International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

An Online Exam Proctoring System Using The GMP-DCNN Approach for the Education Sector

Raksha Puthran[1], Anusha Prashanth Shetty*[2]

Submitted:11/03/2024 **Revised:** 26/04/2024 **Accepted:** 03/05/2024

***Abstract*:** For the education sector, online examination is an effective tool. However, it has many security issues. Thus, various techniques were developed in prevailing research works. But the performance is still lacking. For solving this issue, a Geometric Mean Pooling-based Deep Convolutional Neural Network (GMP-DCNN)-based Online Exam Proctoring (OEP) system is proposed in this paper. Primarily, video, audio, screen recorder, and app setting screenshots are considered as the input. Next, frame conversion, Kendall Rank Correlated Diamond Search (KRCDS), and Weiner Filter (WF) techniques pre-process the video data. Then, by using the Davies Bouldin Score-based K-Means (DBS-KM) algorithm, the objects are segmented. The face points are identified from the detected objects by using Viola Jones (VJ). Subsequently, the features are extracted from the objects and face points. On the other side, by utilizing WF, the noise is removed from the audio signal. Next, from the noise-removed signal, features are extracted. Next, pre-processing and feature extraction phases are also carried out from the screen recorder. The app setting screenshot was also extracted; from the app setting screenshot, the features were also extracted. By utilizing Schaeffer Weighted Kookaburra Optimization (SWKO), significant features are selected from the extracted features. Next, selected features and all the pre-processed data are inputted to the GMP-DCNN. An alert message is sent to the invigilator if any misbehavior is present. Experimental analysis shows that GMP-DCNN achieves 98.8% accuracy.

Keywords: *Online examination, Online Exam Proctoring (OEP), Schaeffer Weighted Kookaburra Optimization (SWKO), Geometric Mean Pooling based Deep Convolutional Neural Network (GMP-DCNN), Kendall Rank Correlated Diamond Search (KRCDS), Davies Bouldin Score based K-Means (DBS-KM), Weiner Filter (WF), and Viola Jones (VJ).*

1. Introduction

Due to widespread threats, offline classes are not possible during the pandemic period (Kaddoura et al., 2022). The entrance exam and hiring process are also affected (Nigam et al., 2021). Thus, to teach and conduct exams for students, the education sector uses the online tool (Ganidisastra & Bandung, 2021). Moreover, several loopholes regarding integrity and security are created by the online examination (Muzaffar et al., 2021). For instance, the students may ask for help from a third party to write the exam (Li et al., 2021). Thus, to minimize the misbehavior activity of the student, automated misbehavior activity detection is required (Garg et al., 2020). Online proctor system is used by many research institutes (Turani et al., 2020).

For the invigilators, the proctor system is highly helpful to invigilate remotely (Motwani et al., 2021). For determining whether the candidate is seeking or taking help in solving the problem from others, the proctor system is highly helpful (Pandey et al., 2020). Many researchers focused on the security of online exam protocols, due to the rapid growth of wireless communication technologies in education (Tiong & Lee, 2021). However, there is still a research gap. Thus, a novel method-based OEP system is proposed in this paper.

[1]*Assistant Professor Gd-II, Department of Master of Computer Application, NMAMIT Nitte (Deemed to be University), Nitte*
Email: raksha.puthran@nitte.edu.in
[2]*Assistant Professor Gd-I, Department of Master of Computer Application, NMAMIT Nitte (Deemed to be University), Nitte*
*Corresponding Author Email: anusha.prashanth@nitte.edu.in

1.1 Problem statement

Existing research work's limitations are,

- None of the prevailing works considered the remote control (i.e., other device connection)-based proctoring system in exams.
- Existing (Nurpeisova et al., 2023) had the complexity issue during proctoring monitoring.
- The method used in existing (Kaddoura&Gumaei, 2022) was unreliable for the other behavior of the student activity.
- Existing (Yulita et al., 2023) considered only a front camera-based proctoring monitoring system, which might provide a misclassification outcome.

For solving these issues, the presented research derives the following objectives:

- To utilize the app setting screenshot for misbehavior detection.
- To use the KRCDS technique for the motion estimation from the screen recorder.
- To consider the face, audio, screen recorder, and app setting screenshot for monitoring.
- To consider both front and back camera input.

The propounded research work's structure is arranged as: The existing OEP system is explicated in Section 2, the proposed GMP-DCNN-based OEP system is described in Section 3, the experimental analysis is explained in Section 4, and the paper is concluded with future enhancement in Section 5.

2. Related work

(Nurpeisova et al., 2023) suggested an artificial intelligence-based online control proctoring system. It included the features of face detection, audio capture, face tracking, and the active capture of system windows. When compared to other traditional models, this model achieved better performance. (Kaddoura & Gumaei, 2022) aimed to develop an effective technique for OEP. The useful features of visual images were utilized in this technique. As per the experimental outcome, the model attained higher performance than the other models. (Yulita et al., 2023) propounded a computer program for detecting cheating in online examinations. Based on the deep learning models, cheating was detected. According to the experimental evaluation, the model achieved a higher F-score value. (Labayen et al., 2021) suggested a diverse biometric technology-centric authentication system for an automatic proctoring system. The outcomes exhibited that the system was fully automatic and achieved better performance in OEP. However, the face image had light and posed illumination issues, thus affecting the model's performance. (Fidas et al., 2023) recommended an open-source source intelligent and continuous student identity management system. The system was grounded on the intelligent face as well as voice identification mechanisms; moreover, it scored well in user experience and usability. But the face and voice identification mechanism suffered from the generic problem of liveness.

3. Proposed GMP-DCNN-Based OEP

In the proposed technique, four types of input are considered for the proctoring system. Figure 1 exhibits the block diagram for the GMP-DCNN-centric proctoring system.

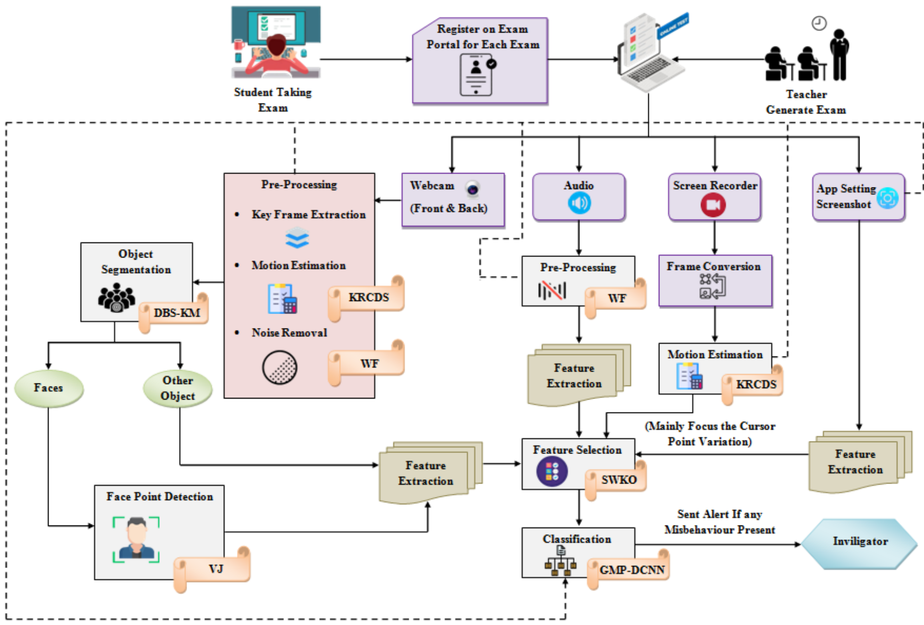


Figure 1: Block diagram for the proposed OEP

3.1 Students and their registration

Primarily, the students register their information into the exam portal.

$$S(R_s) = \{s(r_1), s(r_2), \ldots\ldots, s(r_n)\} \tag{1}$$

Where, $S(R_s)$ is the student registration set and $s(r_n)$ is the n-number of registered students. Webcam, audio, screen recorder, and app setting screenshots are considered during the examination for the OEP.

3.2 Webcam data

From the registered student, the webcam data is taken. Both the front camera and back camera from the webcam video are considered. The video input V_D is given as,

$$V_D = \{v_1, v_2, \ldots\ldots, v_n\} \tag{2}$$

Here, v_n is the n-number of video data.

3.2.1 Pre-processing

The input videos V_D are converted into frames $\Re_f$. Subsequently, to obtain unique frames, motion is estimated between the frames, and noises are removed for enhancing the classification performance.

(a) Motion estimation

This research technique utilizes the KRCDS algorithm for the motion estimation. The Diamond Search (DS) method is centered on the magnitude of a motion vector. But it considers only the difference between the frames and not the strength between the frame-blocks. Thus, the KRC between the blocks of the frame is calculated by this research approach. The location with a minimum difference of blocks (ℓ) is taken. Here, KRC is also considered, which is estimated using equation (3),

$$\varpi_{sg} = \frac{C_n(\ell) - D_s(\ell)}{C_n(\ell) + D_s(\ell)} \quad (3)$$

Here, $C_n(\ell)$ and $D_s(\ell)$ are the concordant and discordant pairs of blocks. The selected frame is signified as Z_s.

(b) Noise removal

This research approach utilizes the WF for noise removal, which is highly appropriate for both image and signal noise removal.

$$R_d = \int Z_s(\lambda)\delta(\kappa - \lambda)d\lambda \quad (4)$$

Here, R_d is the noise-removed image, $Z_s(\lambda)$ is the impulse response of the selected frame Z_s, δ is the random process, and κ is the time interval.

3.2.2 Object segmentation

In this, the objects are segmented from R_d by utilizing DBS-KM. For large amounts of data, the conventional KM is highly suitable with better efficiency. However, the research has an outlier issue in the final clustering process. Thus, this research technique utilizes the DBS function for the centroid calculation, which is expressed in equation (5),

$$\psi_{cen} = \frac{R_d + R_{d+1}}{D_f(R_d)} \quad (5)$$

Where, ψ_{cen} is the calculated centroid value and $D_f(R_d)$ is the difference between the noise-removed images. Next, the difference Ψ_{dis} is estimated between the centroid and R_d.

$$\Psi_{dis} = \sqrt{\sum(\psi_{cen} - R_d)^2} \quad (6)$$

The final cluster set G_d is given in equation (7),

$$G_d = \{g_1, g_2, g_3, \ldots\ldots, g_n\} \quad (7)$$

Where, g_n signifies the n-number of clusters.

3.2.3 Face Point Detection

Here, the face clusters of G_d are taken as the input. The facial points are detected from the face. This research approach utilizes the VJ algorithm for facial point detection. The VJ has haar like feature selection (H_{fs}), integral image creation (I_{ic}), running AdaBoost (R_{ab}), and creating classifier cascades (C_{cc}) processes.

$$F_{pt} = \{H_{fs}, I_{ic}, R_{ab}, C_{cc}\} \quad (8)$$

Here, F_{pt} depicts the facial points.

3.2.4 Feature extraction

In this section, features like Scale-Invariant Feature Transform (SIFT), Gray Level Co-occurrence Matrix (GLCM), et cetera are extracted from F_{pt} and other objects G_d. The extracted feature set $V_D(L_f)$ of video is expressed in equation (9),

$$V_D(L_f) = \{V_D(l_1), V_D(l_2), \ldots\ldots, V_D(l_n)\} \quad (9)$$

Where, $V_D(l_n)$ is the n-number of extracted features from webcam video.

3.3 Audio data

Here, from the video, the audio is extracted; the audio data is A_D. Next, to increase the audio quality, noise is removed by using WF, which is described in section 3.2.1 (b). Features like Mel-frequency Cepstral Coefficients (MFCCs), zero-crossing rate, and so on are extracted from the pre-processed audio. The extracted feature set is indicated as $A_D(L_f)$.

3.4 Screen recorder

Moreover, the screen recorder data S_D is obtained from the student. Initially, the screen recorder is converted into the frame from which the motion estimation is performed as explained in section 3.2.1 (a). The derived motion value is considered as the feature, and it is depicted as $S_D(L_f)$.

3.5 App setting screenshot

Another input data is the app setting screenshot AS_D. App connection information about the device is extracted from the AS_D, and it is signified as $AS_D(L_f)$

3.6 Feature selection

Here, for optimal feature selection, the extracted features are inputted to the SWKO. $\Im_s$ signifies the combination of the extracted features. KO has an intelligent process in prey searching strategy. However, it first chooses the prey randomly; next, the position updating procedure is performed. It increases the iteration count while searching for prey. Thus, this research approach utilizes the Schaeffer Weight function. Here, the kookaburra is regarded as the $\Im_s$. The population's position is derived in equations (10) and (11),

$$w_{u,v} = \varpi_{lo} + \rho.(\varpi_{up} - \varpi_{lo}) \quad (10)$$

$$W = \begin{bmatrix} W_1 \\ \vdots \\ W_u \\ \vdots \\ W_N \end{bmatrix}_{N\times m} = \begin{bmatrix} w_{1,1} & \cdots & w_{1,v} & \cdots & w_{1,m} \\ \vdots & \ddots & \vdots & \therefore & \vdots \\ w_{u,1} & \cdots & u_{u,v} & \cdots & w_{u,m} \\ \vdots & \ddots & \vdots & \therefore & \vdots \\ w_{N,1} & \cdots & w_{N,v} & \cdots & w_{N,m} \end{bmatrix}_{N\times m} \quad (11)$$

Here, W depicts the population matrix of the kookaburra, N signifies the number of the kookaburra, m specifies the number of problem variables, ρ denotes random numbers, and ϖ_{lo} and ϖ_{up} are the lower bound and upper bound, respectively. Then, the fitness value is derived. Here, maximum classification accuracy $Max(A_{ccy})$ is considered as the fitness function J_{fit}. Next, the position is updated utilizing equations (12) to (15),

$$w_{u,v}^{k1} = w_{u,v} + \rho.(\wp_{u,v} - rand.w_{u,v})$$
(12)

$$\wp_{u,v} = (AN_l \times GA_l)/\hbar \quad (13)$$

$$W_u = \begin{cases} W_u^{k1}, & (J_{fit})_u^{k1} < J_{fit}(u) \\ W_u, & else \end{cases} \quad (14)$$

Here, $w_{u,v}^{k1}$ symbolizes the newly generated population position, $rand$ is the random number, $\wp_{u,v}$ indicates the initially selected prey, AN_l is the length of the prey, GA_l is the chest girth of the prey, and $\hbar$ is the constant value. The position updating during the exploitation stage is derived in equations (15) and (16),

$$w_{u,v}^{k2} = w_{u,v} + (1-2\rho).\frac{(\varpi_{up} - \varpi_{lo})}{\xi}$$
(15)

$$W_u = \begin{cases} W_u^{k2}, & (J_{fit})_u^{k2} < J_{fit}(u) \\ W_u, & else \end{cases}$$
(16)

Where, $w_{u,v}^{k2}$ is the new suggested position, and ξ is the iteration count. The pseudocode of the SWKO is expressed below:

Psuedo code for SWKO

Input: $\Im_s$

Output: $B_s = \{b_1, b_2, b_3, \ldots\ldots. b_n\}$

Begin

Initialize population, ξ and maximum iteration $\xi_{\max}$

Evaluate J_{fit}

Set $\xi = 1$

While $(\xi \le \xi_{\max})$ **do**

 Updation

 If $(J_{fit})_u^{k2} < J_{fit}(u)$**{**

 $w_{u,v}^{k2} = w_{u,v} + (1-2\rho).\frac{(\varpi_{up} - \varpi_{lo})}{\xi}$

 } else {

 $w_{u,v}^{k2} = W_u$

 } end if

 Evaluate J_{fit}

 Set $\xi = \xi + 1$

End while

Return B_s

End

The selected features B_s are given as,

$$B_s = \{b_1, b_2, b_3, \ldots\ldots. b_n\} \quad (17)$$

Where, b_n is the n-number of selected features.

3.7 Classification

Here, to predict the misbehavior of the student, the selected features and all the pre-processed data are given as input to the GMP-DCNN classifier. The combination of selected features and all the preprocessed data is signified as ϕ_t The information is automatically extracted from the image using DCNN. However, it only considers the maximum value in the pooling process. It might produce poor accuracy. Thus, this research approach estimates the GMP layer. Figure 2 exhibits the structure of GMP-DCNN,

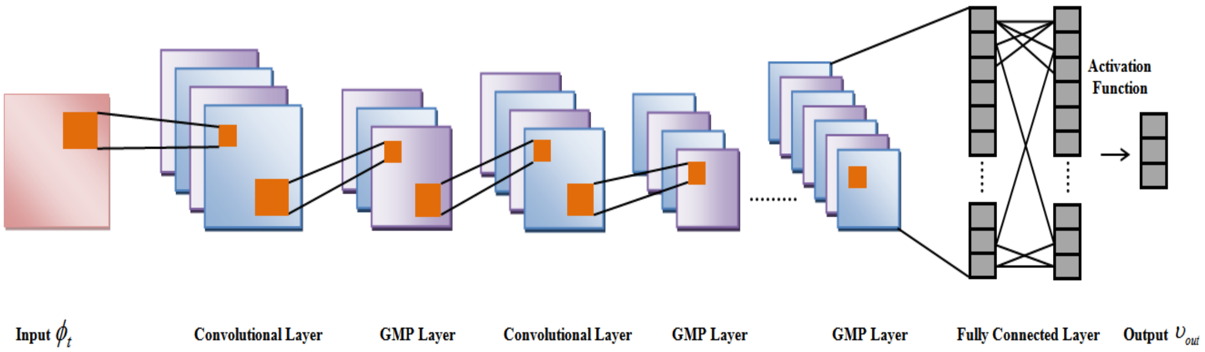


Figure 2: Structure of GMP-DCNN approach

The convolution feature map ε_{con} is given as,

$$\varepsilon_{con} = \chi.\phi_t + \tau \tag{18}$$

Here, χ and τ signify the weight factor and bias term. Next, the ε_{con} is given to the GMP layer ε_{gmp}, which is derived in equation (19),

$$\varepsilon_{gmp} = \left(\prod \varepsilon_{con}\right)^{\frac{1}{n}} \tag{19}$$

Next, to connect the layer output, the fully connected layer is utilized. Subsequently, the softmax activation υ_{out} is utilized in the output layer.

$$\upsilon_{out} = \frac{e^{\varepsilon_{gmp}}}{\sum e^{\varepsilon_{gmp}}}$$

(20)

After that, based on the target TGT, the loss function γ_{loss} is calculated, and it is expressed in equation (21),

$$\gamma_{loss} = -\sum TGT \log \upsilon_{out} \tag{21}$$

The pseudocode of the GMP-DCNN is displayed below,

Psuedo code for GMP-DCNN

Input: ϕ_t

Output: υ_{out}

Begin

Initialize ε_{con}, ε_{gmp}, and γ_{loss}

For each ϕ_t **do**

Perform ε_{con}

Perform $\varepsilon_{gmp} = \left(\prod \varepsilon_{con}\right)^{\frac{1}{n}}$

Calculate υ_{out}

If $\gamma_{loss} == \min$ **{**

Terminate

} else {

Increase iteration

} end if

End for

Return υ_{out}

End

If the behavior is abnormal, then an alert message is forwarded to the invigilator for taking further action.

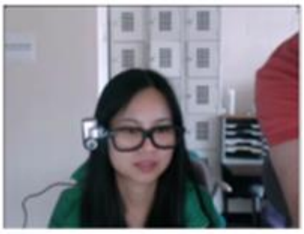
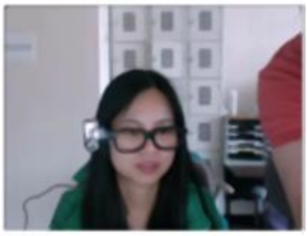
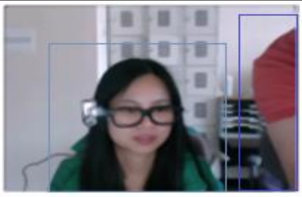
4. Result and Discussion

This section examines the performance of the proposed GMP-DCNN-based proctoring system in the exam. In the working platform of Python, the proposed methodology is implemented.

4.1 Dataset Description

By using the OEP dataset collected from publicly available resources, the research methodology is examined for performance analysis. Moreover, the dataset link is given under the reference section. For training and testing purposes, 80% and 20% of the data are utilized. Table 1 presents the sample output images.

Table 1: Sample input and output images

.No	Input	Filtering	Segmentation
1			
2			
3			

4.2 Performance analysis

4.2.1 Performance analysis of object segmentation

Here, the performance of the DBS-KM is examined with the prevailing K-Means (KM), Farthest First Clustering (FFC), Possibilistic Fuzzy C- Means (PFCM), and Fuzzy C-Means (FCM) techniques.

4.2.2 Performance analysis of feature selection

This section analyzes the proposed SWKO algorithm with the prevailing Kookaburra Optimization (KO), Coati Optimization (CO), Green Anaconda Optimization (GAO), and Egret Swarm Optimization (ESO) algorithms. The graphical plot of fitness vs iteration analysis for the proposed and existing optimization approaches is depicted in Figure 4. Here, the proposed SWKO attains fitness of 92.5% when the iteration count is 20. Nevertheless, since the convergence problem of KO is solved by the Schaeffer Weight function, the prevailing technique attains lower fitness.

In Figure 3, the graphical plot of the clustering time analysis is represented. For clustering the objects, the DBS-KM takes 30921ms time. However, the prevailing algorithm's average clustering time is 36445ms, which is higher than the proposed model. This is because the proposed technique includes the DBS function for solving the outlier issue.

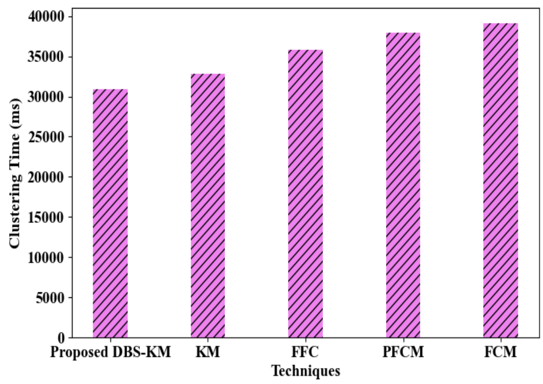


Figure 3: Graphical plot of clustering time analysis

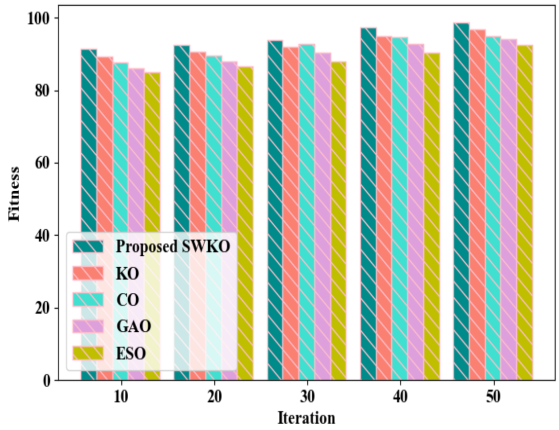


Figure 4: Fitness Vs Iteration Analysis

4.2.3 Performance analysis of classification

Here, based on accuracy, precision, recall, F-measure, specificity, sensitivity, training time, Mean Square Error (MSE), Root MSE (RMSE), Mean Absolute Error (MAE), Mathew's Correlation Coefficient (MCC), False Positive Rate (FPR), and False Negative Rate, the proposed GMP-DCNN is examined with the prevailing Deep Convolutional Neural Network (DCNN), Recurrent Neural Network (RNN), Deep Learning Neural Network (DLNN), and Artificial Neural Network (ANN).

In Table 2, the error analysis of the GMP-DCNN with the existing classifiers is exhibited. Here, the GMP-DCNN has an MSE of 2.1, RMSE of 4.2, and MAE of 3.654. However, the prevailing classifiers have higher errors than the proposed system.

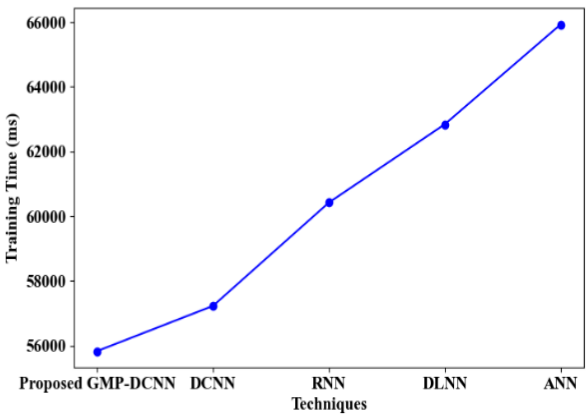
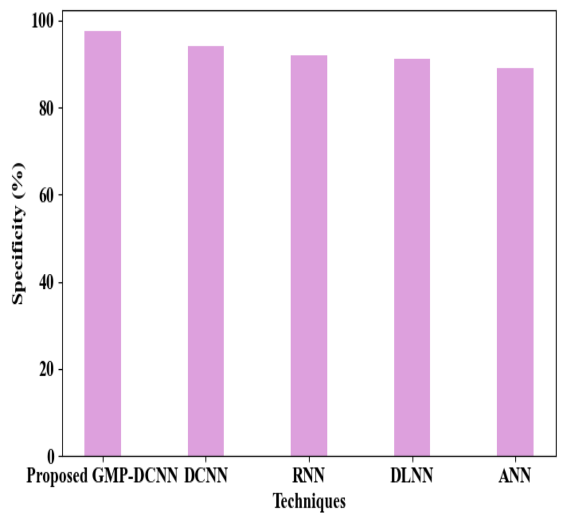


Figure 5: Graphical representation of training time analysis

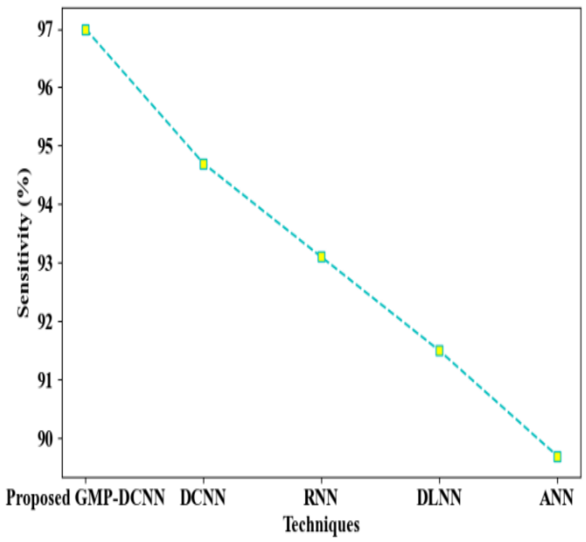
Figure 5 displays the graphical plot of the training time analysis. Here, to classify the student behavior, the GMP-DCNN takes 55834ms time. Because of the GMP function, GMP-DCNN has a lower training time than the existing research works.

Table 2: Performance analysis with respect to error metrics

Metrics	Proposed GMP-DCNN	DCNN	RNN	DLNN	ANN
MSE	2.1	5.6	7.9	10.5	13.6
RMSE	4.2	9.3	13.8	16.1	20.8
MAE	3.654	6.146	8.764	12.934	15.217



(a)



(b)

Figure 6: Pictorial plot for classifier analysis based on (a) specificity and (b) sensitivity

The pictorial plot for the classifier analysis with respect to (a) specificity and (b) sensitivity metrics is presented in Figure 6. Here, the sensitivity and specificity values of GMP-DCNN are 97 and 97.5, correspondingly. However, the sensitivity and specificity values of existing DCNN are 94.7 and 94, RNN are 93.1 and 92, DLNN are 91.5 and 91.1, and ANN are 89.7 and 89, correspondingly.

Table 3: Performance analysis of classifiers based on statistical metrics

Metrics/Methods	**Proposed GMP-DCNN**	**DCNN**	**RNN**	**DLNN**	**ANN**
Accuracy	98.8	95	93.2	91.6	89
precision	97.5	94	92	91.1	89
recall	97	94.7	93.1	91.5	89.7
F-measure	97.2	94.3	92.6	91.3	89.3
MCC	97	94	92.5	91	90
FNR	0.0361	0.0492	0.0674	0.0892	0.1023
FPR	0.0573	0.0798	0.0933	0.1298	0.1473

In Table 3, the performance analysis of the classifier based on the statistical measures is displayed. Here, the accuracy of the GMP-DCNN is 98.8%. But the prevailing techniques have lower accuracy values. Likewise, because of the integration of GMP in the DCNN model, the proposed approach achieves better performance based on the other metrics also.

Table 4: Comparative analysis of the proposed and state-of-art works

Author Name	Methods	Accuracy	Precision	Recall
(Potluri et al., 2023)	Attentive-net	0.87	0.81	0.96
(Liu, 2023)	2-Longitudinal-Stream CNN	89.1	-	-
(Saba et al., 2021)	Fine K-Nearest Neighbour (FKNN)	93.88	75.73	-
(Yaqub et al., 2021)	Machine learning approach	74.06	90.46	51.76
(Hussein et al., 2022)	Speeded Up Robust Feature (SURF) based classifier	91%	-	-
Proposed model	GMP-DCNN	98.8	97.5	97

4.3 Comparative analysis

The comparative analysis of the proposed approach with the traditional works regarding accuracy, precision, and recall metrics is displayed in Table 4. Here, the proposed model has a higher accuracy (98.8%) as it considers the segmentation process with a more input factor-centric

GMP-DCNN classification process. The prevailing FKNN technique attains 93.88% accuracy, which is higher than the other prevailing techniques. However, it also achieves the lowest performance when compared to the proposed model.

5. Conclusion

In this work, a GMP-DCNN-based online exam proctoring system is proposed. By using the publicly available OEP dataset, the performance of the proposed approach is analyzed. The experimental analysis examines the performance of the proposed model with the existing methodologies and state-of-art works. The proposed GMP-DCNN achieves 98.8% accuracy with less training time (55834ms). When compared to the prevailing models, the proposed technique attains better performance based on the other metrics. But, the mask and scarf factors in online exam monitoring were not covered in this research work.

Future scope: To improve the OEP system, the proposed work will include all the factors with advanced techniques in the future.

References

Dataset: http://cvlab.cse.msu.edu/project-OEP.html

[1] Fidas, C. A., Belk, M., Constantinides, A., Portugal, D., Martins, P., Pietron, A. M., Pitsillides, A., &Avouris, N. (2023). Ensuring Academic Integrity and Trust in Online Learning Environments: A Longitudinal Study of an AI Centered Proctoring System in Tertiary Educational Institutions. *Education Sciences*, *13*(6), 1–30. https://doi.org/10.3390/educsci13060566

[2] Ganidisastra, A. H. S., & Bandung, Y. (2021). An Incremental Training on Deep Learning Face Recognition for M-Learning Online Exam Proctoring. *IEEE Asia Pacific Conference on Wireless and Mobile*, 213–219. https://doi.org/10.1109/APWiMob51111.2021.9435232

[3] Garg, K., Verma, K., Patidar, K., Tejra, N., &Petidar, K. (2020). Convolutional Neural Network based Virtual Exam Controller. *Proceedings of the International Conference on Intelligent Computing and Control Systems*, 895–899. https://doi.org/10.1109/ICICCS48265.2020.9120966

[4] Hussein, F., Al-Ahmad, A., El-Salhi, S., Alshdaifat, E., & Al-Hami, M. (2022). Advances in Contextual Action Recognition: Automatic Cheating Detection Using Machine Learning Techniques. *Data*, *7*(9), 1–13. https://doi.org/10.3390/data7090122

[5] Kaddoura, S., &Gumaei, A. (2022). Towards effective and efficient online exam systems using deep learning-based cheating detection approach. *Intelligent Systems with Applications*, *16*, 200153. https://doi.org/10.1016/j.iswa.2022.200153

[6] Kaddoura, S., Popescu, D. E., & Hemanth, J. D. (2022). A systematic review on machine learning models for online learning and examination systems. *PeerJ Computer Science*, *8*, 1–32. https://doi.org/10.7717/PEERJ-CS.986

[7] Labayen, M., Vea, R., Florez, J., Aginako, N., & Sierra, B. (2021). Online Student Authentication and Proctoring System Based on Multimodal Biometrics Technology. *IEEE Access*, *9*, 72398–72411. https://doi.org/10.1109/ACCESS.2021.3079375

[8] Li, H., Xu, M. X., Wang, Y., Wei, H., & Qu, H. (2021). A visual analytics approach to facilitate the proctoring of online exams. *Conference on Human Factors in Computing Systems*, 1–17. https://doi.org/10.1145/3411764.3445294

[9]] Liu, T. (2023). AI proctoring for offline examinations with 2-Longitudinal-Stream Convolutional Neural Networks. *Computers and Education: Artificial Intelligence*, *4*(1), 1–11. https://doi.org/10.1016/j.caeai.2022.100115

[10] Motwani, S., Nagpal, C., Motwani, M., Nagdev, N., &Yeole, A. (2021). AI-Based Proctoring System for Online Tests. In Proceedings of the 4th International Conference on Advances in Science & Technology, 1–6. https://doi.org/10.2139/ssrn.3866446

[11] Muzaffar, A. W., Tahir, M., Anwar, M. W., Chaudry, Q., Mir, S. R., & Rasheed, Y. (2021). A systematic review of online exams solutions in e-learning: Techniques, tools, and global adoption. IEEE Access, 9, 32689–32712. https://doi.org/10.1109/ACCESS.2021.3060192

[12] Nigam, A., Pasricha, R., Singh, T., &Churi, P. (2021). A Systematic Review on AI-based Proctoring Systems: Past, Present and Future. Education and Information Technologies, 26(5), 6421–6445. https://doi.org/10.1007/s10639-021-10597-x

[13] Nurpeisova, A., Shaushenova, A., Mutalova, Z., Ongarbayeva, M., Niyazbekova, S., Bekenova, A., Zhumaliyeva, L., & Zhumasseitova, S. (2023). Research on the Development of a Proctoring System for Conducting Online Exams in Kazakhstan. *Computation*, *11*(6), 1–21. https://doi.org/10.3390/computation11060120

[14] Pandey, A. K., Kumar, S., Rajendran,B.,&Bindhumadhava, B. S. (2020). E-parakh: Unsupervised online examination system. *IEEE Region 10 Annual International Conference*, 667–671. https://doi.org/10.1109/TENCON50793.2020.9293792

[15] Potluri, T., Venkatramaphanikumar, S., & Kishore, K. V. K. (2023). An automated online proctoring system using attentive-net to assess student mischievous behavior. Multimedia Tools and Applications, 82(20), 30375–30404. https://doi.org/10.1007/s11042-023-14604-w

[16] Saba, T., Rehman, A., Jamail, N. S. M., Marie-Sainte, S. L., Raza, M., & Sharif, M. (2021). Categorizing the Students' Activities for Automated Exam Proctoring Using Proposed Deep L2-GraftNet CNN Network and ASO Based Feature Selection Approach. *IEEE Access*, *9*, 47639–47656. https://doi.org/10.1109/ACCESS.2021.3068223

[17] Tiong, L. C. O., & Lee, H. J. J. (2021). E-cheating Prevention Measures: Detection of Cheating at Online Examinations Using Deep Learning Approach -- A Case Study. *ArXiv*, 1–9. http://arxiv.org/abs/2101.09841

[18] Turani, A. A., Alkhateeb, J. H., &Alsewari, A. R. A. (2020). Students Online Exam Proctoring: A Case Study Using 360 Degree Security Cameras. *International Conference on Emerging Technology in Computing, Communication and Electronics*, 1–5. https://doi.org/10.1109/ETCCE51779.2020.9350872

[19] Yaqub, W., Mohanty, M., & Suleiman, B. (2021). Image-Hashing-Based Anomaly Detection for Privacy-Preserving Online Proctoring. *ArXiv*, 1–7. http://arxiv.org/abs/2107.09373

[20] Yulita, I. N., Hariz, F. A., Suryana, I., &Prabuwono, A. S. (2023). Educational Innovation Faced with COVID-19: Deep Learning for Online Exam Cheating Detection. Education Sciences, 13(2), 1–15. https://doi.org/10.3390/educsci1302