students practice different ways and means of cheating in the examination room. To detect and monitor the cheating activities of students, labeling student activity during their exams is very important. Preprocessing, feature extraction, feature selection, and activity categorization are included in the classification of student activities in the exam.

There is a potential that unfair means will be used in any examination, and detecting and anticipating such unfair means is problematic. The higher institution(Universities) code of conduct declares that the students must show well-mannered behavior during the examination. Exam activities are usually administratively handled at an institutional level through an honor exam code of conduct. These codes of conduct label the exam activities as honest and dishonest. This written code of conduct may vary from institution to institution based on their respective university expertization and interest. It is also very difficult to implement fairly and appropriately.

The mandate during examination is given to the students and the assigned invigilator(s). Students are required to create an environment of integrity inside the examination hall by avoiding any rules violation as of student code of conduct. Simple orientation is written and posted in the examination hall. In addition course invigilators also inform them of serious breaks that invalids their exam result, such as not holding any piece of paper, not copying from students near them, and focusing only on their works.

Therefore, it is very important to ensure that restricted activities should be avoided during examinations. Visual observation is given everywhere, today, via the web camera to constantly collect data for the recognition of numerous activities[6], decision-making, and monitoring of scenes. Smarter exam monitoring is becoming needful to mark the student's activities in the exams as normal and abnormal.

This system also helps invigilators to take unbiased action, fair judgment, and immediate action on students found cheating. So, applying visual observation using a surveillance camera to manage student activities is very crucial. Computer vision model, efficient learning algorithm, i.e. action modeling, actions and monitoring of the environment, data processing, and pattern recognition [8-11] are the focus of today's research. Figure 1.1 illustrates different scenarios during exams. Cheating in the examination has a very bad impact on the quality of education.

Since it results in poor educational outcomes, Therefore, it is very important to ensure that dishonesty should be prevented during examinations as much as possible. At any level

of educational institutions, students use dishonest means to get an unfair advantage to get higher scores or just to avoid failure in examinations.

Some other reasons compel students to use unfair means in the examination which include the pressure of their parents to succeed, insufficient educational goals, the desire to get better grades also because of many people doing it, and most of all having no fear of getting caught or getting punished for it.

In this work, a fusion of squeeze net and VGG16 model is used to find a region of interest[face and gestures) student activities from surveillance camera data. The proposed framework for the student's activity detection during exams utilizing a deep learning-based solution can completely solve the problems, especially in this COVID-19 pandemic. Temporal characteristics of action in frame series are derived using the VGG16 and Squeez net model.

Experiments are performed on students' activity recognition in exams dataset prepared from Ethiopian universities on students' activity recognition in exams. The dataset is named CUI-EXAM. The experiment outcomes depict that the suggested approach for exam activity recognition is effective in predicting different actions performed by students during exams. Representation of real scenes is a tough task because of the orientations, posture, and sizes [12].

For analysis of the study result six classes for activity recognition such as back watching, front watching, normal, showing gesture, side watching, and suspicious activity classes are defined for easy confirmation of student’s activity labeling during in exam room. Therefore, this study focus on face movement and gestures during labeling student activities.

The face movement is chosen for easy labeling of students' activities to the six classes mentioned above. Also, the face can be easily captured by the camera. Human activity recognition (HAR) is needful in many areas [13, 14]. In this study the main problems that can affect the performance of the system include camera tilts, inter and intra class likeness, scattered context, different camera views, and various changes in lighting. To solve these issues, the deep framework is utilized along with an ensemble of two feature selection methods for the resolution of exam activity monitoring. This study has the following major contributions as follows:

i. Squeeznet and VGG16 are used as CNN building blocks to extract feature extraction. Sequential fusion is used to merge the features from both networks into a single feature set.

ii. The ensemble of entropy and ant colony optimization-based feature selection methods is employed to acquire the usefulness of two types of feature subsets (filter-based and wrapper-based).

iii. Promising classification results are achieved on the prepared dataset (CUI-EXAM).

(a) (b)

Figure 1.1. Different exam scenarios. where (a) shows students are watching(Back, front, side, normal, and suspicious) and (b) shows students are exchanging code

The above two figures figure 1.1 (a) and (b) show cheating exam scenarios during examinations. Hence, there should be necessary means to catch an event of cheating to promote fairness so that students know that they won't get away with cheating in exams easily. In this work, several models have been used to train the models on the datasets. However, SVM-based and KNN classifiers are showed promising results on the datasets in which the latter shows better results over the former.

The classification system accurately labels the student activities into abnormal and normal classifications using the exam activities detection dataset. The findings show that the proposed framework for activity recognition in exams is quite effective and accurate. We observe that the combination of selected features with 1500 features (Entropy: 750+ ACO: 750 features) provides the best classification results with an accuracy of 91.99%. The satisfactory outcomes demonstrate the system's reliability. Although the results are acceptable, still the performance can further be enhanced. In the future, the performance of exam monitoring can be improved by exploring more feature selection approaches as well as deep learning methods with more big datasets.

1.3.Situations Regarding Unfair Cheating

Examination supervision and monitoring depend on the type of examination. In Ethiopia's higher education institutions(HEI) there are primarily two types of examinations, computerized and paper-based examinations. The majority of universities in the country use paper-based exams except for the two Science and Technology Universities for an entrance exam.

A computerized examination system can be online or offline which may be completed at a dedicated testing center(offline) or home(online). In some high-level exams, students are also monitored through webcams. Detecting cheating during an exam is difficult; however, to avoid cheating, a survey is conducted to assess student knowledge and attitudes toward cheating to create an automated system that monitors the activities of students during the exam. Table 1.1. illustrates the survey results from various students in different education levels engaging in cheating behaviors; examples of cheating activities are listed below.

Table 1. 1. Survey Result Obtained From Different Students

Means of Cheating	Description
Through helping material	helping material includes books, notes, formulas, and long equations that are tough to memorize, also can be on their own body or shirts, hand, or arm that the students can use in their exams.
Use of the electronic device	Students can use mobile phones, the internet, pre-saved values in scientific calculators, hands-free, and some other materials.
Involving another student	Students ask questions to other students for two reasons one when they don't know the answer and the other when not sure about the answer and use to confirm their answer.
Impersonation	In impersonation cases, another person takes the exam in the place of another's student. Superintendents sometimes doubt the student attending the exam is the one who is enrolled in the exam, students use impersonation because their ID card picture mostly differs from the current appearance.
Bribery	Some of the students use bribery to pass their exams so that the teacher ignores that student and he or she can easily cheat from the means whatever he or she wants.

The disclosure of student cheating in the examination hall is often carefully summarized under high attention supervision of invigilators to all of the students enrolled in their exams. In the real scenario, invigilators usually participate in their work including some exam requirements such as attendance maintenance, extra sheet demands, monitoring to other students, or some others to find other entertainment while other teachers may not want to spend long hours watching their students.

Furthermore, with advanced technology, whispering is not a complicated way, there are many other ways such as using mobile phones with voice recordings and photos, through hands-free chat, these all provide intelligent and direct solutions for students. In an examination hall, students must rely on the invigilator's movement and actions to determine and get the chance to cheat another student's exam.

1.4. The Objective of the Study

The primary goal of this study is to create a model that can automatically recognize student activity from video using frames provided.

1.4.1. Specific Objectives

- To achieve the above objectives the following specific objectives must be accomplished.
- To look for effective methodologies that will result in the highest rate of cheating detection.
- To create a system that will automatically detect students using the given frames.
- To create an algorithm for predicting a student's activity that has been detected.
- To improve the standard of education by eliminating cheating, bribery, and other unlawful exam attempts through a fair and accurate assessment process.
- To investigate efficient methodologies that can result in the best cheating recognition rate.
- To develop an algorithm for the prediction of detected student behavior.
- To provide the best accuracy rate for the detection of cheating during the examination.

1.5. Statements of the Problem

Many steps are performed to recognize cheating by unusual and suspicious activities of students. These steps include size normalization, feature extraction and reduction, and classification. Cofactors that affect accuracy are uniform clothing, background effects, and shadow. To maintain their value to society, academic qualifications must reflect real training outcomes. If the exam is performed in a traditional and proctored classroom environment, the students are supervised by an exam invigilator.

During the examinations, the ratio of invigilators to the students taking an examination is very low which makes their job very challenging and makes it easier for the students to cheat. On the other hand, there is no convenient way to supply invigilators to prevent cheating. Students that are seated next to each other can easily copy from each other. To prevent it a very high ratio of invigilators is needed which is typically not possible. An alternative solution to this problem would be to establish a computer vision-based automated student activity monitoring system. So, that invigilators can observe all students perfectly for any suspicious activities like front watching, back watching, side watching, paper exchange, passing cheating materials or showing gestures, and many others during the examination. Cheat monitoring is a very challenging task during the examination. To avoid creating an automated system that monitors the students' activities is the best solution.

To maintain students' value to society, academic qualifications must reflect real training outcomes. The students are supervised by an exam invigilator if the exam is taken in a traditional and proctored classroom environment. During the examinations, the ratio of invigilators to the students taking an examination is very low which makes their job very challenging and makes it easier for the students to cheat. On the other hand, there is no convenient way to supply invigilators to prevent cheating. The followings are the main statements of problem in this study:

- Illumination variation makes it difficult to recognize an activity.
- Intraclass similarity occurs because several activities are distinct, although they have a lot of the same qualities. Such as side watching but also showing gestures
- Varying viewpoint and different behavior of students.
- Occlusion's effect due to other students in the frames.
- Challenging background due to the occurrence of other students in the same frame.
- Effects of cofactors such as uniform clothing.

- Occlusions because of seating arrangement in the class can cause a problem in the accurate detection of individuals.
- Illumination factor can be occurred because of different cameras image/video acquisitions.
- Some familiar activities or performing more than one activity is also a difficult task to judge.

Cheating monitoring is a very challenging task during the examination. Most of the students remain hidden, so it is difficult to capture the student involved in some cheating activity. Therefore, strong features are required to detect and recognize the movement and activities of students doing cheating.

Since a large number of characteristics impacts the recognition rate and slows processing, choosing meaningful features is a major concern. Generally, the aforementioned difficulty occurs during preparation. To avoid creating an automated system that monitors the students' activities is the best solution.

1.6. Research Contribution

The main contribution in this research is the following:

- Providing a high-quality dataset for exam cheating monitoring will aid fellow researchers in their research. A gold standard dataset for monitoring cheating in exams which consists of 6,012 images has been developed. These images are classified into 6 categories, back watching, front watching, showing gestures, side watching, normal, and suspicious watching.
- Introduced the method of feature fusion based on features that were handcrafted such as LBP, SFTA, and Gabor for monitoring of cheating in exams.
- Improving the performance of the already developed statistical method for monitoring cheating in exams is the third contribution of this study found. The state-of-the-art statistical approach is used to track cheating in exams dataset analysis.
- The architecture of the proposed model is designed. Result evaluation made on high accuracy rate, correct classification rate, and correct recognition rate. Results will be represented in tabular and graphical form.

This study would also be beneficial in both student behavior and activity monitoring and in the examination room for reducing cheating activities.

1.7. Limitation of the Study

This research is limited to the exam phase only. The preparation phase of the exam is excluded from the study. Pre-exam orientation, sitting arrangement, student identity checking, and eligibility of students left for exam invigilator. In general, exam taker authentication is not the subject of this research. For example, requesting their identification card(ID), determining a student's exam eligibility, conducting a security screening of the students, and other issues related to the pre-exam setup. The verification of the exam taker must be confirmed by the invigilator(supervisor) of the exam.

The calibration steps to ensure that all cameras are connected and functioning properly are also confirmed by an invigilator. Further, the exam taker must be acknowledging the rules by the supervisor of the exam. Also, the sound detection must be monitored by an invigilator. This research is limited to dealing with only such activities that are supposed to be illegal during the exam.

1.8. Organization of Thesis

The organization of the thesis is based on five chapters. In the first chapter, a brief introduction of the monitoring of cheating in exams and classification of the activities held during cheating have been discussed. The second chapter holds a literature review, in which different existing methodologies are deliberated. The third chapter covers the proposed methodology of this research. The generation of datasets is also presented in this chapter itself. Both methodology and implementation are described in depth in chapter four. The fifth chapter focus on discussion, recommendations, and future works. The experimental results are given in the tables and the graphs in chapter four. Fınally the thesis ends with conclusions and recommendations.

2. LITERATURE REVIEW

This section discusses the different methodologies used to detect human activity and suspicious actions in different literature. Convolutional Neural Networks (CNN) are used in a wide range of activities and have impressive results in a wide range of applications. The recognition of handwritten digits was one of the first applications in which CNN architecture was successfully applied [17]. The addition of new layers and the usage of other computer vision techniques have enhanced CNN networks steadily since its debut [18]. Convolutional Neural Networks are mostly used in the ImageNet Challenge with various combinations of sketch datasets [19]. Few researchers have compared the detection abilities of a human subject to those of a trained network using visual data.

According to the results of the comparison, a human being corresponds to a 73.1 % accuracy rate on the dataset, whereas the results of a trained network reveal a 64 percent accuracy rate [21]. Similarly, when Convolutional Neural Networks were applied to the same dataset, they achieved a 74.9 % t accuracy, exceeding human accuracy [21]. To achieve a significantly higher accuracy rate, the deployed approaches generally make use of the strokes' order. Studies are underway to better understand the behavior of Deep Neural Networks in a variety of circumstances [20]. These experiments show how little adjustments to a picture can drastically alter grouping results. The work also includes images that are completely unrecognized by the public.

There has been a lot of development in the area of feature detectors and descriptors and many algorithms and techniques have been developed for object and scene classification. We generally enticement the similarity between the object detectors, texture filters, and filter banks. Work is abundant in the literature on object detection and scene classification [3]. Researchers mostly use the current up-to-date descriptors of Felzenszwalb and context classifiers of Hoeim [4]. The idea of developing various object detectors for basic interpretation of images is similar to the work done in multi-media communities in which they use a large number of "semantic concepts" for image and video annotations and semantic indexing [22]. In the literature that relates to our work, each semantic concept is trained by using either the image or frames of videos.

As a result, with so many jumbled things in the scene, the technique is difficult to use and understand. Previous methods concentrated on single-object detection and classification based on human-defined feature sets. These proposed methods [3] investigate the relationship between objects in scene classification. The object bank was subjected to a variety of scene classification techniques to determine its utility. Many other forms of study have been done with an emphasis on low-level feature extraction for object detection and classification, such as the histogram of oriented gradient (HOG), GIST, filter bank, and a bag of feature (BoF) implemented through word vocabulary [4].

The various methodologies used in human activities detection and action recognition during the examination for detecting and classifying student activities during the exam. The activities are detected, recognized, and classified using a variety of methods. Exam cheating is becoming a common occurrence around the globe, regardless of educational levels.

To substantiate the conclusion, available research was examined. Various human-based activity approaches that provide a clue to conduct exam activity recognition research were defined in this section. According to the authors of this study, there is a very rare work in the literature that focuses on exam activity tracking using computer vision. However, this area can be associated with other human-based activities recognition frameworks.

Despite the lack of specifically applicable work that can be used as a benchmark for this work, a thorough examination of various invigilation types and exam rules in higher education has been undertaken to solidify the study result. The student code of conduct books of ten (10) Ethiopian higher education institutions were examined to determine what acts are considered to be unlawful examination attempts.

The invigilator assignment guidelines of selected universities (the two Ethiopian universities of science and technology, five from applied universities, and three from research universities) have been carefully investigated. The number of invigilators(s) is assigned depending on the number of students sitting for an exam. The whole(room) of the exam is also considered as input to allocate several invigilators. The review below illustrates the latest approaches found in the human activity categorization.

Various researchers recognize human activity by various models like HMM model for shot boundary detection [12] as shown in Figure 2.1 below adopted from HMM model for shot boundary detection.

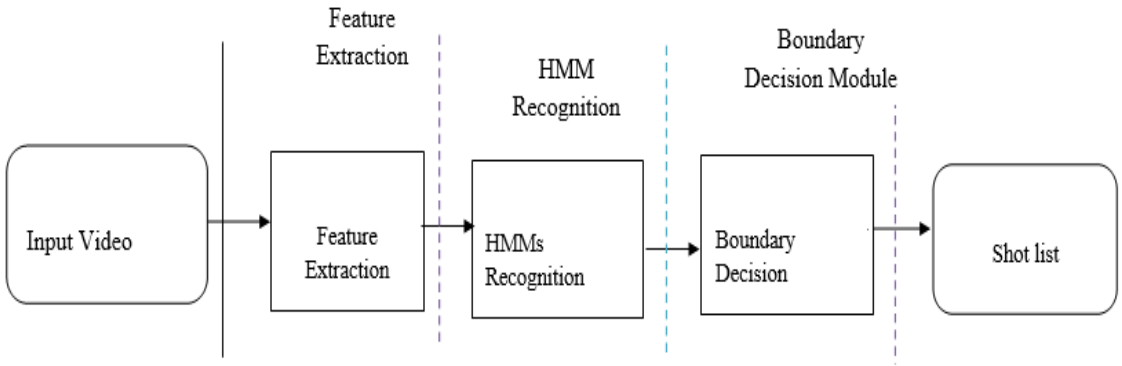

Figure 2. 1 . HMM model for shot boundary detection[32]

Early studies on action recognition relied on hand-designed features and models [36, 35]. Recently various networks have been proposed to capture both the spatial and temporal information for video classification tasks including 2D CNN-based methods [20, 42, 49], RNN-based methods [4], and 3D CNN-based methods [45, 38, 2, 46, 47, 9]. In 2D CNN-based methods, high-level information is usually captured by a 2D CNN for each frame, and various fusion techniques including early and late fusion are applied to obtain the final prediction for each video.

Various approaches are used in the related subfields such as for human detection[32-34], human behavior recognitions[30-33], and human activity detection[28-30]. Zhang et al. [41] propose a new technique detecting the human in RGB-D pictures that integrate region of Interest(ROI) creation, depth size relationship approximation, and human indicator as shown in Figure 2.2. below on the next page.

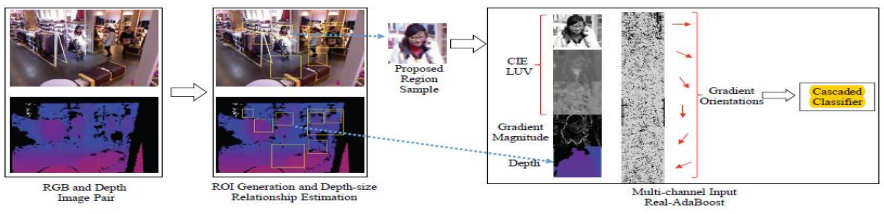

Figure 2. 2. Human Detection Approach [24]

One approach represents soccer video-based action prediction based on recurrent neural network (RNN).

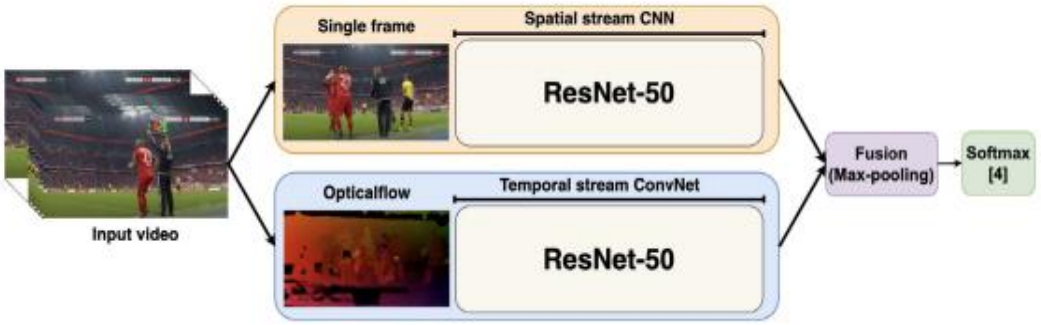

Figure 2. 3. The frames are from one category[18].

The frames from one category can be quite similar to each other, but as can be seen in the picture; there is still great variance [18]. Other works classify football video using LSTM [17-21], CNN-based learned features [8, 15, 22], and IPD. Neural Network (NN) focused methods to understand human activities [23, 24]. One approach uses video description using a bidirectional recurrent neural network as shown in figure 2.4.

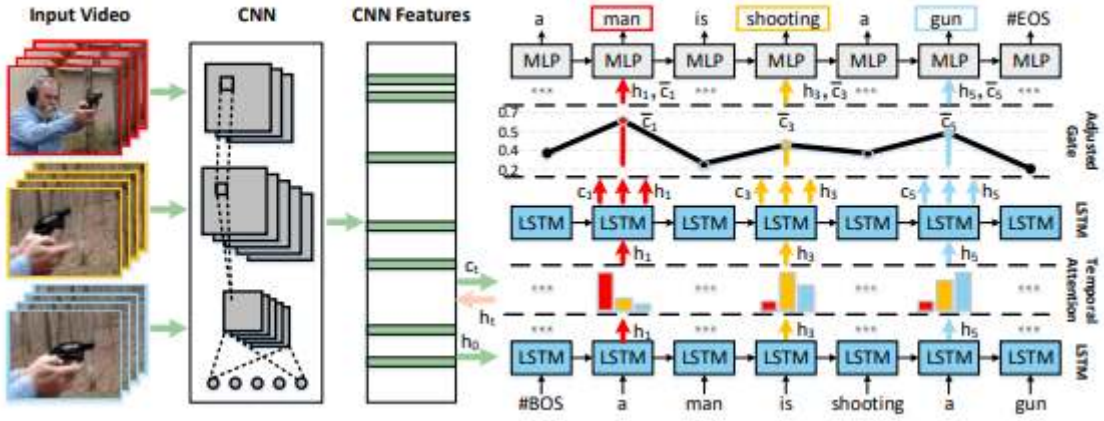

Figure 2. 4. Video description using bidirectional recurrent neural network [17].

Several authors also address different neural network models. In 2D CNN-based methods [25-27], high-level information is captured by a 2D CNN for each frame, and various fusion techniques including early and late fusion are applied. Some interesting analytical studies have recently investigated which categorize of videos require temporal information for recognition. The approaches in [28, 29] utilize the video classification

methods that exploit the appearance information of the object of interest in video to produce a highly accurate 3D classification. Given limited annotated from only 20%-50% annotated samples, the proposed approach can learn CNNs that can potentially outperform those trained in a fully supervised manner. M. Xin et al [30, 31] have performed improvement in the global features. The improvement over the supervised method, details for the generation of natural language explanations in addition to visual information in the video allows for further analysis of video labeling.

Also, there exists a lot of work in human recognition with a vital role in activity recognition [42, 43]. A method for identifying complex human activities is proposed, involving multiple participants and a composite background to provide cognitive intelligence to the computer. B. Keresztury et al. [6] strive to make machines aware of the behavior in the external environment by proposing a new way of identifying human interaction.

The goal of this research is to explore new directions for the intelligent system by embedding cognitive processes in information technology. Using the BIT-interaction-dataset, they were able to identify eight complicated human behaviors. To compare the performance of systems to other state-of-the-art classifiers, the authors made numerous feature algorithms and CNN. It is required to identify the human (student) behavior of unusual actions[44] or suspicious activities during the examination. Although many researchers pay high attention to HAR to ensure human safety and recognize the human suspicious activity for the surveillance system. In computer vision, HAR is a highly focused topic for researchers.

Scholars are also paying more attention to behavior detection. Several systems introduce for human safety purposes. As illustrated in Figure 2.5, the system is based on two stages of video processing.

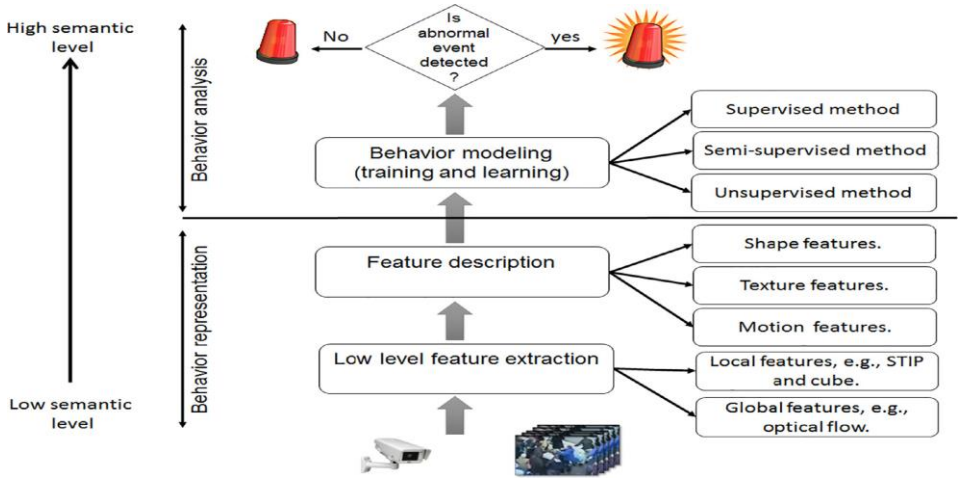

Figure 2. 5. Intelligent Surveillance System Based on Videos[34]

Figure 2.5 describes the system for abnormal behavior recognition. Which represents an intelligent surveillance system based on videos; the objective of this study is to discover an interesting event from a large number of videos to prevent dangerous situations more efficiently. J. Sung et al [25], consider the application of e-health patient rehabilitation as well. A public dataset of human activity recognition is used to explore the performance of activity recognition with their proposed algorithm.

The system of human activity recognition is developed as a framework to empower the continuous monitoring and analysis of human behaviors in different areas such as IHA and a description of behavior [19, 15-18], sports injury detection [29], patient rehabilitation [25], monitor activity shifts amid elderly citizens that might be helpful.

Similar to this work there exists several challenging applications including face detection and recognition[14-16], automatic attendance system[27, 28], criminal tracking system [19, 20], usual and unusual activity monitoring [31], objection detection[32, 13] and many others are the fields of computer vision. Therefore, this proposed work is also the contribution in computer vision field in the terms of exam activity monitoring system, which is also one of the most challenging applications of computer vision because these are some aspects that must be considered which includes occlusion and illumination variations [24], pose variations, expression variation, and all other factors.

J. Sung et al. [25] performed human activity detection in an unstructured environment. Their proposed technique is based on an RGBD sensor used to take input, and extract features based on human motion and pose variation. They make the scenario of twelve

different activities and perform well. F. Baradel et al. [16] recognize human activity from unstructured feature points.

They design a module to get visual attention that learns and performs prediction against glimpse sequences for each frame. Further, it corresponds to interesting points available in the scenario which is more relevant to classified activities. For cheating detection during the examination, various approaches can be used to accomplish the goal.

Many techniques are used in the related subfields such as for identifying and labeling normal and extraordinary human activities[21, 27-31], human behavior recognition, and human activity detection [17, 22, 23]that can also be useful for monitoring of cheating in exams. There are numerous models and descriptors available for feature extraction which is the base technique of detection of single or multiple objects from any scenario. Following are the descriptors used by researchers in their work to improve the detection and recognition rate.

2.1. Feature

Feature descriptors are used to extract the features from images. In our case, we have extracted features from the video in the RIO to label the student's activities in the exam. The main regions focused to extract features in this work are face and gesture. Based on the features extracted from the two regions(face and hand) the models label students' activities accordingly. A feature is a parameter that describes the image. It also describes the contents of the image and represents both the local and global properties of an image[24-26]. Image features can be categories into two types of features. There are features at both the low levels and high levels.

2.2. High-level Features

High-level features are used to demonstrate the abjectness of a picture with a coarse spatial position [77]. High-level features are used to define a semantic aspect of an image as per human perception [78, 79]. These features typically include machine learning models, deep models, and algorithms to imitate human perception. High-level features have a semantic meaning. For example, point to an object or the global information present or not in the image [70-72].

2.3. Low-level Features

Low-level properties represent the image in terms of spatial characteristics, color, shape, and texture features [73]. Low-level features are also known as local features, these features nowadays play a critical role in computer vision. The purpose of these features is to discriminatively extract features from a set of patterns, placed as the elements of foreground and background of an image itself [74-76].

Low-level features include the vector dimensions that have no intrinsic meaning but show the basic understanding of the image which includes the edges or color information such as color features and texture features. These elements are helpful to recognize the similar parameters between similar images [77-79]. To detect the interesting points and regions from an image and events from large videos, low-level features are considered. The primitives of low-level features are obtained to describe interest regions.

An entire image has a complex structure, which includes several objects, complex background, and a foreground. So that to extract-strongest features from images, productive component-descriptors are required [79,80]. From the above text analysis, we found that the feature descriptors can generally As illustrated in Figure 2.6, global-feature descriptors and local-feature descriptors can be separated into two types.

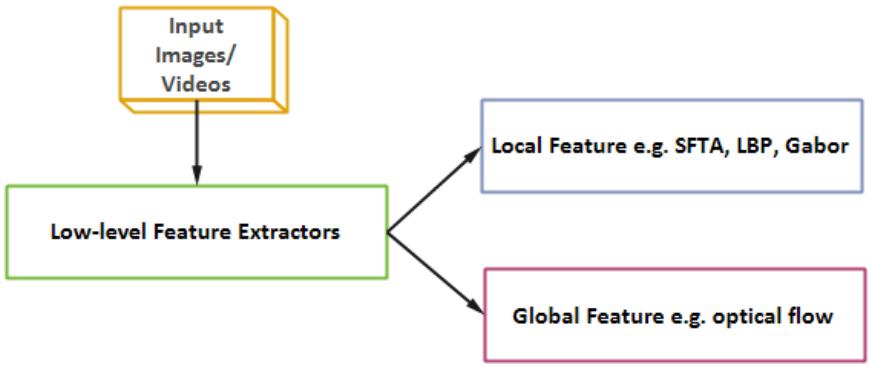

Figure 2. 6. Low-level Features Extractions [80]

The feature descriptors are used to encode features or interesting information into numerical values which can be used as feature-vector for the retrieval of information and also for the further process of the feature extraction from other feature descriptors. The more efficient the descriptor, high retravel accuracy score will be achieved.

2.4. Global Feature Descriptor

To incorporate all of an image's features, global feature descriptors are used in the works [71, 64, 75]. High-level features are used to define the semantic aspect of an image as per human perception such as the classification of satellite images [79]. These features typically include machine learning models and algorithms to imitate human perception.

2.5. Local Feature

We made use of many computer vision algorithms that rely on local features and associated descriptors, which are compact vector representations of a local region. Image registration, object identification and categorization, tracking, and motion estimation are some of their uses. These methods can better manage scale changes, rotation, and occlusion by using local characteristics.

Local image feature detectors are different from each other in a way that how each one can extract the features, e.g. points, circles, ellipses [57,58, 76]. In addition, the type of feature also determines that they can handle a class of transformations like Euclidean transformations, similarities, and affinities [49]. Local image feature detectors can be divided as follows: hand-crafted detectors and standard deep models [55, 60].

2.6. Hand-Crafted Local Features

These detections use different perceptual patterns as anchors for the features, such as corners or other image intensity operators like the Hessian of Gaussian [62] or the structure tensor [72]. Integral pictures are used to approximate the Hessian feature response in the SURF detector [73].

Training detectors aim to find or enhance the visual anchors used for detection, a far more difficult task than utilizing hand-crafted anchors. Genetic programming was applied in the early versions [57, 40]. K. Moo et al. [68] recently discovered that deep learning can be used to determine the orientation of feature points.

The TILDE detector [49] for lighting invariance is a similar method. The LIFT framework [71] employs patches to train detector, descriptor, and orientation estimation simultaneously, whereas Super Point [51] uses entire pictures. DNET[100], which is trained using the covariance constraint and no supervision, is another technique for unsupervised

learning of keypoint detectors. TCDET [73], which combines geometry and appearance losses, is a variant of this detector. The covariant constraint is extended for affine adaptation in [79, 64]. To encode features from sub-parts (small group of pixels) of an image, local feature descriptors are used. For this purpose, an image is partitioned into sub-parts and also can encode features by using segmentation followed by collecting information about edges, corners, etc.[55].

The feature descriptors often extract low-level features from an image. The most popular low-level or local feature extractor includes HOG [76], LBP [80], SFTA [31, 21], SIFT, SURF, Gabor [77] [63], Haralick [55], GLCM, ORB, PHOG and many more [46]. In the below section, we have discussed some prominently used feature descriptors to extract features in images.

2.7. Hybrid Features

Hybrid features are the same as feature fusion in which we combine different feature vectors from multiple sources [27-32]. This technique of combing feature vectors is commonly used to maximize the more relevant information about the content. Researchers used this technique to improve the discrimination capability as well as reliability and provide the opportunity of minimizing data retained. Feature fusion technique is derived from different methods to improve the classification rate [62]. There exist mainly two techniques for feature fusion: serial-based feature fusion and parallel-combination-based feature fusion as shown in Figure 2.7.

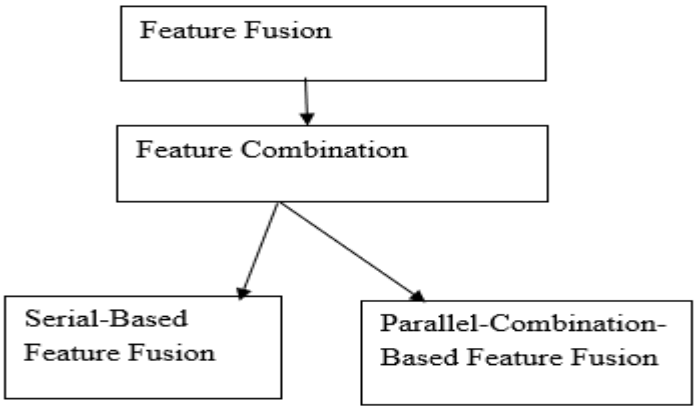

Figure 2. 7. Feature Fusion Strategy [34]

As the resultant of feature fusion, one of the best and more relevant feature subsets is returned from the original feature set. For the task of classification, we can use this resultant feature subset or also can obtain a new feature set based on multiple selected features.

2.8.Serial-Based Feature Fusion

Serial-based feature extraction involves the combinations of different feature vectors from different sources and concatenates these features together [55]. This technique involves the feature combination as the concatenation of features. If we combined two features i and j having weights of w1 and w2, respectively, according to serial-based feature-fusion the concatenated features are [ix w1, jx w2]. such as M. A. Khan et al. [76] use serial-based feature fusion in their proposed work as shown in Figure 2.8.

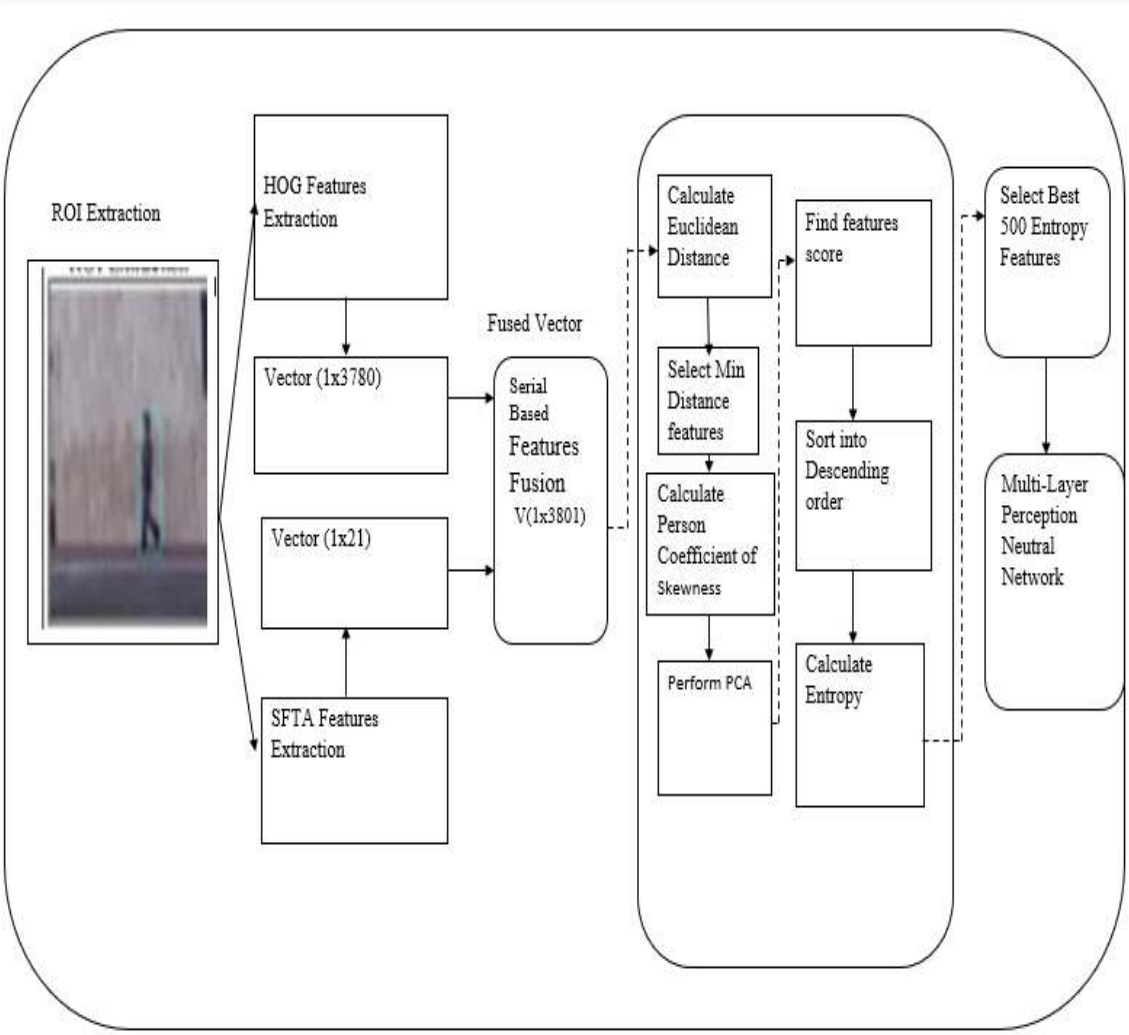

Figure 2. 8. Serial based Feature Fusion for action classification[76]

2.9. Parallel Combination Based Feature Fusion

Parallel-combination-based feature extraction involves the combinations of different feature vectors from different sources and combined these features [80]. In parallel combination of feature fusion, complex-variable is used for combing two features into complex features. The resultant absolute value of complex features can be taken as a resultant fused feature. Such as Qiuhong Ke et al. [39] use parallel-combination-based feature fusion in their work of action recognition for the representation of skeleton sequences which are represented in the given proposed model as shown in Figure 2.9.

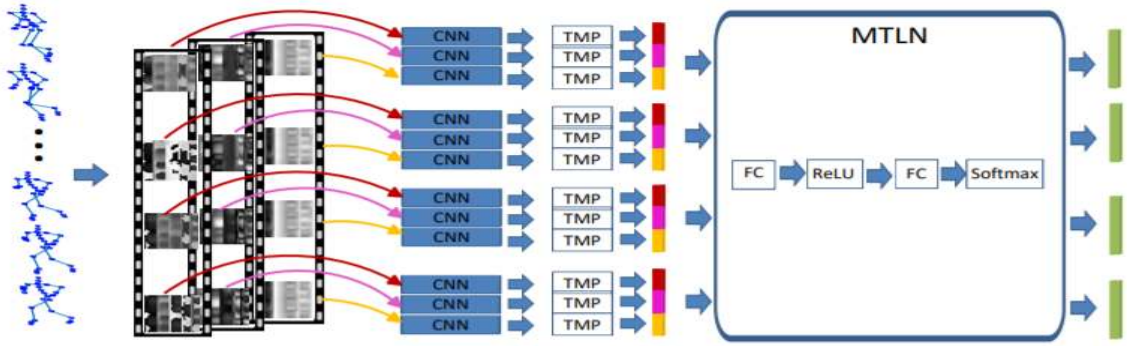

Figure 2. 9. Parallel Fusion Based Action Recognition [69]

3. PROPOSED WORK

3.1. Introduction

This section discusses how the suggested model will be implemented. A method is proposed to detect and monitor the cheating activities held by the students and to categorize into the means or types they use for cheating purposes, this study proposes a method for classification of students involved in cheating during their examination, classification will be based on suspicious activities and the ways of cheating activity held by the student during the examination.

3.2. Research Methodology

The proposed methodology is developed for the system that is based on computer vision that consists of these major steps: (1) Image resizing and grayscale conversion; (2) Feature extraction; (3) Feature selection; (4) Feature fusion for extracted features; and (5) Classification. Our approach integrates resizing the images in the whole dataset and their conversion to grayscale images, feature extraction, selection, and classification of images.

Feature descriptors such as local binary patterns and texture features are used to extract features (SFTA and Gabor). Following that, the fused features are chosen using the Principal Component Analysis approach. Finally, a support vector machine and fine KNN are used to classify the selected features. To accomplish the tasks the following procedure has been carefully done.

- The dataset has been formed.
- The collected dataset was crafted manually.
- An effective and potential algorithm has been defined.
- The proposed algorithm was implemented to gather the results.
- The result has been evaluated and compared based on performance measures.

In addition, the accuracy rate of detecting cheating, as well as the distribution into accurate classification and recognition rate under various conditions that affect recognition performance, must be enhanced. The dataset has been trained and tested according to people engaging in various sorts of cheating while taking an exam to evaluate the suggested system.

So that the suggested system's accuracy, resilience, and efficiency may be determined based on experimental data.

To get the best and unique feature focusing on extracting middle-level features from video frame are important. We extracted features of video frames like gaze and phone, looking at nearest students answer, copying from a note, and prohibited watching. Then a joint decision across all frames high-level temporal features will be extracted as all frames result combination. Therefore, this new feature is used to train the proposed system.

The proposed method detects and classifies the way of cheating, a student used in his or her exam. This study would also be beneficial in both student activity monitoring and in the examination room for reducing such types of cheating activities.

The proposed method uses the newly created dataset to evaluate its effectiveness. The design of the proposed system methodology is shown in Figure 3.1 given below:

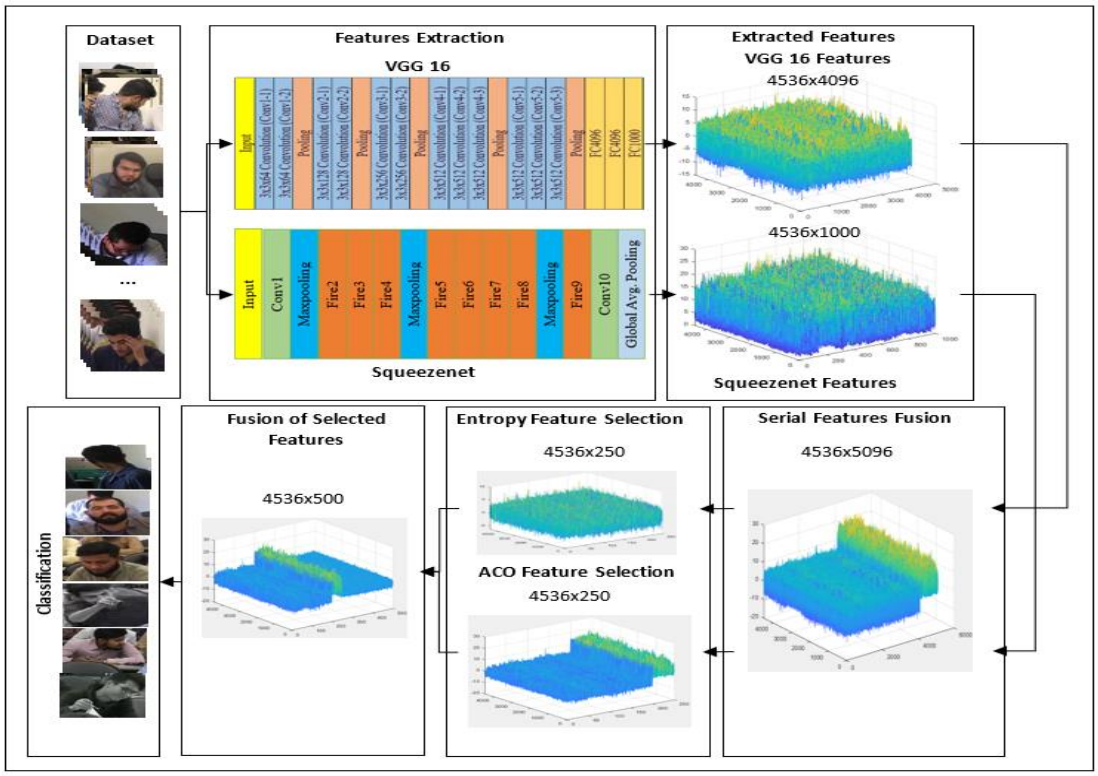

Figure 3. 1. Representation of the proposed model

The model detects and classifies the way of cheating, a student detected in his/her exam. This framework can also be used for monitoring any type of exams like recruitment exam, exit exam, COC, and others.

3.3. Feature Extraction

In the depiction of images, feature extraction is critical. Three sorts of features are extracted for this purpose: (1) SFTA (Texture Features), (2) LBP (Features encode local texture information), and (3) Gabor, and these all are efficient methods of feature extraction. These feature vectors consist of shape, texture, local texture information as well as the information about frequency and orientation representation of an image or the matrix respectively. These all are imagined to be very similar in the human visual information system. It has a great importance of efficient and accurate feature extraction and the selection of strongest features because it affects directly the results of classification. There selected 121 features that are combined or fused into three categories including SFTA with 21, LBP with 40, and Gabor with 60 features, description of each feature set are given in the following section.

3.4. Segmentation Based Fractal Texture Analysis

SFTA is also a feature-extraction descriptor used to extract texture information, these features are implemented with the basic filters. SFTA features are used to get the structure for the mass-boundaries, features vector obtains in the result of applying SFTA descriptor having 21 Region-based texture features corresponding to the information of gray levels, dimensionality and ROI's area. It returns the features in the form of decomposed binary values using the algorithm of the single threshold.

3.5. Local Binary Pattern

LBP is a useful feature extraction algorithm and provides resistance in light variations. In this work, the threshold value is assigned according to pixel values that are adjacent to the central pixel. After that, the calculations against the LBP-matrix are performed according to given values of local-neighborhood that may be in-clockwise or in anticlockwise-direction. The LBP process returns the most important features that are resistance changes in gray-level and computational-simplicity labeling of the pixels is performed. LBP descriptor extracts the features and gives a feature vector of size 1 x 59.

Some resultant images after implementing the LBP feature descriptor are shown in Figure 3.2.

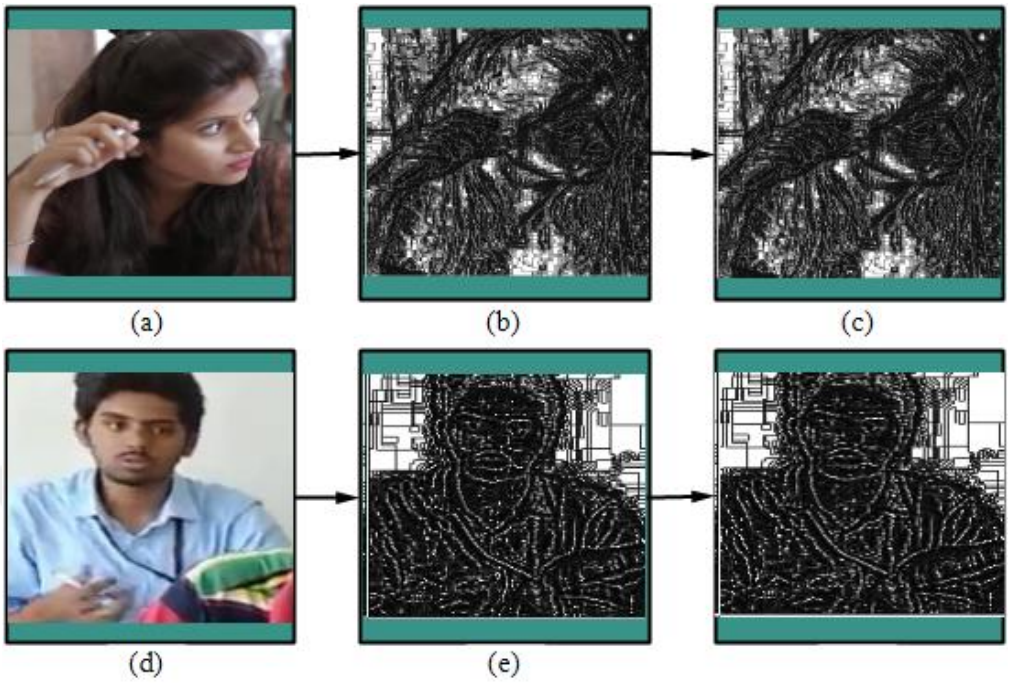

Figure 3. 2. LBP Resultant Images

Where Gabor resultant images (a) and (d) are original images of side watching and front watching respectively, image (b) and (d) are real parts of Gabor filters and (c) and (f) represent the resultant Gabor filtered Images. In this work, feature matrices containing 1x40 strongest features are obtained for each image through LBP_algorithm. Further classifiers are trained on all datasets using property metrics.

3.6. Gabor Based Features

Gabor features are commonly used to extract features at different magnitudes and different scales and orientations and decompose the images into the components, also captures visual properties including orientation selectivity, spatial frequency, and localization.

In Figure 3.2, Gabor means amplitude and Gabor square energy of 1×30 features are obtained, in discrete domain, 2-dimensional Gabor filter to extract features from the given images, we change image area size that is parsed.

3.7. Feature Selection and Feature Fusion

Three feature vectors are acquired for each feature produced after extracting features from SFTA, LBP, and Gabor feature descriptors. The SFTA procedures a feature vector with a size of 1 21, the LBP produces a feature vector of 1 59, while Gabor returns a feature vector with a size of 1 60. After extracting these features, the strongest features are selected by using the PCA algorithm. And then all of these 3 features are fused to obtain the resultant fused feature. Description of both techniques are described in the following subsections:

3.7.1. Feature Selection

In the feature selection, we use the PCA algorithm to select the strongest and robust features and to elicit the selected featured subset from the set complete feature set. In the cheating dataset, it's very tough to identify an individual parameter that further characterizes the performance of matching across several features [54].

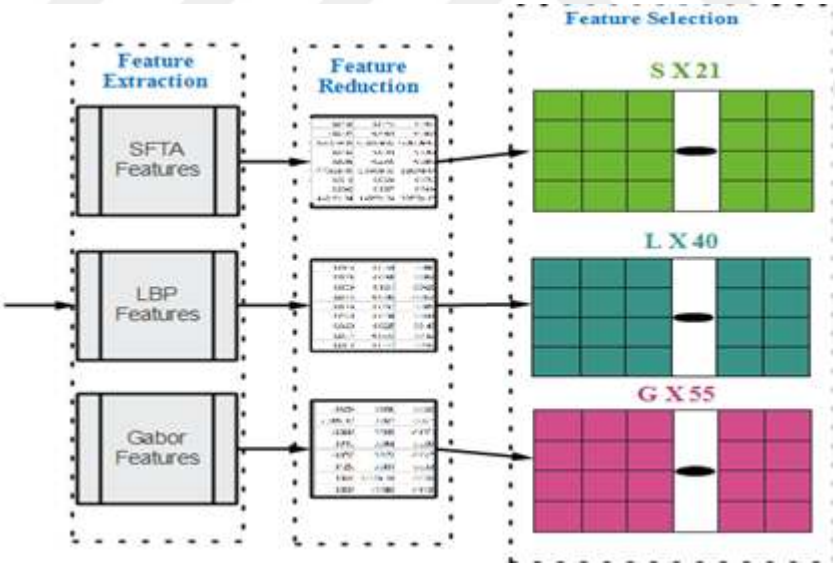

Figure 3. 3. The feature selection process for each method used

In the feature selection, PCA is used to extract more relevant features. This technique to reduce the irrelevant and redundant information from all the feature vectors and feature matrices containing 1x40 strongest features are for each image through LBP feature vector, 1 x 21 strongest features are obtained for each image through SFTA feature vector, and 1 x 55 strongest features are obtained for each image through Gabor feature vector, which is

more informative features. All of these feature vectors are represented in Figure 3. 3 as SFTA to S × 21, LBP to L × 40, and Gabor to G × 55. The main subject to use PCA is that it's the easiest non-parametric method, which extracts the features that have the most relevant information from the original feature set having noisy and redundant features. PCA is processed by eliminating the non-contributing features from the main feature vector. Due to eliminating noisy, non-contributing, and redundant features, the vector size reduces. So that the irrelevant information is removed. Further classifiers are trained on all datasets using property metrics.

Table 3. 1. Summary of the significance of this study

Impact Type	Effect	Achieving Impact Predicted Time
Academic impact:	Will be Line for others Contribute to the field	After successful completion
Economic Impact	Save human resource Save time	After successful compilation
National Security Impact:	Has no impact on national security	Any level of this study

3.8. Proposed Framework

The proposed framework for cheating recognition overview is as follows. It has some basic primary steps that must be followed during training. The steps are; labeled dataset acquisition, image coloring, and size normalization, feature extraction, strong features selection, and classification as cheating or normal. The general architecture of the proposed framework is shown in Figure 3.1.

3.9. Materials and Methods

In this section, we'll go over the resources and methods for classifying exam activities. The following are the four major steps in our proposed method: a) Extraction of detailed features b) the fusion of features c) dimensionality reduction through an ensemble of a

wrapper and a filter-based method and d) activity classification. In the first step, we extracted two types of deep learning features by using the VGG16 and SqueezeNet architecture models. We then serially fused their deep features into a single feature set. Afterward, the feature set is selected by applying ant colony optimization and entropy-based approaches to achieve a strong feature vector. Finally, the classification results are achieved by feeding the ensembled selected feature vector to the SVM-based classifiers. Figure 3.1 depicts a detailed block diagram of our method. The steps depicted in the block diagram are discussed one after the other in the accompanying section.

3.9.1. Deep Learning-Based Feature Extraction

Two pre-trained deep CNN networks are used to extract features in the proposed model. CNN is made up of many layers that are placed in a pipeline shape. Convolution (Conv-Layer), Pooling (POOL-Layer), Rectified Linear Units (ReLU-Layer), Fully-Connected (FC-Layer), and Dropout are some of the most frequent types (DO-Layer). The following are the CNN network configurations that have been chosen.

3.9.2. VGG16 Features Extraction

VGG-16 is a kind of CNN that comprises sixteen layers. Following is the configuration: (a) Input layer: The input is fixed to 224x224x3 (3 depicts the channels of RGB) size. (b) Convolution layers (Conv-layers): The sixteen layers contain twelve Conv-Layers and three fully connected (FC) layers. The twelve Conv-layers are divided into five sets.

The first three sets contain two Conv-Layers and the remaining two sets contain three Conv-Layers. 3x3 filters are used in each Conv-Layer. The depth of the filter in the first set of Conv-Layers is 64 which increases with the power of 2 in the next sets. (c) Fully Connected layers (FC-layers): FC-layers are also Conv-layers. Total three FC layers are used in VGG-16 These layers contain the features from all previous layers. The last FC layer with size 4096 is utilized for features acquisition. (d) Spatial pooling Layers (SP-Layers): Five SP-layers are employed in VGG-16 after every set of Conv-layer. In such layers, 2x2 MAX pools are used. (e) SoftMax Layer: This layer is used for classification, but we are not using this layer in this work. Instead, we use SVM classifiers for predictions. Output features with size 1 x 4096. are collected at the FC1000 layer.

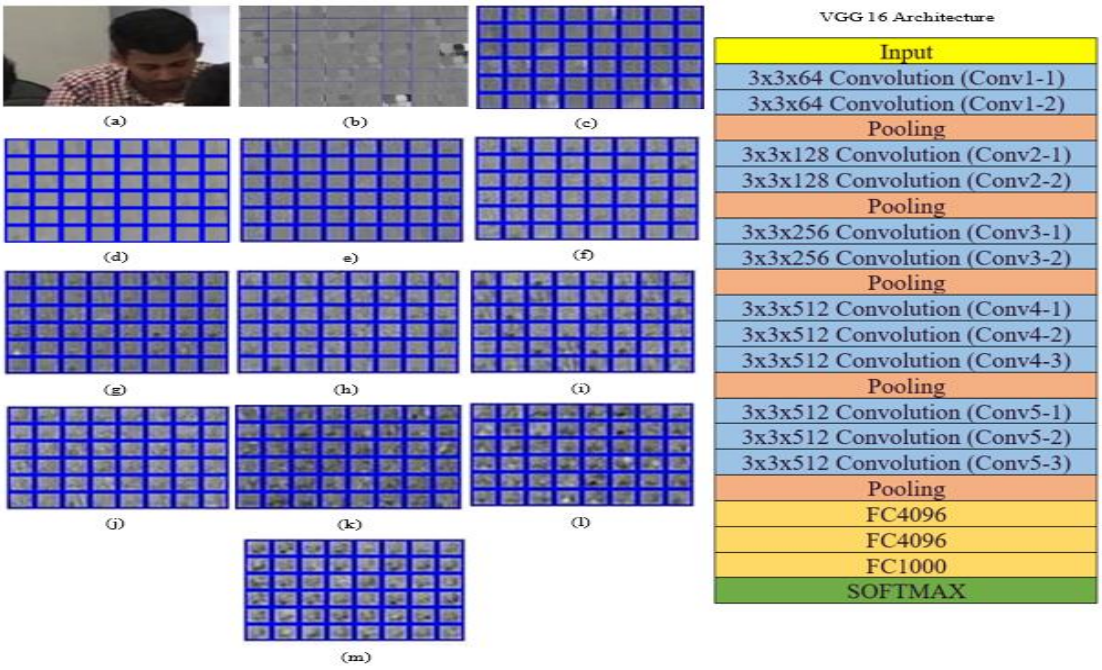

Figure 3. 4. Features Visualizations on an image at different convolution layers of VGG16.

3.9.3. Holdout at High-Scored Feature and Classifier

After taking the results against the best classifiers, the best features are selected after calculating the results on all possible combinations of features. We obtained that the best Accuracy rate against the resultant combinations. Further, we check accuracy at 5 folds and 10 folds. We take results of accuracy rate through the validation checking at 50 % holdout validation and 75% holdout validation. The accuracy scores were achieved at various accuracy rates also the ER and validation variation are presented in the following Table 3.2

Table 3. 2. Holdout Validation at Best 3 Classifiers and Features Combination at 5-Fold

Features	**Classifier**	**Accuracy**	**Holdout Validation at 50%**	**Holdout Validation at 75%**
HOG+SFTA+ LBP+ Gabor	Cubic SVM	71.9%	72.9%	69.3%
	Fine KNN	94.0%	92.8%	93.4%
SFTA+ LBP+ Gabor	Cubic SVM	71.1%	69.4%	68.1%
	Fine KNN	93.8%	92.7%	91.8%
SFTA+LBP	Cubic SVM	67.9%	68.4%	67.0%
	Fine KNN	91.7%	90.1%	89.8%

The experimental studies show that all of these combinations result in high accuracy score at the Fine KNN classifier, as well as the best-experienced combination of features having SFTA, LBP, and Gabor shows high performance across both Cross-Validations but on the Holdout Validation, the combination of HOG, SFTA, LBP, and Gabor show comparatively high accuracy results than the other two combinations.

3.9.3.1. The Results on the Feature Extraction and Selection Method (SFTA+ LBP+ Gabor) Accomplished

After the above analysis, this is the resultant combination of both of the best classifiers Cubic SVM and Fine KNN, and feature combination of SFTA, LBP, and Gabor that provides the best Accuracy rate at cross-Validation at 5-fold and 10-folds as well as calculating the holdout validation rate at 50% and 75%. The results of these validations are as summarized in Table 3.3 below:

Table 3. 3. The results for the improved method (SFTA+ LBP+ Gabor)

Best Feature's Combination SFTA + LBP + Gabor		**No. of Features**			**Cross-Validation Folds**		**Holdout Validation**	
		SFTA	**LBP**	**Gabor**	**5**	**10**	**50%**	**70%**
Best Classifier's Combination	Cubic SVM	21	40	60	71.1%	72.1%	69.4%	68.1%
	Fine KNN	21	40	60	93.8%	94.6%	92.7%	93.8%

From all the above analysis, we conclude that the resultant feature combination of SFTA, LBP, and Gabor on both of the best classifiers Cubic SVM and Fine KNN provides the best Accuracy rate. All of the experiments show that the Fine KNN classifier performs well and returns a high Accuracy score.

3.9.4. SqueezeNet Features Extraction

SqeezeNet is a convolutional network proposed in February 2016 [32]. It is a deep learning-based method and executes with better performance than the AlexNet model. This network has a maximum of 15 layers including eight fire blocks (starting from fire-2 up to fire-9), three max-pooling layers, two convolution layers, one global-average pooling layer, and one softmax layer. All the images have a fixed Input size: 227×227×3 (3=3 channels of RGB) pixels and Output features size: 1000. Each fire block starts with a convolution layer which is followed by a ReLU layer. After which, two branches are emerged with convolution, and rely on layer lie in parallel. These branches are then added catenation layer.

In this research, we obtained features on the FC1000 layer. The obtained feature vector is of size N×1000, where N shows the image count of the dataset. The squeeze phase uses the filter of sizes 3×3 and 1×1. The architecture and features visualizations on a sample input image at different convolution layers of Squeezenet are given in Figure 3.5.

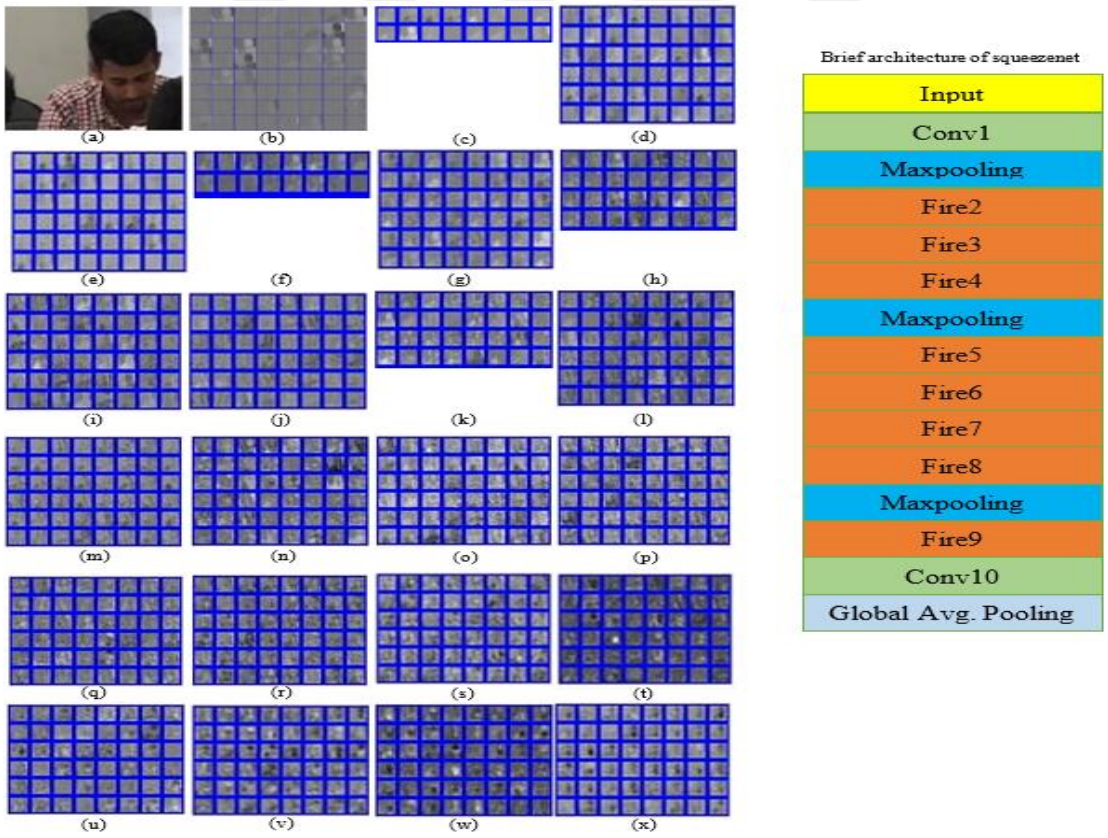

Figure 3. 5. The architecture and features visualizations on an input image at different convolution layers of Squeezenet.

3.10. Feature Subset Selection

The features sub-selection is performed to acquire the optimal features. For this purpose, we acquire selected features from both filters based and wrapper-based approaches. For the filter-based approach, Entropy-based feature ranking is used while the ant colony system is utilized for the wrapper-based approach. The two types of selected features are employed to acquire the characteristics of both features' selection approaches.

3.10.1. Entropy-based Feature Ranking for Features Subset Selection

Entropy [33] is used for ranking the features. The reason we perform ranking of features is to select the top-ranked feature subset. Let n be the number of extracted features (ξ_1,ξ_2,…,ξ_n) acquired from the feature extraction phase. The entropy value is calculated for each feature by using the following equation:

$$\varrho(\xi_1, \xi_2, \dots, \xi_n) = -\sum_{f_1} \dots \sum_{f_n} þ(f_1, \dots, f_n)\ LOG\ þ(f_1, \dots, f_n) \quad (1)$$

where ϱ represents entropy, $(\xi_1, \xi_2, \dots, \xi_n$n) depicts the features (which are deemed as factors randomly), (f_1,…,f_n) are the associated random variable's values. þ(f_1,…,f_n) shows the probability of the points (f_1, .,f_n). The entropy values are sorted in descending order. Finally, first, k values are then selected as a feature subset mathematically:

$$\varrho(\xi_1, \xi_2, \dots, \xi_k) = Select_{1-k}\left(sort\left(\varrho(\xi_1, \xi_2, \dots, \xi_n)\right)\right) \quad (2)$$

3.10.2. Ant Colony Optimization-based Features Subset Selection

This approach comes under the category of wrapper-based approach. It is also known as ant colony system-based feature optimization. It depends on the probability theory [34]. It is inspired by ants' behaviors. The ants while traveling, spread a substance known as a pheromone. This substance intensity reduces with time. The ants follow the route with probabilistically high intensity of pheromone. This guides them to seek the least cost route.

The movement of ants is just like traveling in a graph i.e., from node to node. A node depicts a feature and the edges among the nodes show the option to select the following feature. The algorithm seeks the optimal features. The algorithm stops when the least number of nodes are visited and a stopping condition is reached. All nodes in the graph are connected in a mesh-like structure. The pheromone values are coupled with nodes (features). The

feature is selected by an ant depending on the probability at a certain time mathematically given as:

$$\text{Þ}_j^m(\text{Ŧ}) = \begin{cases} \frac{|\,\Gamma_j(\text{Ŧ})|^{\bar{\text{w}}}|\,\rho_j|^{\text{Ꮗ}}}{\sum_{v\in(\xi_1,\xi_2,\ldots,\xi_m)}|\,\Gamma_v(\text{Ŧ})|^{\bar{\text{w}}}|\,\rho_v|^{\text{Ꮗ}}} \; if\; j \in (\xi_1,\xi_2,\ldots,\xi_m) \\ 0\; otherwise \end{cases} \quad (3)$$

where (ξ_1,ξ_2,…,ξ_k) represents the feature set. If these features are not yet visited, then they will be the component of the unfinished solution. Γ_j (Ŧ) and ρ_j depict pheromone and empirical values linked with the ith feature. w̄ and ꞶƆ show the cost of pheromone and empirical knowledge, respectively. Ŧ portrays the time limit.

3.10.3. Fusion on the Features

In this work, we used feature fusion to utilize the benefits of both entropy and ACO techniques. Mathematically, the fusion approach entails serial concatenation of both feature subsets:

$$\varrho(\xi_1,\xi_2,\ldots,\xi_{k+m}) = \varrho(\xi_1,\xi_2,\ldots,\xi_k) \oplus A(\xi_1,\xi_2,\ldots,\xi_m) \quad (4)$$

where $\varrho(\xi_1,\xi_2,\ldots,\xi_k)$ show the entropy-based feature subset and $A(\xi_1,\xi_2,\ldots,\xi_m)$ represents the ACO-based feature subset. $\oplus$ depict the concatenation process.

After the process of feature extraction from three features extractors: SFTA, Gabor, and LBP, we obtain three individual feature vectors for all these descriptors as SFTA having feature vector of size 1 × 21, LBP having feature vector of size 1 × 40, and Gabor having 1 x 55-pixel feature vector, after that, These three vectors were serially merged or perform a simple concatenation to merge these feature vectors. The objective of concatenation is to yield a single fused vector or a new feature vector.

This new feature vector is obtained by augmenting the resultant vector of SFTA, LBP, and Gabor and then implementing the feature selection on the resultant feature vector. In the resultant vector of feature fusion, we obtain one of the best and more relevant feature subsets from the original feature set. A serial-based fusion technique is used in this study. If we describe SFTA feature vectors as:

$$F_{S_{1\times d}} = \{\, \text{S}_{1\times1}, \text{S}_{1\times2}, \text{S}_{1\times3} \ldots\ldots \text{S}_{1\times\text{d}}\} \quad (4.1)$$

Where $F_{S_{1\times d}}$ represents the vector dimensions of the SFTA feature vector, Furthermore LBP feature vector describes as:

$$F_{L_{1\times c}} = \{ \mathrm{L}_{1\times 1}, \mathrm{L}_{1\times 2}, \mathrm{L}_{1\times 3} \dots \dots . \mathrm{L}_{1\times \mathrm{c}}\} \quad (4.2)$$

Where $F_{L_{1\times d}}$ represents the vector dimensions of the LBP feature vector, Furthermore Gabor feature vector describes as:

$$F_{G_{1\times r}} = \{ \mathrm{G}_{1\times 1}, \mathrm{G}_{1\times 2}, \mathrm{G}_{1\times 3} \dots \dots . \mathrm{G}_{1\times \mathrm{r}}\} \quad (4.3)$$

Where $F_{G_{1\times d}}$ represents the vector dimensions of the Gabor feature vector, the given input features vectors are simply linked together successively or horizontally. The concatenated vector is represented as:

$$F_{R_{1\times j}} = \sum (F_{S_{1xd}}, F_{L_{1xc}}, F_{G_{1xr}}) \quad (4.4)$$

Where $F_{R_{1\times j}}$ the output resultant vector is the combined feature vector after concatenation and j= d+c+r then the mean is calculated of the output feature vector as:

$$F_k = \frac{1}{j} \sum_{i=0}^{j} F_{R_i} \quad (4.5)$$

Following that Euclidian distance is searched out for each feature w.r.t mean value and put on an activation function that will organize the vector in the least distance order. The Euclidian distance of the features are explained as:

$$F_d = \{d_1, d_2, d_3, \dots \dots \dots . . d_j\} \quad (4.6)$$

The resultant feature vector is represented as:

$$f_{F_{1\times k}} = \{F_{1\times 1}, F_{1\times 2}, F_{1\times 3}, \dots \dots \dots . F_{1\times K}\} \quad (4.7)$$

The resultant feature vector is represented as:

$$f_{F_{1\times k}} = \{F_{1\times 1}, F_{1\times 2}, F_{1\times 3}, \dots \dots \dots . F_{1\times K}\} \quad (4.8)$$

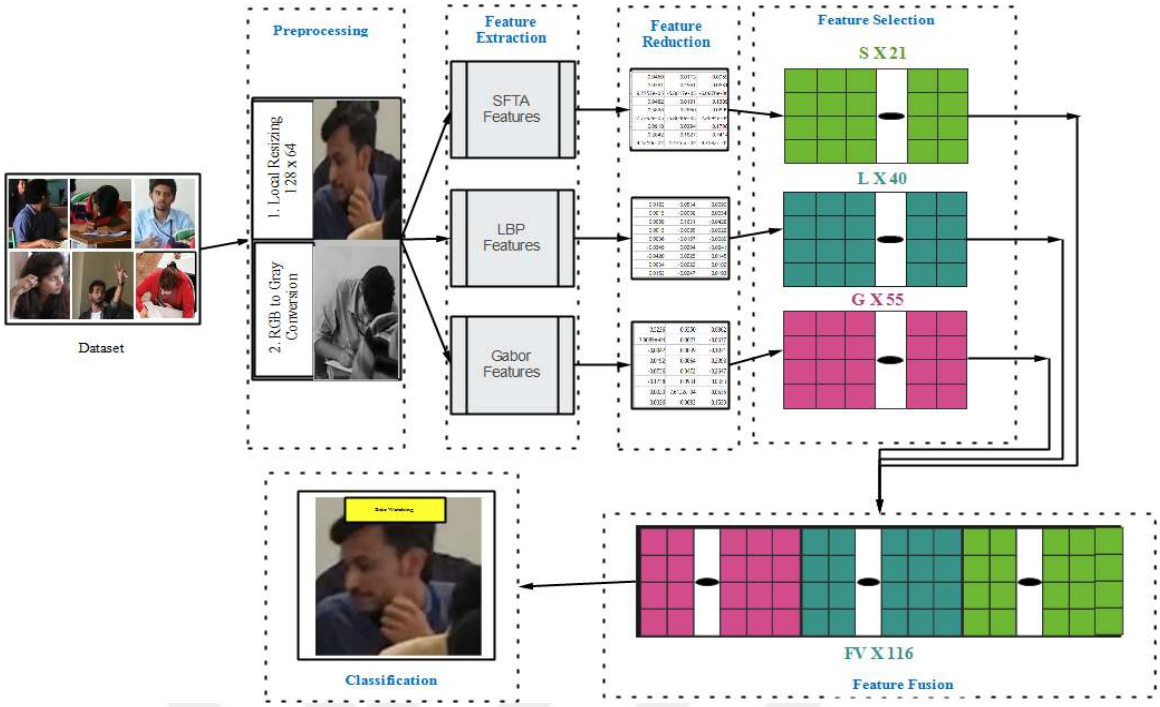

Figure 3. 6. Representation of Feature Fusion

3.10.4. Classification

Next after extracting the feature, the system is classified utilizing different classifiers and a verification process is implemented based on these selected features. The obtained picture features are categorized by these selected features. In the human detection process consider the object as a human or-non-human and in the phase of activities classification by applying the selected classifiers and considered in either different classes where class A denotes jogging, B clapping, and similarly many other activities [79]. Most popular choices of classifiers are KNN [42] , linear SVM [44], Random forest [46], AdaBoost [48] and ANN classifier [53].

In this phase of activities classification, we apply the selected classifiers Fine KNN and Cubic SVM and consider in either different classes where A denotes Back Watching, B Front Watching, C Side Watching, and many others.

The fused features in the previous phase are then fed to SVM-based classifiers for exam activities categorization. To ensure that our hypothesized model is effective, we train and evaluate it with six different SVM classifiers: cubic SVM (C-SVM), linear SVM (L-SVM), quadratic SVM (Q-SVM), fine Gaussian SVM (FG-SVM), medium Gaussian SVM

(MG-SVM), and coarse Gaussian SVM (CG-SVM). The performance evaluation (in the following section) shows that C-SVM outperforms alternative predictors.

4. RESULT AND DISCUSSION

4.2. Introduction

In this section, results of the proposed work have been described which are used to measure the Accuracy Score for the classification measures using the classifiers of SVM's and KNN's, different experiments are performed on the six classes of student monitoring during the examination on different cheating scenarios, these classes include: 1) front watching, 2) back watching, 3) side watching, 4) normal, 5) suspicious watching and 6) showing gestures, Figure 4.1 shows representative images from the aforementioned classes. Different experiments are performed with a distinct number of features.

4.3. Results

All classifier learners are used to evaluate or detect the high accuracy score against all the features individually as well as with the combination of different features. There are some characteristics for checking the performance of the proposed work are accuracy (ACC) at different folds mainly 5 fold and 10 folds are considered, holdout validation of 50% and 70% are considered. On a 64-bit operating system with a Core i7 with 8GB RAM and a 2.4 GHz processor, all steps of the planned work are implemented using the tools MATLAB R2018a and MATLAB R2018b.

For this study, there are no public image datasets available for the paper-based examination held in an examination hall. So, a new dataset is prepared by framing out the videos into images, cropping the boundaries of the interesting area in these images, and assigning labels to these boundaries according to their relative class. The classification is performed according to the gesture of students that specify a way of cheating they use and some certain activities students mostly perform during the cheating. This will be a challenging dataset because of the diversity of images, effects of illumination, occlusion, and dynamic variation in pose, also due to contrast variations.

As some images may look very close or similar to the images of another class due to minute variation in gestures, it becomes hard to distinguish them. So, the process of manually labeling the images and distribution of these images into their related classes is also a very

challenging activity because of the minute difference in their poses, gestures showing, and many other factors. Due to these variations, the dataset will be distributed or classified into some major classes as 1) front watching having 1295 images, 2) back watching having 537 images, 3) side watching having 851 images, 4) normal having 2137 images, 5) suspicious watching having 602 images, and 6) showing gestures having 590 images. Some sample images for distribution and classification in the dataset can be visualized as in Figure 4.1.

Figure 4. 1. Some sample images collected from different classes in the dataset

The classification will be performed according to the gesture activities students mostly perform during cheating. This will be a challenging dataset due to the diversity of images, illumination effects, occlusion, pose variation, and contrast variations.

Some images may very similar to the images of another class, due to variation in gestures, it is hard to distinguish them. Manually labeling the images and distributing these images into their related classes is also a very challenging activity because of the difference in their poses. So, the dataset will be classified into some major classes as front watching, back watching, side watching, suspicious watching, showing gestures, copying from notes, copying phone, asking or giving an answer for a classmate, and normal watch. The distribution and classification of the dataset can be visualized as in Figure 4.2.

Figure 4. 2 . Representation of dataset including some sample classes of cheating

4.4. Dataset (preparation and augmentation)

The training and performance evaluation in this work is performed on the dataset collected by the researcher. The suspicious activity dataset (see Figure 4.3 for sample images) is prepared. The dataset is based on videos acquired from surveillance cameras recorded during the actual exam.

The suggested framework for feature reduction is learned and tested on the given dataset (CUI-EXAM) taken from the students conducting the exam at different Ethiopian universities, which is comprised of approximately 2000 images of students 'activities acquired through CCTV cameras during exams. Dataset (CUI-EXAM) is for students behavior detection and classification containing bounding boxes from 50 videos annotated into 6 action classes as given in Table 4.1. Image augmentation is performed to enhance the size of the dataset by flipping the pictures. This doubles the size of the dataset. Some images belonging to this dataset are shown in Figure 4.2 representing different activities scenes including (a) back watching, (b) front watching, (c) normal watching, (d) showing gestures, (e) side watching, and (f) suspicious.

Table 4. 1. CUI-EXAM Dataset detail

Labels	Original	Original + Flipped
BackWatching	407	812
FrontWatching	286	570
Normal	561	1120
Showing-gestures	327	652
Sidewatching	380	758
Suspicious	313	624
Total	2274	4536

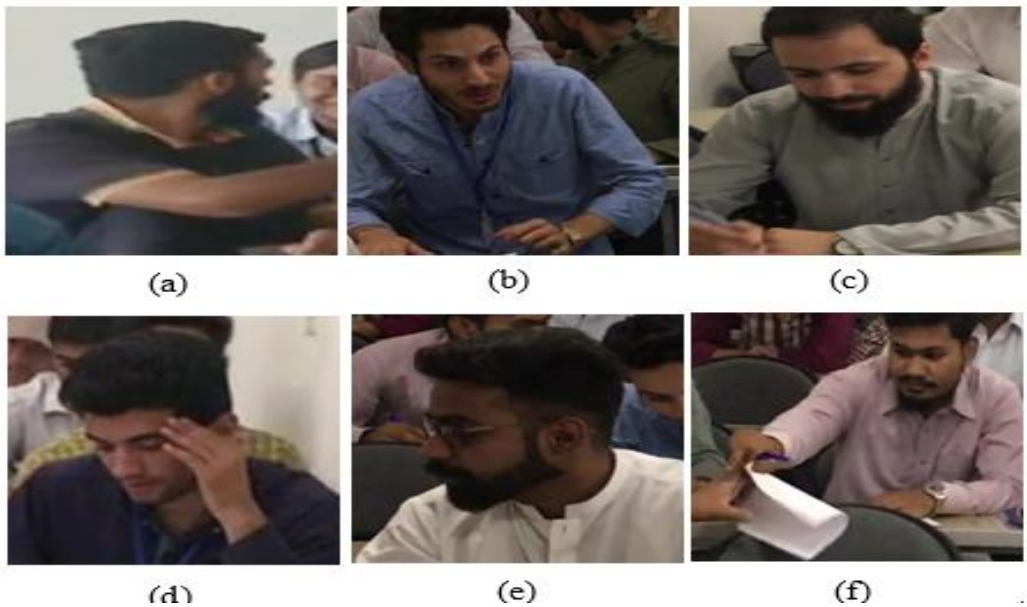

Figure 4. 3 . Examples of images from the CUI-EXAM dataset

4.5. Performance Evaluation

We use a variety of performance indicators to assess the proposed work's performance in this study. To evaluate the quality of the algorithm, some evaluation measurements such as accuracy, confusion matrix, misclassification rate, TPR, FPR, sensitivity (SEN), specificity (SPE), precision (PRE), prevalence, error rate (ER), F-Score, and ROC curves, and AUC are calculated from output image. First, the algorithm is implemented on the given

input image taken from the dataset of cheating monitoring during an examination to extract the output image or feature image.

The evaluation measure is then calculated to determine an algorithm's efficiency. To determine the algorithm's correctness, sensitivity and specificity were calculated. Higher accuracy scores, SEN, and SPE guarantee the classifier's usefulness as well as the algorithm's quality. Other evaluation measures can ensure a particular algorithm's error rate of erroneous outcomes. TPR and TNR are used to obtain the true classification rate of a given algorithm. The value of TPR and TNR or 1 value ensures that the given algorithm produces an error-free and better result. FPR and FNR are used to obtain the error rate of a given algorithm.

The value of FPR and FNR or zero value ensures that the correctness of the given algorithm. Every point on the ROC curve shows the SEN/SPE correspondence towards a particular decision threshold. AUC measures the extent to which the parameters distinguish between different classes.

These figures are calculated using a confusion matrix. Mathematically these measures are given as follows.

$$\text{Sensitivity (recall/TPR)}=\frac{\text{TruePositive}}{\text{TruePositive+FalseNegative}} \tag{5}$$

$$\text{Accuracy}=\frac{\text{TruePositive +TrueNegative}}{\text{TruePositive+TtrueNegative+FalsePositive+FalseNegative}} \tag{6}$$

$$\text{Specificity (TNR)}=\frac{\text{TrueNegative}}{\text{TrueNegative+FalsePositive}} \tag{7}$$

$$\text{AUC}=\frac{\text{TruePositive+TrueNegative}}{\text{TruePositive+TrueNegative+FalsePositive+FalseNegative}} \tag{8}$$

$$\text{Precision}=\frac{\text{TruePositive}}{\text{TruePositive+FalsePositive}} \tag{9}$$

$$\text{F-measure}=2\times\frac{\text{Pricisions}\times\text{Recals}}{\text{Pricisions}\times\text{Recals}} \tag{10}$$

$$\text{G-Mean}=\sqrt{TPR\times TNR} \tag{11}$$

4.6. Overview of the Experiments Performed

Extensive testing is carried out with various combinations of selected features using the proposed model to achieve the best results. A few important experiments are listed here. Table 4.2 explains the summary of described experiments and the combination of features. These chosen experiments are explained in detail in the accompanying section. The performance in all of these mentioned experiments is evaluated using a 5-folds cross-validation mechanism.

Table 4. 2. A breakdown of the studies carried out

Test #	Model	Unique characteristics		Total features used for classification
		Entropy	ACO	
1	SqueezeNet (4096) +VGG16 (1000)	150	150	300
2	SqueezeNet (4096) +VGG16 (1000)	250	250	500
3	SqueezeNet (4096) +VGG16 (1000)	500	500	1000
4	SqueezeNet (4096) +VGG16 (1000)	750	750	1500
5	SqueezeNet (4096) +VGG16 (1000)	1000	1000	2000
6	SqueezeNet (4096)	250	-	250
7	VGG16 (1000)	-	250	250

4.6.1. Experiment 1: VGG16+SqueezeNet (Entropy: 150+ ACO: 150 features)

In this test, we were using the fused feature map to evaluate the suggested method's performance. We employ SqueezeNet and VGG16 features in this experiment and the feature extraction. The feature vector of size 4536×4096 and 4536×1000 is utilized for VGG16 and SqueezeNet, respectively. Entropy and ACO are applied parallelly in the next

step on the fusion of both obtained feature vectors for feature selection. In this experiment, we have selected the 150 features from both entropy and ACO-based approaches.

A total of 5748 images were collected that were further distributed into sub-classes including back watching, front watching, normal, side watching, showing gestures, and suspicious watching. For the results of these experiments, we adopted a ratio of 50:50 to train and test the algorithm. This means that these classes further contain 273 back watching, 590 showing gestures, 2137 normal, 851 side watching, 602 suspicious watching, and 1295 front watching images. These all are selected to test the algorithm

These selected features are later fused to get a feature matrix of size 4536×300. This feature matrix is fed to the classifiers for classification. In this experiment, C-SVM has attained the best accuracy of 92%. and MG-SVM has achieved the second-highest accuracy of 83.73%. Table 4.3 shows the findings of this experiment. Similarly, Figure 2.5 elaborates the time chart of VGG16+SqueezeNet (Entropy: 150+ ACO: 150 features). It is observed that our method works best on the C-SVM classifier for the CUI-EXAM dataset.

These classification results under the Fine KNN are confirmed through the confusion Metrix of the selected classifier (Fine KNN) shown in Table 4.3. The evaluation below reveals that the specified technique performs better and has a high Accuracy score for a feature vector of size 1 x 80. The resultant performance evaluation measures in Table 4.3 reports the analysis.

Table 4. 3. Outcomes of VGG16+SqueezeNet (Entropy: 150+ ACO: 150 features)

Classifier	Percentage of Correctness	Percentage of sensitivity	Percentage of specificity	Percentage of Precision	Percentage of F-measure	G-mean in percent
L.SVM	64.53	73.28	62.62	29.94	42.52	67.74
Q-SVM	83.22	91.01	81.53	51.79	66.01	86.14
FG-SVM	42.68	47.29	41.68	15.02	22.80	44.39
MG-SVM	83.73	90.52	82.25	52.65	66.58	86.28
CG-SVM	55.07	58.37	54.35	21.80	31.75	56.33
C-SVM	90.52	94.33	89.69	66.61	78.08	91.98

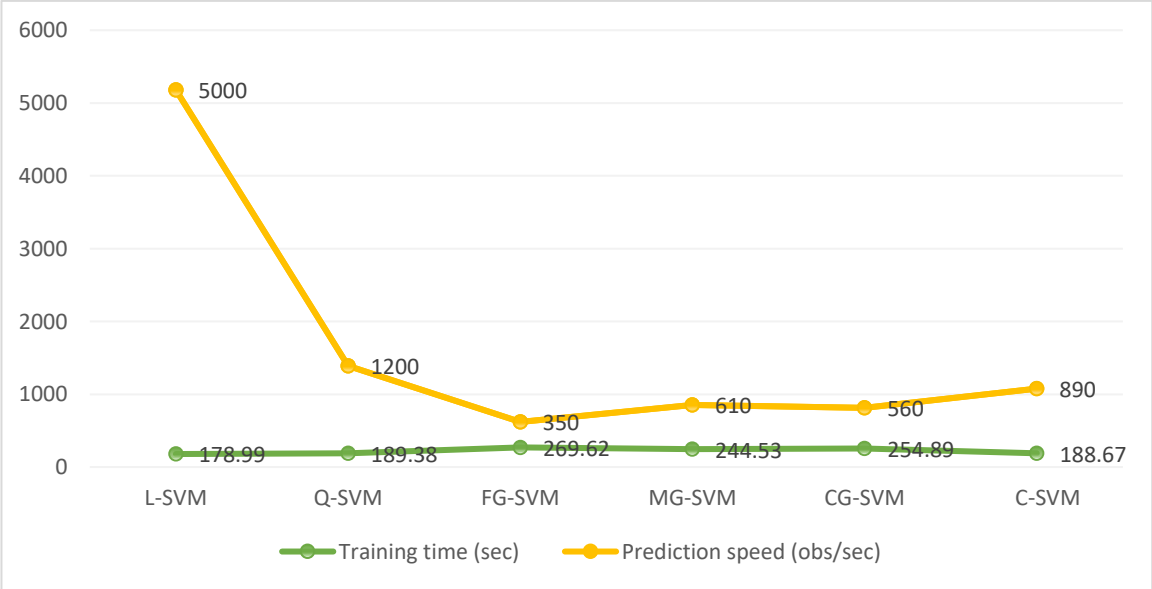

Figure 4. 4. VGG16+SqueezeNet (Entropy: 150+ ACO: 150 features) time graph

4.6.2. Experiment 2: VGG16+SqueezeNet (Entropy: 250+ ACO: 250 features)

In experiment 2, a total of 5748 images were collected that were further distributed into subclasses including Back watching, Front watching, Normal, Side watching, Showing gestures, and Suspicious watching. For the results of these experiments, we adopted a ratio of 50:50 to train and test the algorithm. This means that these classes further contain 273 Back watching, 590 Showing gestures, 2137 Normal, 851 Side watching, 602 Suspicious watching, and 1295 Front watching images. These all are selected to test the algorithm. For the evaluation of these results, we use 5-fold cross-validation.

We use VGG16 and SqueezeNet features in this experiment and in the feature extraction, we obtain the feature vectors of size 4536×4096 and 4536×1000 for VGG16 and SqueezeNet, respectively.

Entropy and ACO are applied parallelly in the next step on the fusion of both obtained feature vectors for feature selection. In this experiment, we select the 250 features from both entropy and ACO-based approaches. Those are later fused to get a feature matrix of size 4536×500. For labeling, this feature matrix is given to SVM-based classifiers. In this experiment, C-SVM attains the best accuracy of 90.85%. and MG-SVM has achieved the second-highest accuracy of 88.49%. The results of this experiment are given in Table 4.4. and Figure 4.5 shows the time chart of VGG16+SqueezeNet (Entropy: 250+ ACO: 250

features). For the CUI-EXAM dataset, we found that our technique outperforms the C-SVM classifier.

The results of the preceding analysis demonstrate that the defined technique performs better and has a high Accuracy score for a feature vector of size 1 x 103. Table 4.5 shows the performance evaluation measures that resulted.

Table 4. 4. VGG16+SqueezeNet Results (Entropy: 250+ ACO: 250 features)

Classifier	Percentage of Correctness	Percentage of sensitivity	Percentage of specificity	Percentage of Precision	F-measure Percentage	G-mean in percent
LSVM	66.34	74.38	64.58	31.41	44.17	69.31
Q-SVM	84.46	92.00	82.81	53.86	67.94	87.28
FG-SVM	42.57	43.97	42.27	14.24	21.51	43.11
MG-SVM	88.49	93.10	87.49	61.87	74.34	90.25
CG-SVM	58.73	64.90	57.38	24.93	36.02	61.03
C-SVM	90.85	95.44	89.85	67.22	78.88	92.60

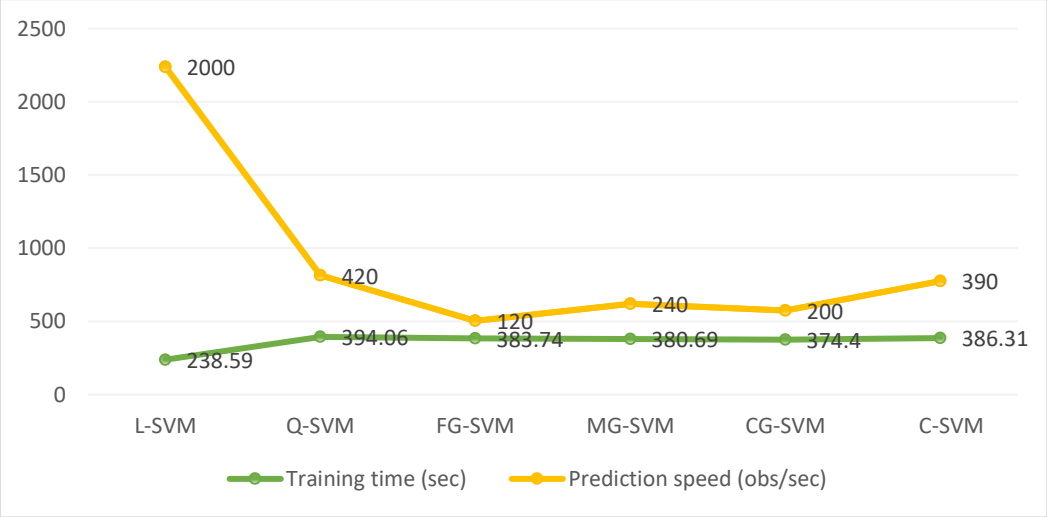

Figure 4. 5. VGG16+SqueezeNet (Entropy: 250+ ACO: 250 features) time graph

4.6.3. Experiment 3: VGG16+SqueezeNet (Entropy: 500+ ACO: 500 features)

In experiment 3, a total of 5748 images was collected that were further distributed into sub-classes including back watching, front watching, normal, side watching, showing gestures, and suspicious watching. For the results of these experiments, we adopted a ratio of 50:50 to train and test the algorithm. This means that these classes further contain 273 Back watching, 590 Showing gestures, 2137 Normal, 851 Side watching, 602 Suspicious watching, and 1295 front watching images.

SqueezeNet and VGG16 features are employed in this experiment and the feature extraction, we obtain the feature vector of size 4536×4096 and 4536×1000 for VGG16 and SqueezeNet respectively. Entropy and ACO are applied parallelly in the next step on the fusion of both obtained feature vectors for feature selection. We select the 500 features each from both entropy and ACO-based approaches.

That is later fused to get a feature matrix of size 4536×1000. This feature matrix is fed to the classifiers for classification. In this experiment, C-SVM has attained the best accuracy of 91.51%. MG-SVM, on the other hand, has the 2nd accuracy of 90.08 percent. Table 4.5 summarizes the detail of this trial. Figure 4.6 represents the time chart of VGG16+SqueezeNet (Entropy: 500+ ACO: 500 features). It is found that our method proves best on the C-SVM classifier for the CUI-EXAM dataset. The resultant performance evaluation measures are shown in Table 4.5.

Table 4. 5. Results of VGG16+SqueezeNet (Entropy: 500+ ACO: 500 features)

Classifier	Percentage of Correctness	Percentage of sensitivity	Percentage of specificity	Percentage of Precision	Percentage of F-measure	G-mean in percent
L.SVM	68.10	77.34	66.08	33.21	46.47	71.49
Q-SVM	86.27	93.10	84.77	57.14	70.82	88.84
FG-SVM	42.31	42.49	42.27	13.83	20.86	42.38
MG-SVM	90.08	91.50	89.77	66.10	76.76	90.63
CG-SVM	66.05	75.49	63.99	31.37	44.32	69.50
C-SVM	91.51	95.32	90.68	69.05	80.08	92.97

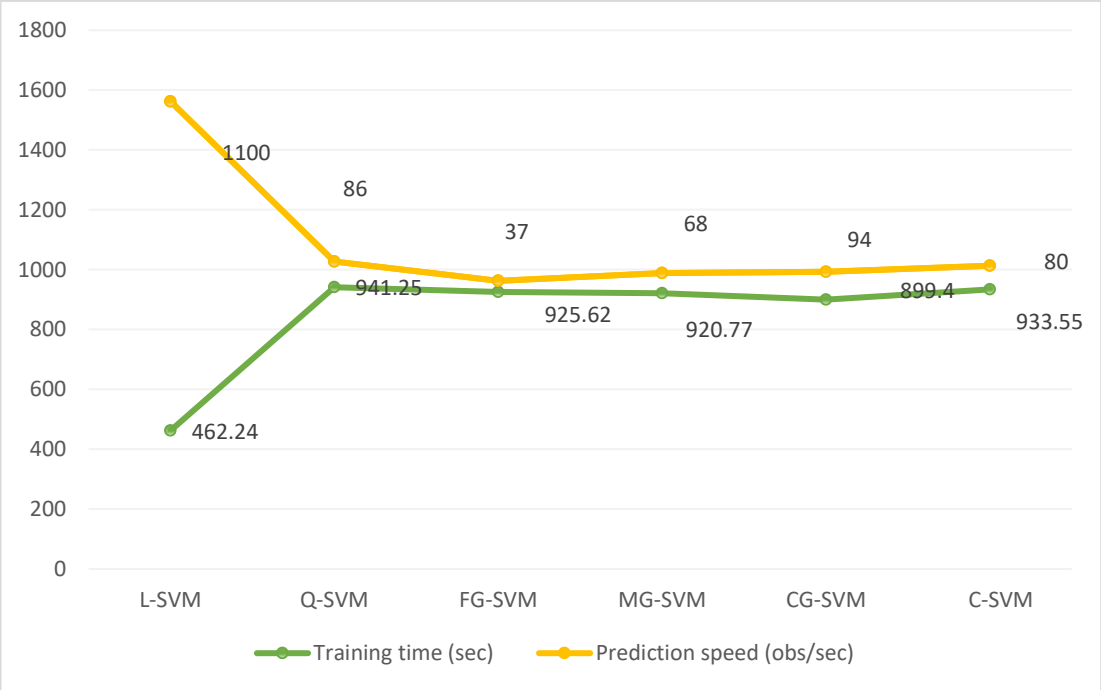

Figure 4. 6. VGG16+SqueezeNet (Entropy: 500+ ACO: 500 features) time chart

4.6.4. Experiment 4: VGG16+SqueezeNet (Entropy: 750+ ACO: 750 features)

In experiment 4, a total of 5748 images were collected that were further distributed into subclasses including back watching, front watching, normal, side watching, showing gestures, and suspicious watching. For the results of these experiments, we adopted a ratio of 50:50 to train and test the algorithm. This means that these classes further contain 273 Back watching, 590 Showing gestures, 2137 Normal, 851 Side watching, 602 Suspicious watching, and 1295 Front watching images. These all are selected to test the algorithm. For the evaluation of these results, 5 fold cross-validation is used.

In this analysis, SqueezeNet and VGG16 features are employed and in the feature extraction, we obtain the feature vector of size 4536×4096 and 4536×1000 for VGG16 and SqueezeNet respectively. Entropy and ACO are applied parallelly in the next step on the fusion of both obtained feature vectors for feature selection. We select the 750 features each from both entropy and ACO-based approaches.

These are then fused to produce a feature matrix with a dimension of 4536x1500. This feature matrix is fed to the classifiers for classification. C-SVM has attained the best accuracy of 92.31%. and Q-SVM has achieved the second-highest accuracy of 85.58%. Table 4.6 displays the results of this investigation. Figure 4.7 depicts the time chart of

VGG16+SqueezeNet (Entropy: 750+ ACO: 750 features). It is found that our method proves best on the C-SVM classifier for the CUI-EXAM dataset. These classification results under the Fine KNN are confirmed through the confusion Metrix of the selected classifier shown in Table 4.6.

Table 4. 6. Results of VGG16+SqueezeNet (Entropy: 750+ ACO: 750 features)

Classifier	Percentage of Correctness	Percentage of sensitivity	Percentage of specificity	Percentage of Precision	Percentage of F-measure	G-mean in percent
LSVM	68.23	74.14	66.94	32.84	45.52	70.45
Q-SVM	85.58	93.10	83.94	55.83	69.81	88.40
FG-SVM	41.42	41.75	41.35	13.44	20.33	41.55
MG-SVM	77.87	80.91	77.20	43.63	56.69	79.03
CG-SVM	70.99	80.91	68.82	36.14	49.96	74.62
C-SVM	92.31	96.18	91.46	71.06	81.74	93.79

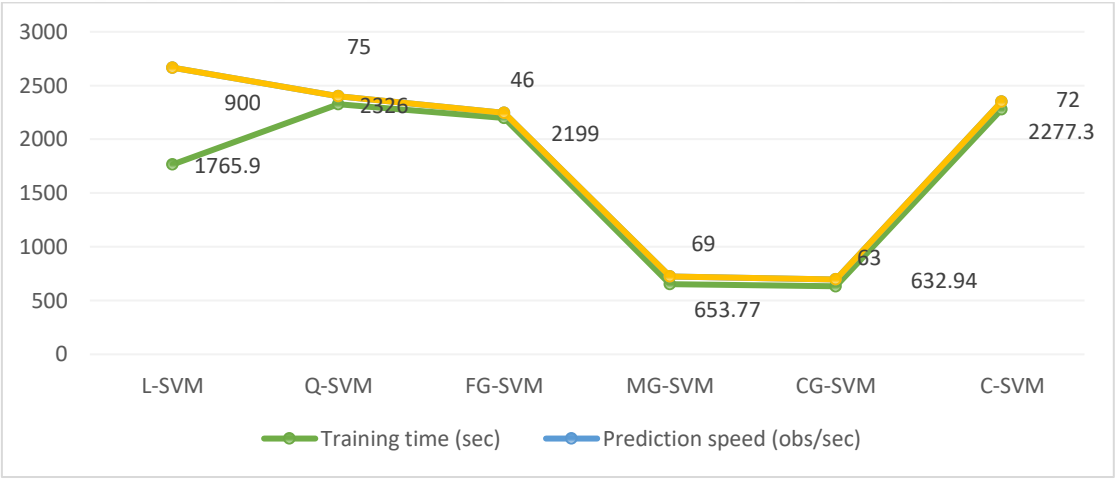

Figure 4. 7. VGG16+SqueezeNet (Entropy: 750+ ACO: 750 features) time chart

It is also observed that C-SVM performs best with this combination of feature selection (Entropy: 750+ ACO: 750 features) as compared to the rest of the experiments. With a rise in the number of features, speed decreases. Figure 4.8 shows the ROC curves and corresponding AUCs, showing the acceptable values with all classes having AUC greater

than 0.95. The true positives in Table 4.8's confusion matrix suggest that the proposed model with selected 1500 attributes performs well.

4.7. Classification Results on the Experimental Studies

In experiment 5, a total of 5748 images were collected that were further distributed into subclasses including back watching, front watching, normal, side watching, showing gestures, and suspicious watching. For the results of these experiments, we adopted a ratio of 50:50 to train and test the algorithm. This means that these classes further contain 273 Back watching, 590 Showing gestures, 2137 Normal, 851 Side watching, 602 Suspicious watching, and 1295 Front watching images. These all are selected to test the algorithm. For the evaluation of these results, we use 5 fold CrossValidation. In test 5, we achieved 73.5% the highest Accuracy score at classifier Fine KNN as given in Table 4.13 and Table 4.14. This table shows that compared with other classification measures, the classifier Fine KNN performs well than the other ones in terms of ACC (73.5), SEN (70.667), SPE (5.867), PRE (73.667), FNR (26.5), FPR (0.06), AUC (0.8233) and F Score (72.1358) under the execution time of 2.8752seconds. These classification results under the Fine KNN are confirmed through the confusion Metrix of the selected classifier (Fine KNN) shown in Table 4.16. The results of the preceding analysis demonstrate that the defined technique performs better and has a high Accuracy score for a feature vector of size 1 x 90. Table 4.7 shows the performance evaluation measures that resulted.

Table 4. 7. ClassificationResults of Experiment 5

Classifier	Performance Evaluation								
	ACC (%)	SEN (%)	SP (%)	PRE (%)	FNR (%)	FPR (%)	AUC (%)	F-Score	Execution Time
Cubic SVM	70.5	63.33	7.43	71.0	29.5	0.06	0.89	66.95	28.29
Fine KNN	73.5	70.66	5.867	73.66	26.5	0.06	0.82	72.14	2.88

4.7.1. Confusion Matrix of Best Experimental Result (Experiment. 3)

The confusion matrix presented in Table 4.8 is the resultant of high accuracy measure, this confusion matrix can be used to calculate the FPR and AUC for all the classes including back watching, front watching, normal, side watching, showing gestures, and suspicious watching separately.

Table 4. 8. Confusion Matrix of Experiment 3

True Classes \ Predicted Classes	Back watching	Front watching	Normal	Showing gestures	Side watching	Suspicious
Back watching	96%	0%	1%	<1%	3%	0%
Front watching	<1%	89%	7%	1%	1%	1%
Normal	0%	1%	94%	1%	2%	1%
Showing gestures	1%	1%	3%	92%	2%	1%
Side watching	3%	1%	9%	2%	84%	1%
Suspicious	0%	1%	4%	2%	2%	91%

4.7.2. Resultant ROCs of Best Experimental Result (Experiment. 3)

These ROC curves are under the resultant of high Accuracy measure, these curves are used to calculate the FPR and AUC for all the classes including back watching, front watching, normal, side watching, showing gestures, and suspicious watching separately. The resultant values of the ROC curve are shown in figure 4.8.

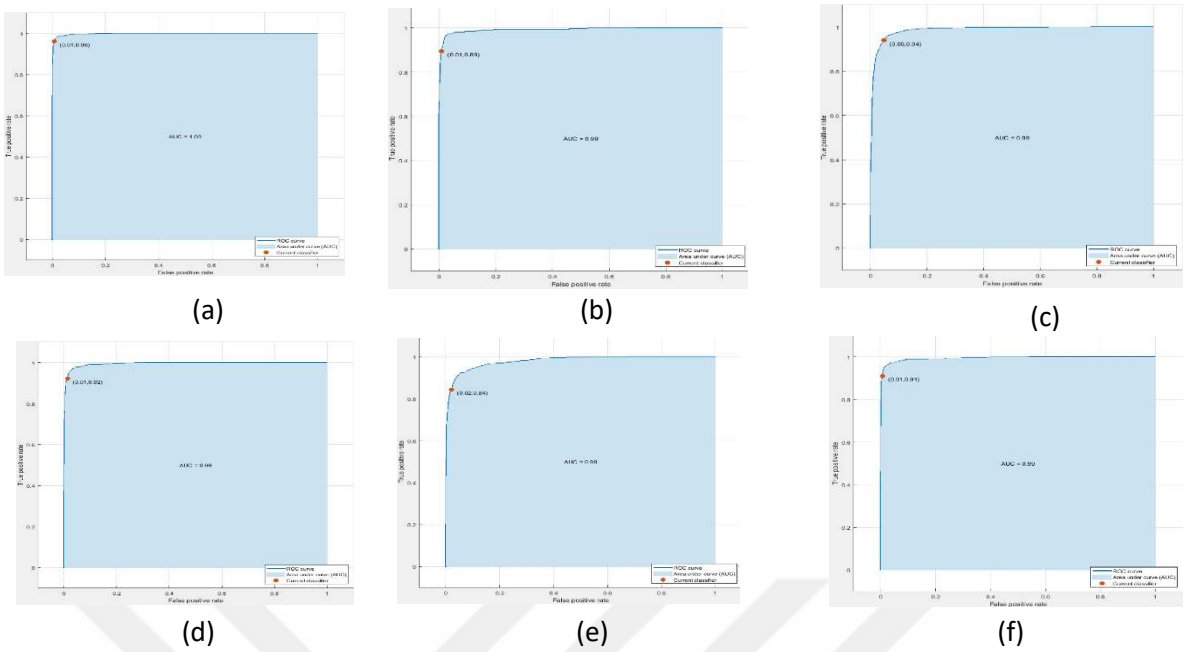

Figure 4. 8. Class wise AUCs of best-predicted classifier Entropy: 750+ features

Table 4. 9. Confusion matrix of best-selected ensembled features

True Classe						
Back watching	781	0	4	5	20	2
Front watching	3	498	34	15	17	3
Normal	5	23	1043	16	28	5
Showing gestures	1	9	7	619	15	1
Side watching	19	12	47	15	658	7
Suspicious	0	5	9	14	8	588
	Back watching	**Front watching**	**Normal**	**Showing gestures**	**Side watching**	**Suspicious**

Predicted Classes

4.7.3. Experiment 5:VGG16+SqueezeNet(Entropy:1000+ ACO: 1000 features)

In this investigation, we use the fusion vector to evaluate the suggested method's efficiency. We employ SqueezeNet and VGG16 features in this experiment and the feature extraction. The feature vector of size 4536×4096 and 4536×1000 is utilized for VGG16 and SqueezeNet, respectively. Entropy and ACO are applied parallelly in the next step on the fusion of both obtained feature vectors for feature selection. In this experiment, we have selected the 1000 features from both entropy and ACO-based approaches. Those are later fused to get a feature matrix of size 4536×2000. This feature matrix is fed to the classifiers for classification. In this experiment, C-SVM has attained the best accuracy of 91.99%. and Q.SVM ranks second with an accuracy of 85.71 percent. The results of this experiment are given in Table 4.10. Similarly, in Figure 4.9, the time chart of VGG16+SqueezeNet (Entropy: 1000+ ACO: 1000 features) is shown. It is observed that our method works best on the C-SVM classifier for the CUI-EXAM dataset. However, the performance degrades with this feature combination since the number of characteristics has increased.

Table 4. 10. Results of VGG16+SqueezeNet (Entropy: 1000+ ACO: 1000 features)

Classifier	Percentage of Correctness	Percentage of sensitivity	Percentage of specificity	Percentage of Precision	Percentage of F-measure	G-mean in percent
L.SVM	68.21	75.37	66.65	33.01	45.91	70.88
Q-SVM	85.71	93.47	84.02	56.06	70.08	88.62
FG-SVM	40.59	40.52	40.60	12.95	19.62	40.56
MG-SVM	59.41	57.51	59.83	23.79	33.66	58.66
CG-SVM	74.10	84.48	71.83	39.54	53.87	77.90
C-SVM	91.69	95.69	90.82	69.44	80.48	93.22

Figure 4. 9. VGG16+SqueezeNet (Entropy: 1000+ ACO: 1000 features) time chart

4.7.4. Experiment 6:SqueezeNet Features only (Entropy:750+ACO:750 features)

In this work, we employ the SqueezeNet feature vector to evaluate the suggested method's accuracy. We obtain the feature vector of 4536×750 for the SqueezeNet model. We select 750 features each with Entropy and ACO-based feature selection approaches from the obtained total SqueezeNet feature vector.

The obtained feature matrix with size 4536×1500 is fed to the classifiers for classification. In this experiment, C-SVM has attained the best accuracy of 88.8%. MG-SVM, on the other hand, has the foremost accuracy of 86 percent. Table 4.12 shows the outcomes of this trial.

Table 4. 11. SqueezeNet Results (Entropy: 750+ ACO: 750 features)

Classifier	Accuracy (%)	Sensitivity (%)	Specificity (%)
L-SVM	71.8	71.67	71.69
Q-SVM	83.9	83.89	84.00
FG-SVM	41.8	41.69	41.90
MG-SVM	86.0	85.90	85.90
CG-SVM	54.4	54.39	54.42
C-SVM	91.99	90.82	93.16

4.7.5. Experiment 7: VGG16 features only (Entropy: 750+ ACO: 750 features)

The proposed method's performance is evaluated using the VGG16 feature vector in this experimentation. We obtain the feature vector of 4536×750 for the SqueezeNet model. We select 750 features each with Entropy and ACO-based feature selection approaches from the obtained total VGG16 feature vector. The obtained feature matrix with size 4536×1500 is fed to the classifiers for classification. In this experiment, C-SVM has attained the best accuracy of 91.99%. MG-SVM, on the other hand, does have the furthermore accuracy of 84.9 percent. Table 4.12 shows the outcomes of this experiment.

Table 4. 12. Finding of VGG16 (250 features), in terms of performance measures

Classifier	Accuracy (%)	Sensitivity (%)	Specificity (%)
L-SVM	70.1	69.98	70.00
Q-SVM	81.8	80.99	81.67
FG-SVM	42.2	42.01	41.99
MG-SVM	84.9	84.79	84.69
CG-SVM	52.2	51.88	52.00
C-SVM	91.99	92.01	91.98

4.8. Experimental Results

The experimental results are presented with graphical and numerical values containing proper justification. This section provides the results individually for each step of the

proposed work with different variations and also described the archived results and the methodology we adopted.

This procedure includes the results of the proposed procedure's algorithm which is used to measure the accuracy score for the classification measures using the classifiers of SVM's and KNN's, different experiments are performed on the six classes of students monitoring during the examination on different cheating scenarios. In all of these classes, different experiments are performed with distinct no. of features and distinct combination of features on all of the classifier learners to evaluate or detect the high accuracy score against all the features individually as well as with the combination of different features. There are some characteristics we used for the evaluation of performance measures. This includes: accuracy (ACC) at different folds mainly 5 fold and 10 fold are considered, holdout validation of 50% and 70% are considered, confusion matrix, misclassification rate, TPR, FPR, SEN, SPE, precision, prevalence, ER, F-Score and ROC curves, all of these steps of proposed work are implemented.

4.9. Training/ Testing Procedures

For the performance evaluation of the proposed technique, 5748 images of students' (during the examination) database are trained and tested, these images are trained on all SVM's (Support vector machine) and KNN's classifier learners including all SVM classifiers and KNN classifiers. The results of this performance evaluation against the proposed technique of feature fusion, there is the number of experiments performed using different features both in, individual and combination of multiple features for the evaluation of best feature combination based on their high Accuracy score, the results of all of these combinations are as presented in the next sections.

4.9.1. Benchmark for Classifier Selection Using 5_Fold and 10_Fold

The model is trained with the combination of all given features with parallel computation of different classifiers to take the result percentage and to check the Accuracy score against multiple classifiers including LSVM, QSVM, CSVM, FG, FineGaussian, CGSVM, FineKNN, Medium KNN, CKNN, cubic-KNN and Weighted-KNN, the resultant accuracy score and there training time taken by each classifier to show the results on both fivefold and Tenfold CV accordingly, results on each classifier learner with their training

time, as well as their FNR (false negative rate) at both 5 fold CV and 10-fold CV of resultant accuracies Table 4.13 and Table 4.14, illustrate the results, respectively.

Table 4. 13. Optimized Classifier Selection at 5-Fold

Features	Classifier	Accuracy (%)	Training Time (sec)
Fused Feature SFTA+LBP+Gabor	Linear SVM	57.4	109.32
	Quadratic SVM	68.4	167.26
	Cubic SVM	80.0	168.23
	Fine Gaussian	39.2	276.44
	Medium Gaussian SVM	67.8	267.41
	Coarse Gaussian SVM	51.2	297.79
	Fine KNN	94.0	222.07
	Medium KNN	66.2	276.89
	Coarse KNN	33.5	301.58
	Cosine KNN	75.0	304.10
	Cubic KNN	58.9	1931.7
	Weighted KNN	81.5	322.76
	Linear Discriminant	56.0	7.9836
	Quadratic Discriminant	67.4	7.4460
	Complex Tree	43.6	12.551
	Medium Tree	43.2	9.0218
	Simple Tree	41.1	8.7630
	Ensemble Boosted Trees	45.6	185.58
	Ensemble Bagged Trees	59.7	44.958
	Ensemble Subspace Discriminant	54.6	17.777
	Ensemble Subspace KNN	56.1	163.40
	Ensemble RUSBoosted Trees	39.5	79.502

Table 4. 14. Optimized Classifier Selection at 10_Fold

Features	Classifier	Accuracy (%)	Execution Time (sec)
Fused Feature SFTA+LBP+Gabor	Linear SVM	57.6	66.551
	Quadratic SVM	69.1	106.76
	Cubic SVM	72.1	126.02
	Fine Gaussian	39.6	187.16
	Medium Gaussian SVM	68.3	166.33
	Coarse Gaussian SVM	51.9	183.19
	Fine KNN	94.0	151.99
	Medium KNN	62.2	174.84
	Coarse KNN	43.4	187.53
	Cosine KNN	67.4	194.45
	Cubic KNN	60.0	745.03
	Weighted KNN	67.4	208.23
	Linear Discriminant	56.4	7.8600
	Quadratic Discriminant	68.5	10.400
	Complex Tree	44.4	31.498
	Medium Tree	43.3	22.054
	Simple Tree	41.3	15.954
	Ensemble Boosted Trees	45.5	663.38
	Ensemble Bagged Trees	60.3	309.24
	Ensemble Subspace Discriminant	54.7	236.66
	Ensemble Subspace KNN	58.8	499.93
	Ensemble RUSBoosted Trees	40.1	487.09

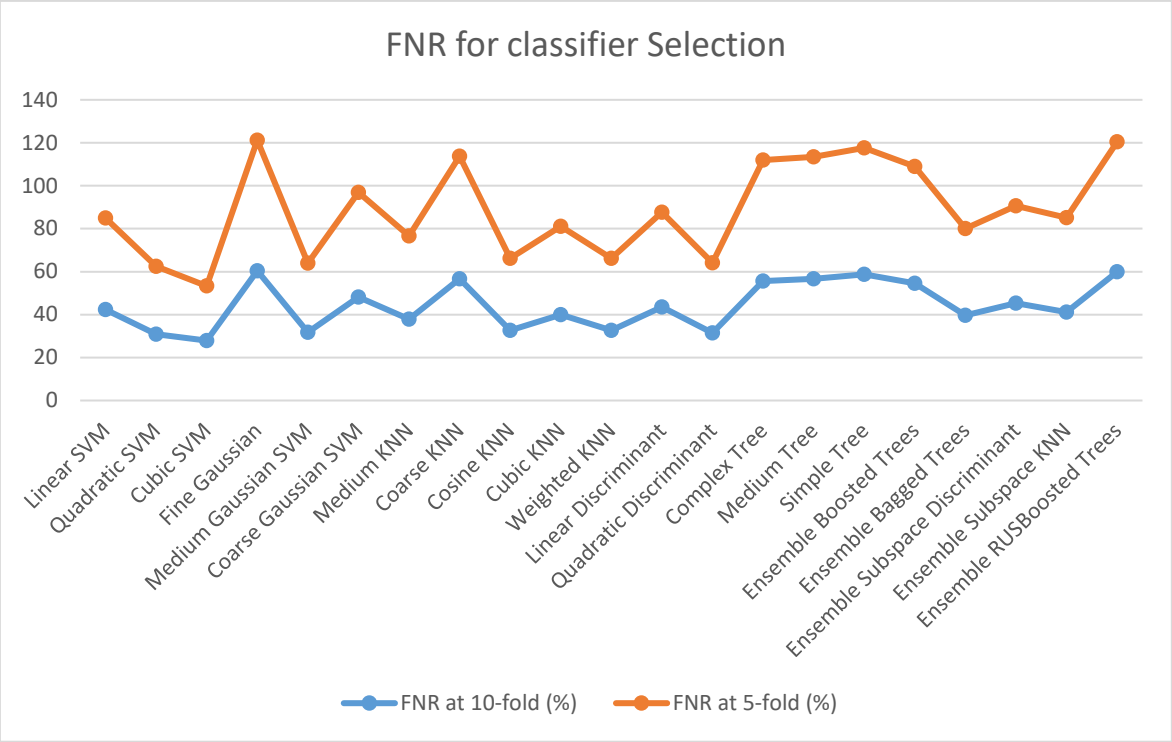

Figure 4. 10. FNR at 5-fold and 10-fold CV for Selection of Optimized Classifier

4.9.2. Benchmark for Feature Selection using 5_Fold and 10_Fold

Based on the above results (Table 4.13 and 4.14), which shows the best classifiers resulting in the high Accuracy score on the combination of features having SFTA, LBP, and Gabor, this table shows that the Fine KNN resulting highest accuracy score and then the Cubic SVM showing second-highest Accuracy score. So, based on their highest Accuracy resultants, we choose two classifiers returning high Accuracy scores. After the selection of best classifiers, now we further proceed towards the best feature combination selection, for this purpose, we perform several experiments on the features individually and on their combinations using multiple features. Different combinations of features are Appling to check Accuracy against the best-selected classifiers that are cubic SVM and fine KNN chosen from the classifier learner having comparatively highest accuracy score against the combinations of all the features. The accuracy score against different combinations of features with given classifiers Cubic SVM and Fine KNN and their execution time, as well as their FNR, are shown as follows in Table 4.15 and 4.16 respectively:

Table 4. 15. Optimized Feature Selection at 5-Fold Cross-Validation

Features	Classifier	Accuracy (%)	Execution Time (sec)
HOG	Cubic SVM	65.3	33.677
	Fine KNN	86.7	3.3011
SFTA	Cubic SVM	65.1	63.826
	Fine KNN	81.3	1.1828
LBP	Cubic SVM	60.8	49.816
	Fine KNN	81.7	40.961
Gabor	Cubic SVM	60.8	30.456
	Fine KNN	64.9	2.2373
HOG + SFTA	Cubic SVM	68.4	31.072
	Fine KNN	89.2	3.7699
HOG + LBP	Cubic SVM	68.8	78.24
	Fine KNN	68.9	91.589
HOG + Gabor	Cubic SVM	68.9	33.041
	Fine KNN	69.4	5.2976
SFTA + LBP	Cubic SVM	67.9	29.202
	Fine KNN	71.7	2.9529
SFTA + Gabor	Cubic SVM	67.9	27.882
	Fine KNN	71.2	2.6299
LBP + Gabor	Cubic SVM	67.8	57.074
	Fine KNN	93	56.08
HOG + LBP + Gabor	Cubic SVM	70.4	32.032
	Fine KNN	93.8	7.2072
HOG + SFTA + Gabor	Cubic SVM	70.7	32.32
	Fine KNN	94.0	5.4982
SFTA + LBP + Gabor	Cubic SVM	71.1	186.23
	Fine KNN	73.8	222.07
HOG + LBP + SFTA	Cubic SVM	70.6	72.34
	Fine KNN	93.9	74.368
HOG + SFTA + LBP + Gabor	Cubic SVM	71.9	34.599
	Fine KNN	93.5	6.9369

Table 4. 16. Optimized Feature Selection at 10-Fold Cross-Validation

Features	Classifier	Accuracy (%)	Execution Time (sec)
HOG	Cubic SVM	65.3	64.727
	Fine KNN	89.0	3.6373
SFTA	Cubic SVM	67.0	143.22
	Fine KNN	94.0	1.1729
LBP	Cubic SVM	60.5	69.63
	Fine KNN	82.5	1.7613
Gabor	Cubic SVM	62.2	63.356
	Fine KNN	85.0	2.6443
HOG + SFTA	Cubic SVM	69.4	59.522
	Fine KNN	70.8	4.4587
HOG + LBP	Cubic SVM	69.5	61.888
	Fine KNN	88.0	4.8048
HOG + Gabor	Cubic SVM	69.9	59.992
	Fine KNN	80.0	5.7239
SFTA + LBP	Cubic SVM	69.1	61.099
	Fine KNN	82.4	2.5273
SFTA + Gabor	Cubic SVM	68.8	56.963
	Fine KNN	92.7	2.9948
LBP + Gabor	Cubic SVM	68.5	57.605
	Fine KNN	80.9	3.7966
HOG + LBP + Gabor	Cubic SVM	71.3	61.83
	Fine KNN	72.0	8.0122
HOG + SFTA + Gabor	Cubic SVM	71.7	60.267
	Fine KNN	82.2	6.2633
SFTA + LBP + Gabor	Cubic SVM	72.1	126.02
	Fine KNN	84.6	151.99
HOG + LBP + SFTA	Cubic SVM	71.7	61.503
	Fine KNN	82.4	6.6532
HOG + SFTA + LBP + Gabor	Cubic SVM	72.8	63.078
	Fine KNN	93.9	7.9951

After compressing results, we concluded the fused features resulted in high accuracy score as compared to the individual ones. For this purpose, we tested four feature descriptors as HOG, SFTA, LBP, and Gabor. First, we took results on these descriptors individually,

but on an individual basis, they did not perform too well, for further improvement we fused these feature descriptors and get the scores on both 5-fold and 10-fold Cross-Validations.

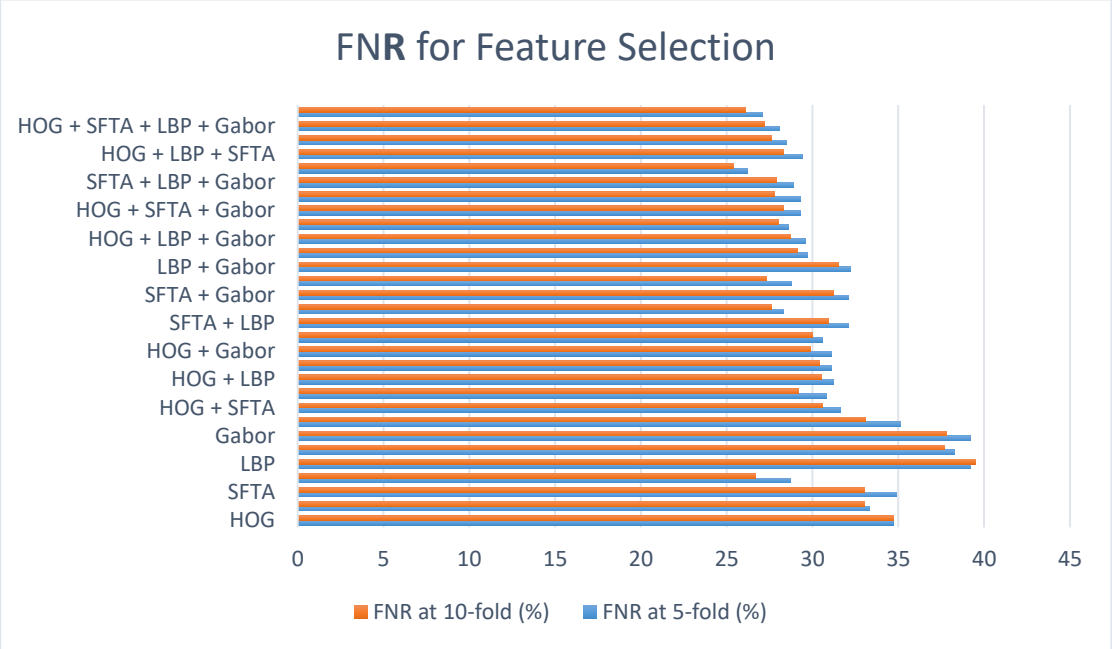

Figure 4. 11. FNR at 5-Fold and 10-Fold CV for the Selection of Optimized Features

This results in us the best combination of feature descriptors of the highest returning Accuracy score of the classifiers of Cubic SVM and Fine KNN.Table. 4.17 shows that the Combination of SFTA, LBP, and Gabor result in us the highest Accuracy score on both 5-fold and 10-fold Cross-Validations also having minimum error rate so that we conclude that the combination of SFTA, LBP, and Gabor perform well than the others, and then HOG, SFTA, LBP, and Gabor shows second highest and the third highest combination is SFTA and LBP.

4.9.3. Holdout at High-Scored Feature and Classifier

After taking the results against the best classifiers, the best features are selected after calculating the results on all possible combinations of features, we obtained that the best Accuracy rate against the resultant combinations, further we check Accuracy at 5 folds and 10 folds same as we take results of Accuracy rate through the validation checking at 50 % holdout validation and 75% holdout validation. Accuracy scores were achieved at various accuracy rates also the ER and validation variation are shown in the following (Table 4.17).

Table 4. 17. Holdout Validation at Best 3 Classifiers and Features Combination at 5-Fold

Features	Classifier	Accuracy	Holdout Validation at 50%	Holdout Validation at 75%
HOG+SFTA+ LBP+ Gabor	Cubic SVM	71.9%	72.9%	69.3%
	Fine KNN	92.9%	93.8%	91.4%
SFTA+ LBP+ Gabor	Cubic SVM	71.1%	69.4%	68.1%
	Fine KNN	93.8%	93.7%	91.8%
SFTA+LBP	Cubic SVM	67.9%	68.4%	67.0%
	Fine KNN	91.7%	91.1%	93.8%

These experiments show that all of these combinations result in high accuracy score at the fine KNN classifier, as well as the best-experienced combination of features having SFTA, LBP, and Gabor shows high performance across both cross-validations but on the holdout validation, the combination of HOG, SFTA, LBP, and Gabor show comparatively high accuracy results than the other two combinations.

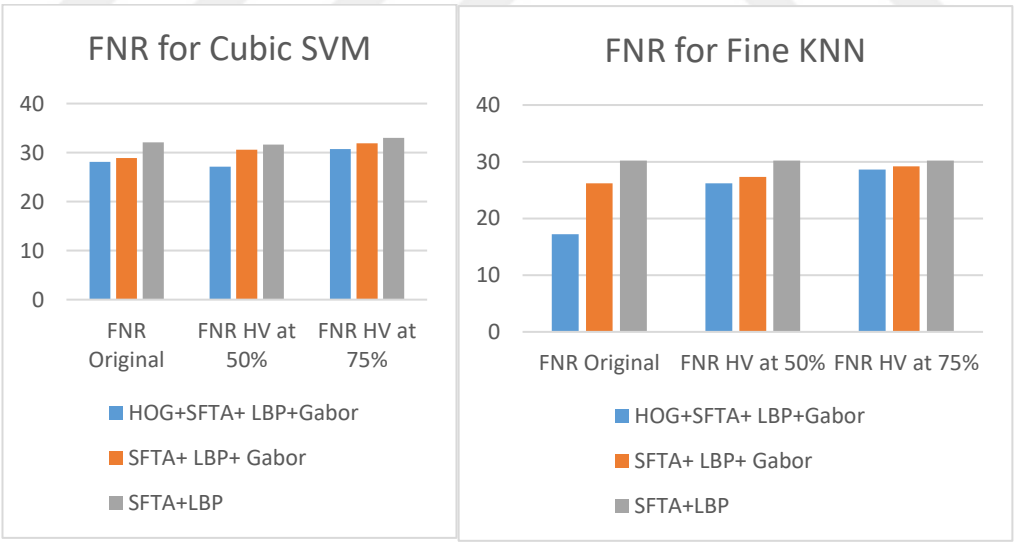

Figure 4. 12. FNR readings for Holdout Validation at Best 3 Classifiers and Features Combination at 5-Fold

By comparing the results of all of these combinations (shown in Table. 4.17), the combination of SFTA, LBP, and Gabor resulting a high Accuracy score, based on these results we select this combination for further processing.

After performing several experiments, we conclude that the combination of SFTA, LBP, and Gabor resulted in us the comparative best results so that we selected this combination for further processing. Now, we further perform the selection of no. of features and on these no. of features we perform different observations for the best resultant accuracy score, we perform these observations on 5-fold CV, we summarized the experimental observations which are given in Table 4.18.

Table 4. 18. Summary of Experimental Observations

Experiments	Classes	No. of Features			Total Features	Classifier	Accuracy (%)	Execution Time (sec)
		SFTA	LBP	Gabor				
Experiment. 1	6	20	30	30	80	Fine KNN	73.3	2.305
						Cubic SVM	69.1	27.876
Experiment. 2	6	18	35	50	103	Fine KNN	93.4	3.1822
						Cubic SVM	70.7	29.281
Experiment. 3	6	21	40	60	121	Fine KNN	93.8	3.7798
						Cubic SVM	71.1	30.447
Experiment. 4	6	19	45	55	119	Fine KNN	73.0	3.7594
						Cubic SVM	70.5	28.551
Experiment. 5	6	20	25	45	90	Fine KNN	93.5	2.8752
						Cubic SVM	70.5	28.295

As we perform different experiments on the resultant combination of features with both classifiers including Cubic SVM and Fine KNN. These experiments show the best accuracy score on Experiment. 3. The comparison of both of the classifiers is shown in graphical representation.

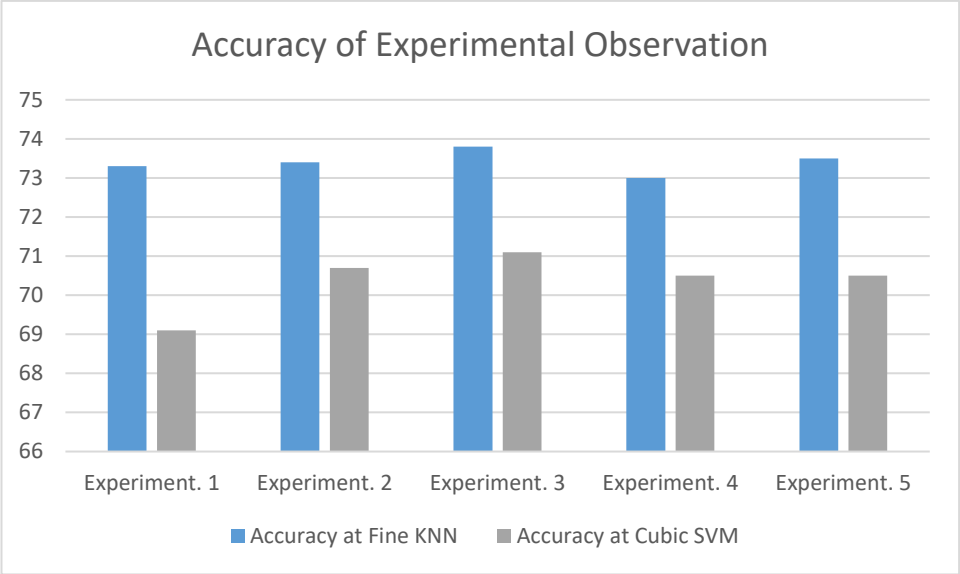

Figure 4. 13. Accuracy Comparison on both Cubic SVM and Fine KNN Classifiers

4.9.4. Confusion Metrix Readings on Resultant Feature and Classifier

To check the different performance measures including accuracy at different folds and Holdout Validation, these measures include accuracy result, sensitivity result, specificity, precision result, TPR, FPR, F-Score, confusion matrix, misclassification rate, and ROC curves, and using these ROC curves we further calculated area under the curve (AUC). all of these steps of proposed work that are implemented, CM curve provides the best measures of all performance evaluations.

The confusion matrix presented in Table 4.8 is the resultant of high accuracy measure, this confusion matrix can be used to calculate the FPR and AUC for all the classes including back watching, front watching, normal, side watching, showing gestures, and suspicious watching separately.

5. CONCLUSION AND FEATURE WORK

5.1. Result and Discussion

The major purpose of our research is to develop a CNN structure for tackling the supplied dataset. The recommended Entropy and ACO CNN Network are only used to extract powerful features after feature selection. Our dataset is used for the pretraining.

The proposed approach utilizes VGG16 and SqueezeNet based features. The features are fused, and then feature selection algorithms based on ACO and entropy are used to the fused feature matrix. The single feature matrix is created by combining the selected features. Different approaches are followed to finalize the proposed architecture in Figure 3.1. The foremost approaches include fine-tuning, adding, and removing different features. In its final form, the 750, entropy, and 750 ACO feature combination are found good with the best outcomes in terms of performance.

For prediction, this feature matrix is supplied into SVM-based classifiers. ACO and entropy-based feature selection methods are used in a variety of experiments with various amounts of selected features. It is observed that C-SVM proves best in all the experiments. The best accuracy of 91.99% is attained with 1500 features (Entropy: 750+ ACO: 750 features) by the C-SVM classifier. Figure 5.1 represents the comparison of accuracies based on the number of features.

The discussion and interpretation of the outcomes of the proposed framework are described in this portion. The dataset is first collected and defined in this thesis. The procedure for evaluating performance is then portrayed. Finally, the tests are well discussed. All the mentioned experiments in this thesis are conducted on a Pentium core i-5 system containing 8 GB of memory. The training is aided with NVIDIA GTX 1070 GPU comprising 8GB RAM. The coding is performed by using MATLAB 2019a.

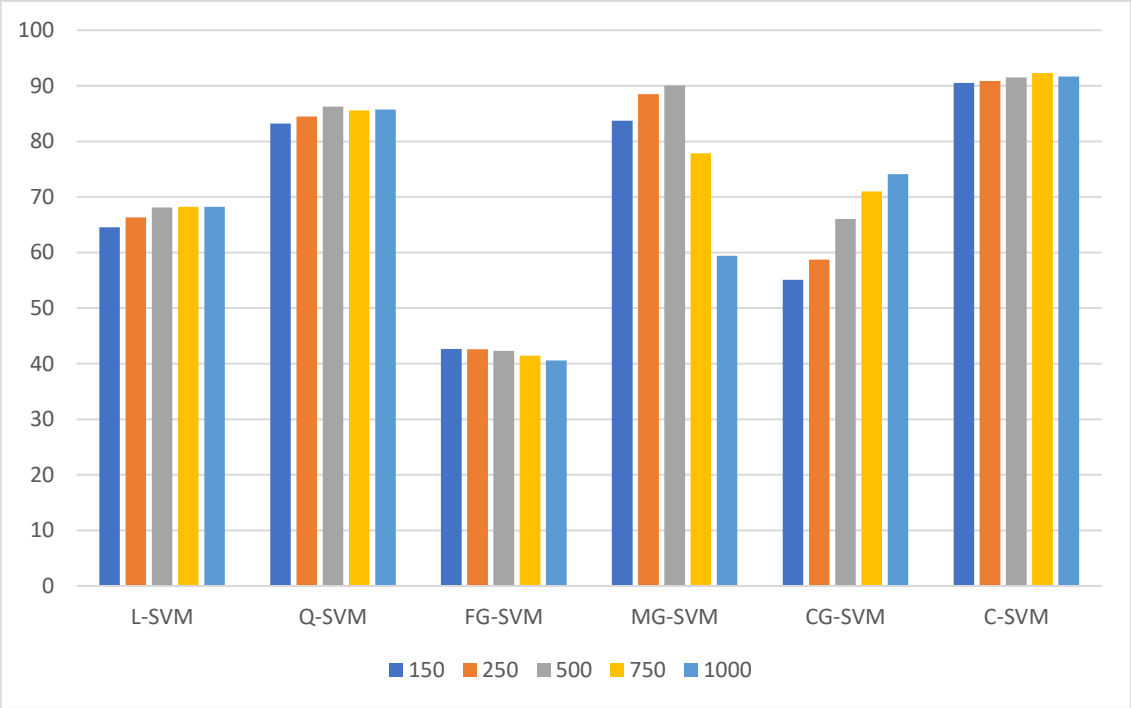

Figure 5. 1. Comparison of accuracies based on the number of features

5.2. Conclusion

The purpose of this study is to label student activities on the exam using cropped images from surveillance cameras. The images are put into VGG-16 and squeeze net at the same time to generate deep features.

To enhance multi-modal matching performance for cheating monitoring and categorizing different scenarios of cheating and classification from pre-labeled images have been proposed. The proposed model is based on four pipeline procedures such as pre-processing, -feature extraction, feature selection, and classification. From all of the above results, we conclude that using the above-mentioned proposed strategy, the classification of different cheating scenarios is tackled and performed well on the given dataset. Moreover, using the above results section, it is concluded that the scale oriented and texture features have vital importance for the classification of all of the selected classes such as back watching, front watching, side watching, normal, showing gestures, and suspicious watching.

To generate a single vector, the deep features are fused serially. ACO and an entropy-based feature subset obtaining approach are utilized to get feature subsets from the fused feature vector. After that, the two feature subsets are serially fused to create an ensembled

feature vector. For prediction, SVM-based Classification is applied to this ensembled vector. The CUI-EXAM dataset has been trained or modeled. We performed extensive experiments with various combinations of selected features. We observe that the combination of selected features with 1500 features (Entropy: 750+ ACO: 750 features) provides the best classification results with an accuracy of 91.99%. The satisfactory outcomes demonstrate the system's reliability.

Regarding Performance evaluation protocol, the 5-folds and 10-folds mechanism for the cross-validation is employed for booth learning and assessment. To assess the efficacy of the proposed research, various evaluation assessment procedures were used in this research. The confusion matrix created during the classification challenge's testing process is used in the majority of these protocols. 5-fold cross-validation is used for ground reality.

Although the results are acceptable, still the performance can further be enhanced. In the future, the performance of exam monitoring can be improved by exploring more feature selection approaches as well as deep learning methods with more big datasets.

Extensive testing of various iterations of selected features is carried out to find the best results. A few main reviews are listed in this section. Table 3.2 provides a succinct summary of the accuracy of each test discussed. At the feature selection stage, several tests are run with various numbers of features. Table 3.3 depicts the results of improved method parameter values used for the classification optimization approach. Figure 3.6 illustrates the representation of feature fusion. Table 4.7. shows the results of VGG16+SqueezeNet (Entropy: 750+ ACO: 750 features)

5.3. Future Work

In the future, the performance of exam monitoring can be improved by exploring more feature selection approaches as well as deep learning methods with more big datasets. This work aims at suggesting a rough guideline for novices and experienced researchers who have interest in deep learning methods for student activity detection and labeling in the exam. In this work, we presented the dataset, feature extraction techniques, the framework for labeling the features, a novel semi-supervised federated learning framework for student activities detection and labeling in the exam. To the best of our knowledge, EUI-EXAM is the first dataset collected for student activity labeling in the exam. In future work, we plan to use features such as sound detection, pre-exam environmental set-up, student

authentication for sitting for the exam, adjusting the surveillance camera depending on the type of exam, exam taker level, and exam halls.

Indeed, CUI-EXAM is more effective when the collaborative model is trained considering users with similar physical traits and other human activity recognition algorithms. It will be also better if Transfer learning and LTSM is used for text-based video description of student activities. This approach would require adapting other video description techniques to automatically group students activities who share similar characteristics and, at the same time, label their activities.

Arsi University
ዐርሲ ዩኒቨርሲቲ

የቢዝነስና ኢኮኖሚክስ ኮሌጅ ዲን ጽ/ቤት
Business & Economics College Dean office

ቁጥር/Ref. No: CoBE-061/2014
ቀን/Date: 11/11/2021

Lema Teshome Beyecha (Asst. Prof.)
Dean, College of Business and Economics

To: Whom it May Concern

Subject: Professional Ethics Report

This is too kindly inform you that photographs and videos taken at class by our University staff and PhD Candidate Musa Dima Genemo belongs to him and the pictures and videos are taken by our permission.

In the meantime, we would like to stress that the studentents' pictures and videos can only be used for the PhD dissertation. Any use of the pictures/videos for other purposes is strictly forbidden!

Sincerely

Lema Teshome Beyecha (Asst. Prof.)
Dean, College of Business and Economics

- Mr. Musa Dima Genemo
- CoBE Dean Office
  Asella

Tel: +251-228-119-481
+251-911-750-800

Arsi University
Ethiopia

E-mail: getlemma@gmail.com
Website: www.arsiun.edu.et

6. REFERENCES

1. J. Nishchal, et al., "Automated Cheating Detection in Exams using Posture and Emotion Analysis," in 2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), 2020, pp. 1-6.

2. M. Asadullah and S. Nisar, "An automated technique for cheating detection," in 2016 Sixth International Conference on Innovative Computing Technology (INTECH), 2016, pp. 251-255.

3. G. Migut, et al., "Cheat me not: automated proctoring of digital exams on bring-your-own-device," in Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education, 2018, pp. 388-388.

4. K. Butler-Henderson and J. Crawford, "A systematic review of online examinations: A pedagogical innovation for scalable authentication and integrity," Computers & Education, p. 104024, 2020.

5. M. Raza, et al., "Appearance-based pedestrians' head pose and body orientation estimation using deep learning," Neurocomputing, vol. 272, pp. 647-659, 2018.

6. M. A. Khan, et al., "Hand-crafted and deep convolutional neural network features fusion and selection strategy: an application to intelligent human action recognition," Applied Soft Computing, vol. 87, p. 105986, 2020.

7. L. Chen and C. Nugent, "Ontology-based activity recognition in intelligent pervasive environments," International Journal of Web Information Systems, vol. 5, pp. 410-430, 2009.

8. K. Soomro, et al., "UCF101: A dataset of 101 human actions classes from videos in the wild," arXiv preprint arXiv:1212.0402, 2012.

9. J. Naz, et al., "Detection and Classification of Gastrointestinal Diseases using Machine Learning," Current Medical Imaging, 2020.

10. Z. Wu, et al., "Multi-Stream Multi-Class Fusion of Deep Networks for Video Classification," Current Medical Imaging, pp. 791-800, 2016.

11. M. Sekma, et al., "Human action recognition based on multi-layer Fisher vector encoding method," Pattern Recognition Letters, vol. 65, pp. 37-43, 2015.

12. H. J. Zhang, et al., "An integrated system for content-based video retrieval and browsing," Pattern Recognition, vol. 30, pp. 643-658, 1997.

13. Q. Zhang, et al., "An Efficient Deep Learning Model to Predict Cloud Workload for Industry Informatics," IEEE Transactions on Industrial Informatics, vol. 14, pp. 3170-3178, 2018.

14. Z. Wang, et al., "Image quality assessment: from error visibility to structural similarity," IEEE transactions on image processing, vol. 13, pp. 600-612, 2020.

15. Á. Peris, et al., "Video description using bidirectional recurrent neural networks," in International Conference on Artificial Neural Networks, 2016, pp. 3-11.

16. Z. Xu, et al., "Learning temporal features using LSTM-CNN architecture for face anti-spoofing," in 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR), 2015, pp. 141-145.

17. J. Song, et al., "Hierarchical LSTM with adjusted temporal attention for video captioning," arXiv preprint arXiv:1706.01231, 2017.

18. B. Hasani and M. H. Mahoor, "Facial expression recognition using enhanced deep 3D convolutional neural networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 30-40.

19. J. Amin, et al., "Brain tumor detection: a long short-term memory (LSTM)-based learning model," Neural Computing and Applications, vol. 32, pp. 15965-15973, 2020.

20. G. Meditskos, et al., "Multi-modal activity recognition from egocentric vision, semantic enrichment and lifelogging applications for the care of dementia," Journal of Visual Communication and Image Representation, vol. 51, pp. 169-190, 2018.

21. K. Wang, et al., "3D Human Activity Recognition with Reconfigurable Convolutional Neural Networks," pp. 97-106, 2014.

22. S. Venugopalan, et al., "Improving LSTM-based video description with linguistic knowledge mined from text," arXiv preprint arXiv:1604.01729, 2016.

23. X. Huang, et al., "Lung nodule detection in CT using 3D convolutional neural networks," in 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), 2017, pp. 379-383.

24. E. Hosseini-Asl, et al., "Alzheimer's disease diagnostics by adaptation of 3D convolutional network," in 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 126-130.

25. S. Xie, et al., "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 305-321.

26. S. Hershey, et al., "CNN architectures for large-scale audio classification," in 2017 IEEE international conference on acoustics, speech and signal processing (icassp), 2017, pp. 131-135.

27. M. Xin, et al., "ARCH: Adaptive recurrent-convolutional hybrid networks for long-term action recognition," Neurocomputing, vol. 178, pp. 87-102, 2016.

28. B. Morgenstern, "DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment," Bioinformatics (Oxford, England), vol. 15, pp. 211-218, 1999.

29. F. Iandola, et al., "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. February 2016," arXiv preprint arXiv:1602.07360, 2019.

30. colony optimization feature selection and extreme learning machine," Neurocomputing, vol. 226, pp. 66-79, 2017.

31. S.-C. Hsu, C.-H. Chuang, C.-L. Huang, R. Teng, and M.-J. Lin, "A video-based abnormal human behavior detection for psychiatric patient monitoring," in 2018 International Workshop on Advanced Image Technology (IWAIT), 2018, pp. 1-4.

32. Z. Zhigang, D. Guangxue, L. Huan, Z. Guangbing, W. Nan, and Y. Wenjie, "Human behavior recognition method based on double-branch deep

convolution neural network," in 2018 Chinese Control And Decision Conference (CCDC), 2018, pp. 5520-5524.

33. A. Jalal, M. Maria, and M. Sidduqi, "Robust Spatio-temporal features for human interaction recognition via an artificial neural network," in IEEE Conference on International Conference on Frontiers of information technology, 2018.

34. R. K. Tripathi, A. S. Jalal, and S. C. Agrawal, "Suspicious human activity recognition: a review," Artificial Intelligence Review, vol. 50, pp. 283-339, 2018.

35. M. Irfan, L. Tokarchuk, L. Marcenaro, and C. Regazzoni, "ANOMALY DETECTION IN CROWDS USING MULTI SENSORY INFORMATION," in 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2018, pp. 1-6.

36. H. Zhang, "3D Robotic Sensing of People: Human Perception, Representation and Activity Recognition," 2014.

37. M. M. Hassan, S. Huda, M. Z. Uddin, A. Almogren, and M. Alrubaian, Floating search methods in feature selection, Pattern Recognition Letters, 15, 11 (1994), 1119-1125.

38. A. K. Singh and V. Bansal, "SVM Based Approach for Multiface Detection and Recognition in Static Images," Journal of Image Processing and Artificial Intelligence, vol. 4, 2018.

39. X. Sun, P. Wu, and S. C. Hoi, "Face detection using deep learning: An improved faster RCNN approach," Neurocomputing, vol. 299, pp. 42-50, 2018.

40. R. Ranjan, V. M. Patel, and R. Chellappa, "Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, pp. 121-135, 2019.

41. S. J. Elias, S. M. Hatim, N. A. Hassan, L. M. A. Latif, R. B. Ahmad, M. Y. Darus, et al., "Face Recognition Attendance System Using Local Binary Pattern (LBP)," Bulletin of Electrical Engineering and Informatics, vol. 8, 2019.

42. P. C. Brewer, D. Eubanks, H. Gupta, W. A. Scanlon, P. L. Venetianer, W. Yin, et al., "Object tracking and best shot detection system," ed: Google Patents, 2018.

43. N. Danielsson and A. Hansson, "Method and system for tracking an object in a defined area," ed: Google Patents, 2018.

44. K. Li, F.-Z. He, and H.-P. Yu, "Robust Visual Tracking based on convolutional features with illumination and occlusion handling," Journal of Computer Science and Technology, vol. 33, pp. 223-236, 2018.

45. J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from RGB images," in 2012 IEEE international conference on robotics and automation, 2012, pp. 842-849.

46. F. Baradel, C. Wolf, J. Mille, and G. W. Taylor, "Glimpse clouds: Human activity recognition from unstructured feature points," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 469-478.

47. Y. S. Chong and Y. H. Tay, "Abnormal event detection in videos using a spatiotemporal autoencoder," in International Symposium on Neural Networks, 2017, pp. 189-196.

48. Y. Kim and T. Moon, "Human detection and activity classification based on micro-Doppler signatures using deep convolutional neural networks," IEEE geoscience and remote sensing letters, vol. 13, pp. 8-12, 2015.

49. M. Sabokrou, M. Fayyaz, M. Fathy, Z. Moayed, and R. Klette, "Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes," Computer Vision and Image Understanding, vol. 172, pp. 88-97, 2018.

50. C. Lu, J. Shi, W. Wang, and J. Jia, "Fast Abnormal Event Detection," International Journal of Computer Vision, pp. 1-19, 2018.

51. X. Bian, C. Chen, L. Tian, and Q. Du, "Fusing local and global features for high-resolution scene classification," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 10, pp. 2889-2901, 2017.

52. P. Xiong, G. Li, and Y. Sun, "Combining local and global features for 3D face tracking," in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2529-2536.

53. K. T. Ahmed, A. Irtaza, and M. A. Iqbal, "Fusion of local and global features for effective image extraction," Applied Intelligence, vol. 47, pp. 526-543, 2017.

54. G. Lee, Y.-W. Tai, and J. Kim, "Deep saliency with encoded low-level distance map and high-level features," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 660-668.

55. Q. V. Le, M. A. Ranzato, R. Monga, M. Devin, K. Chen, G. S. Corrado, et al., "Building high-level features using large scale unsupervised learning," arXiv preprint arXiv:1112.6209, 2011.

56. W. Yang, X. Yin, and G.-S. Xia, "Learning high-level features for satellite image classification with limited labeled samples," IEEE Transactions on Geoscience and Remote Sensing, vol. 53, pp. 4472-4482, 2015.

57. L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 801-818.

58. R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 586-595.

59. N. Serrano, A. E. Savakis, and J. Luo, "Improved scene classification using efficient low-level features and semantic cues," Pattern Recognition, vol. 37, pp. 1773-1784, 2004.

60. R. Zhang, P. Isola, A. Efros, E. Shechtman, and O. Wang, "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric," in CVPR, 2018.

61. C. Zhang, X. Pan, H. Li, A. Gardiner, I. Sargent, J. Hare, et al., "A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 140, pp. 133-144, 2018.

62. M. Zang, D. Wen, T. Liu, H. Zou, and C. Liu, "A pooled Object Bank descriptor for image scene classification," Expert Systems with Applications, vol. 94, pp. 250-264, 2018.

63. S. Vasamsetti, S. Setia, N. Mittal, H. K. Sardana, and G. Babbar, "Automatic underwater moving object detection using multi-feature integration framework in complex backgrounds," IET Computer Vision, vol. 12, pp. 770-778, 2018.

64. H. F. Ng and C. Y. Chin, "Effective scene change detection in complex environments," International Journal of Computational Vision and Robotics, vol. 9, pp. 310-328, 2019.

65. K. Lenc and A. Vedaldi, "Large scale evaluation of local image feature detectors on homography datasets," arXiv preprint arXiv:1807.07939, 2018.

66. S. Pang, J. Ma, J. Xue, J. Zhu, and V. Ordonez, "Image retrieval using heat diffusion for deep feature aggregation," arXiv preprint arXiv:1805.08587, 2018.

67. T. Kalsum, S. M. Anwar, M. Majid, B. Khan, and S. M. Ali, "Emotion recognition from facial expressions using hybrid feature descriptors," IET Image Processing, vol. 12, pp. 1004-1012, 2018.

68. R. A. Shaikh, I. Memon, R. Hussain, A. Maitlo, and H. Shaikh, "A contemporary approach for object recognition based on the spatial layout and low-level features' integration," Multimedia Tools and Applications, pp. 1-24, 2018.

69. R. Al-Akam and D. Paulus, "Local and Global Feature Descriptors Combination from RGB-Depth Videos for Human Action Recognition," in ICPRAM, 2018, pp. 265-272.

70. S. Hore, S. Chatterjee, S. Chakraborty, and R. K. Shaw, "Analysis of different feature description algorithm in object recognition," in Computer Vision: Concepts, Methodologies, Tools, and Applications, ed: IGI Global, 2018, pp. 601-635.

71. C. G. Harris and M. Stephens, "A combined corner and edge detector," in Alvey vision conference, 1988, pp. 10-5244.

72. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of computer vision, vol. 60, pp. 91-110, 2004.

73. A. Baumberg, "Reliable feature matching across widely separated views," in Proceedings, IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662), 2000, pp. 774-781.

74. N. Alamgir, K. Nguyen, V. Chandran, and W. Boles, "Combining multi-channel color space with local binary co-occurrence feature descriptors for accurate smoke detection from surveillance videos," Fire Safety Journal, vol. 102, pp. 1-10, 2018.

75. N. Tajbakhsh and K. Suzuki, "Comparing two classes of end-to-end machine-learning models in lung nodule detection and classification: MTANNs vs. CNNs," Pattern Recognition, vol. 63, pp. 476-486, 2017.

76. M. Sajjad, A. Ullah, J. Ahmad, N. Abbas, S. Rho, and S. W. Baik, "Integrating salient colors with rotational invariant texture features for image representation in retrieval systems," Multimedia Tools and Applications, vol. 77, pp. 4769-4789, 2018.

77. J. Bian, W.-Y. Lin, Y. Matsushita, S.-K. Yeung, T.-D. Nguyen, and M.-M. Cheng, "GMS: grid-based motion statistics for fast, ultra-robust feature correspondence," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4181-4190.

78. D. Dooley, B. McGinley, C. Hughes, L. Kilmartin, E. Jones, and M. Glavin, "A blind-zone detection method using a rear-mounted fisheye camera with a combination of vehicle detection methods," IEEE Transactions on Intelligent Transportation Systems, vol. 17, pp. 264-278, 2015.

79. M. Zuliani, C. Kenney, and B. Manjunath, "A mathematical comparison of point detectors," in **2004 CCPR Recognition Workshop**, 2004, pp. 172-172.

80. Z. Boulkenafet, J. Komulainen, and A. Hadid, "Face antispoofing using speeded-up robust features and fisher vector encoding," IEEE Signal Processing Letters, vol. 24, pp. 141-145, 2016.

BIOGRAPHY

He graduated from primary and secondary school in Lode Hetosa Woreda. In 2012, I received a B.Sc. in Computer Science from Jimma University, and in 2015, I received an M.Sc. in Software Engineering from Adama Science and Technology University. In 2017, I began Computer Science Doctoral Studies at Karadeniz Technical University. Since 2013, I've been a lecturer at Arsi University in Ethiopia's computer science department.

Publications

M. D. Genemo and M. Ekinci, Monitoring of exams activities by employing ensemble feature selection on fused deep features, AMA, Agricultural Mechanization in Asia, Africa, and Latin America, Volume 51, Issue 01, June 2021, Page 929-948

M. D. Genemo and M. Ekinci, A deep learning-based algorithm for monitoring cheating in exams, AMA, Agricultural Mechanization in Asia, Africa, and Latin America, Volume 52, Issue 01, October 2021