

# Smart Quiz Through Computer Vision

Ms. Siddhi Khemnar, Riya Nemade, Aishwarya Pawar, Sayali Shriwastav

Students, Computer Engineering

Asst. Prof. B. S. Gholap

Department of Computer Engineering

MVP'S Karmaveer Adv. Baburao Ganpatrao Thakare College Of Engineering, Nashik.

\*\*\*

**Abstract** - These days, computer-based multiple-choice exams are utilized all around within the enlistment, examination, and other forms. Making the setup for a computer based test is exceptionally costly since it requires a parcel of gear and computer program. We are working on a venture that will utilize computer vision to diminish the sum of equipment, like a console and mouse, required for computer-based exams. The recommended arrangement would utilize computer vision to display the address on screen after saving it to a csv record to begin with. Second, we are going for hand

developments within the discuss that can be utilized to reply to the displayed questions employing a webcam and computer vision. The programmed reviewing framework in it'll at that point show the review.

This approach made it easy to reply to questions and indeed made the assessment method clear. To confirm that no negligence happened amid the exam, this innovation guarantees more security by checking the number of individuals who are before the webcam, in the event that more than one individual the exam naturally closes. Critically, since of exceptionally small innovation is required, the exam won't be ended within the center due to a equipment disappointment or breakdown. Moreover it'll incorporate versatile multiple-choice address testing framework that conveys MCQ appraisals to clients over the web and adjusts the questions based on the user's execution and capacities. For case, in the event that the understudy answers a address accurately, the stage may display a more challenging address another, whereas an erroneous reply may result in an less demanding address which can be great at measuring a run of student ability.

**Key Words:** Computer Vision, Difficulty Level, Face Detection, Hand Gesture, Smart Quiz

## 1.INTRODUCTION

Computer use is now widespread and has a big impact on how we live our lives. Computer applications require human-computer interaction. We frequently use hand gestures, which are simply the movement, position, or posture of the hands, to communicate nonverbally. To categorize hand motions in videos and images for various

purposes, several research are being conducted. The actual movement of the human hand is what creates gestures. Hand gesture recognition is a natural and intelligent method of Human Computer Interaction (HCI). The term "computer-based testing," or "CBT," refers to the administration of tests using computers as opposed to the traditional pen-and-paper approach. The web or a computer-aided office can be utilized to regulate such a test online. Computer technology is used in computer-based testing (CBT), which means that applicants use computers to respond to questions displayed on monitors.

The test-taker enters the response using a keyboard or mouse. The "client computer" the test-taker is using is referred to as such. Providers of online examination content should focus on creating useful assessment questions and giving students exam feedback. Students or trainees might use a computer to complete exercises or tests with the help of online examination systems. The issue with the examiner workload in the manual system has been resolved. This will verify the right response, save the examiner's time, and ensure that the exam is administered efficiently. In contrast to the existing approach, reviewing the answer sheets after completing the test wastes the examiner's time. A web-based online assessment system's major goal is to assess the student using a fully automated method thoroughly and efficiently that not only saves a ton of time but also provides quick and accurate results.

## 2. LITERATURE REVIEW

Using OpenCV and Python, Surya Narayan Sharma and Dr. A Rengarajan developed Hand Gesture Recognition in 2021[1]. Using computer vision with Python and Numpy, the aim of this technique of implementation is to develop a comprehensive system that can recognize, locate, and interpret hand motions. Only plain backgrounds can be used for this detection, but accuracy needs to be increased. Implementation and accuracy issues will result from even a little problem. In this paper, Hand gesture recognition applications range from medical rehabilitation to consumer electronics control (e.g. mobile phone) [2]. In order to distinguish hand gestures, various kinds of sensing techniques are utilized to obtain signals for pattern recognition. This paper contains the misplaced

person recognition and detection system advanced technology that can identify the person while the process of detection and show the id of the person which are stored in the database [3]. This paper proposed a RGB-D based recognition method of static and dynamic hand gesture for natural human-computer interaction application with virtual environment[4]. Multiple-choice tests administered using a computer are almost always employed in today's hiring, testing, and other procedures. These software programmers were made with a variety of programming languages. In the same way that Er. Pranay Meshram, Rajat Tarale, Vaishnavi Upare, and Shruti Wakodikar developed the idea for an online MCQ test PHP and SQL are used in the web application development of the online examination system [5]. The administration can see, delete, and update student information with efficiency. However, the main drawbacks of this type of application are that setup necessitates more expensive hardware systems, and if there are any interruptions, it is impossible to finish exams on time. Additionally, as there is no automatic evaluation mechanism in place for them, it takes longer to review the test and publish the results. Face detection and identification using opencv and python had been implemented by Dhawle, Tejashree, Urvashi Ukey, and Rakshandha Choudante. The major objective of this study is to develop or construct a system that will use the computer's camera to recognize and identify a person's or an individual's face using the Python programming language and the OpenCV tool known as the Open Face[6]. Dlib and face recognition are the system's two most important libraries. Face recognition technology has several advantages, including quicker processing, identity automation, privacy invasion, massive data storage, the best results, increased security, real-time face recognition of employees and students at corporate offices, smartphone unlocking, and many more. There are some drawbacks to this system, including its high costs or funding requirements, the requirement for high-quality cameras, poor image quality that may limit its efficacy, the significance of image size because it becomes difficult to recognize faces in small images, face angles that may decrease the accuracy of face recognition, and the requirement for enormous amounts of storage.

### 3. PROPOSED METHOD AND METHODOLOGY

Our objective is to create a software system that uses computer vision and a webcam to answer questions on multiple-choice assessments that are computer-based utilizing hand gestures. The candidate's score will also be computed by this system and displayed on the screen. There won't be any hardware interruptions, unlike with older systems, due to the system's minimal hardware configuration, which also makes implementation and maintenance expenses affordable. We employ hand gestures in this project to provide answers to queries, and the cvzone module in Python is used to track the hand movements. This computer vision package facilitates the

execution of AI and image processing operations. It primarily makes use of the OpenCV and Mediapipe libraries[7]. Cvzone was employed in our project. Hand Tracking modules for utilizing the media pipe algorithm. It offers a hand detector feature that can find hands and palms. This library makes use of Media Pipe, a Google cross platform library that uses machine learning methods to extract 21 3D hand landmarks from a single frame. Using Media Pipe for hand tracking takes two steps:

- Palm detection: Media Pipe uses the entire image as input and outputs a cropped image of the hand.
- Identification of hand landmarks: MediaPipe identifies the 21 hand landmarks on the hand's cropped image.

MediaPipe Hands is a high-quality hand and finger following framework. It extrapolates 21 3D hand landmarks from a single frame using machine learning (ML)[8]. Unlike other cutting-edge systems, which mostly rely on potent desktop settings for inference, our method provides realtime performance on a cell phone and even scales to numerous hands. We believe that opening up access to this hand perception functionality to a wider research and development community will spark the development of creative use cases and new lines of inquiry. A model for detecting palms that makes use of the complete image and generates an orientated hand bounding box is part of the MediaPipe Hands ML pipeline. A hand landmark model that produces extremely precise 3D hand key points while operating on the palm detector-cropped image region. This method is comparable to the one used in our MediaPipe Face Mesh solution, which combines a face detector with a face landmark model. Giving the hand landmark model a properly cropped hand image greatly reduces the requirement for data augmentation (such as rotations, translations, and scaling) and allows the network to concentrate its resources on accurate coordinate prediction[8]. Palm discovery is as it were performed to delocalize the hand when the point of interest demonstrate is incapable to do so. The crops in our pipeline can also be produced using the hand landmarks found in the previous frame. A dedicated hand renderer subgraph is utilized for rendering, and a hand landmark tracking subgraph from the hand landmark module is used in the pipeline's implementation as a MediaPipe graph. The hand landmark tracking subgraph uses both a hand landmark subgraph from the same module and a palm detection subgraph from the palm detection module internally

#### 3.1 PALM DETECTION MODEL

In a manner similar to the face detection model in MediaPipe Face Mesh, we created a single-shot detector model intended for mobile real-time uses to detect initial hand locations. The issue of detecting hands is incredibly challenging because both our lite model and complete

model must be able to recognize occluded and selfoccluded hands while operating over a wide scale span (20x) relative to the image frame. It is somewhat challenging to identify hands correctly from their visual features alone, in contrast to faces, which exhibit high contrast patterns, such as in the region of the mouth and eyes. Accurate hand localization is instead made possible by adding additional context, such as aspects of the arm, torso, or person. Our approach employs many techniques to address the aforementioned problems. First, we train a palm detector rather than a hand detector because it is far easier to estimate the bounding boxes of stiff objects like fists and palms than it is to detect hands with articulated fingers. The non-maximum suppression technique also performs well in two-hand self-occlusion scenarios like handshakes since palms are smaller objects. The number of anchors can be reduced by a factor of 3-5 by modelling palms with square bounding boxes, or anchors in machine learning terminology. In order to provide greater scene context awareness, even for small objects, an encoder-decoder feature extractor is used.

### 3.2 HAND LANDMARK MODEL

After identifying the palm over the full image, our next step is to achieve exact key point localization of 21 3D hand-knuckle coordinates inside the identified hand regions using regression, or direct coordinate prediction. Self-occlusions and partially visible hands have no effect on the model's ability to obtain a trustworthy internal hand posture representation. To gather ground truth information, we manually added 21 3D coordinates to about 30K actual photographs, as seen below (we take Zvalue from image depth map, if it exists per corresponding coordinate). In addition, to better cover the range of potential hand positions and provide extra supervision on the nature of hand geometry, we render a high-quality synthetic hand model over a variety of backgrounds and map it to the related 3D coordinates.

- The fig-1 below depicts the 21 hand points that MediaPipe recognizes[8]:

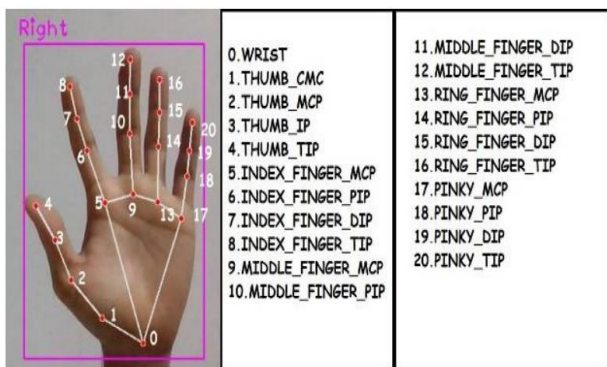


Fig -1: 21 Hand Points

### 3.3 Dlib

Dlib was used in this project to count the faces in the webcam. The C++ programming language was used to create the general-purpose, cross-platform software library known as Dlib [9]. Its design is significantly influenced by concepts from component-based software engineering and design by contracts. It is therefore first and primarily a collection of separate software parts. The software is opensource and distributed under the Boost Software License. Dlib has developed to now contain a large number of tools since development first started in 2002. It now has software elements for dealing with networking, threads, graphical user interfaces, data structures, linear algebra, machine learning, image processing, data mining, XML and text parsing, numerical optimization, Bayesian networks, and many more tasks. Although Dlib is primarily a C++ library, you can use some of its capabilities in Python programs..

#### Live Video Capturing :

This is the Initial step taken by the system Immediately after the candidate logged into the exam. Video capturing is done with the help of an initialized webcam using OpenCV library.

#### MCQ Details :

The questions, options and correct answer for the question are stored in a csv file. Later this data is useful for displaying during the exam and for evaluating the student's answer. And to calculate the score of the candidate.

#### Presenting the Questions :

In this step the system extracts the details from the csv file where the questions and options are already stored in it the csv file is extracted using csv library in python. Then the questions and options are displayed on the screen on live video using OpenCV and CVZONE libraries.

#### Counting number of Faces :

In this stage, the system uses the C++-written dlib package to identify the candidate's face. This library makes it easier to count the faces in front of the webcam[6] . It is useful to know how many people are present while the exam is taking place. Once the number of people has been counted, if more than one person is found, the system will debar the candidate from the exam and the process will continue until the exam is over in order to ensure that no malpractice occurred.



**Tracking and Detecting Hand :**

Next the system tracks the hand and detects the palm of the candidate through a webcam so that it can understand the gesture shown by the candidate. This detection and Tracking are done with the help of a hand tracking class in CVZONE library. This CVZONE library uses Media Pipe Algorithm which is created by google for tracking hands and detecting the palm. This Media pipe will detect 21 hand landmarks in the palm which make the system get better understanding about the gestures shown. Answering the questions is based on clicking the option button. The system identifies the clicked button using the distance between the index finger and Middle finger if the distance is less than the button in front of the hand is considered as clicked. For the exit button it considers the distance between ring and thumb fingers. The distance between the fingers is calculated by the CVZONE library as it identifies the hand landmarks according to that the tip of index finger and middle finger is given with landmark numbers as 8 and 12 respectively by the media pipe algorithm. Then the distance between these points is calculated if the distance is less than 60 then it considers the option is clicked. Similarly, the distance between the tip of the ring and thumb finger is calculated using their landmarks mentioned in media pipe.

**4. RESULTS**

In fig 3, page with Login options for admin and student displayed.



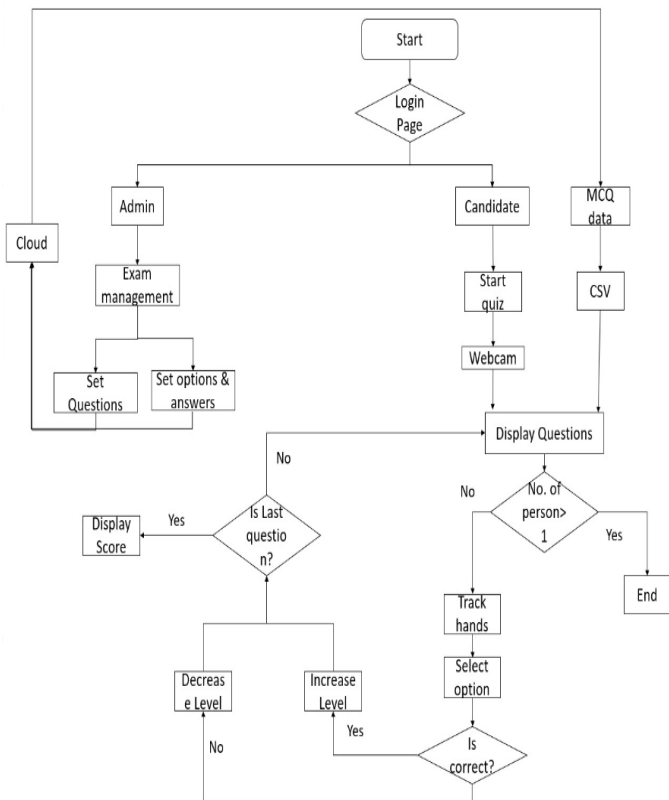
**Fig -3:** Home page

Later for admin, admin login is available as shown in fig 4.



**Fig -4:** Admin Login page

After logging in, admin will displayed page where he can add questions to dataset according to subjects as shown in fig 5.



**Fig -2:** Flow Chart

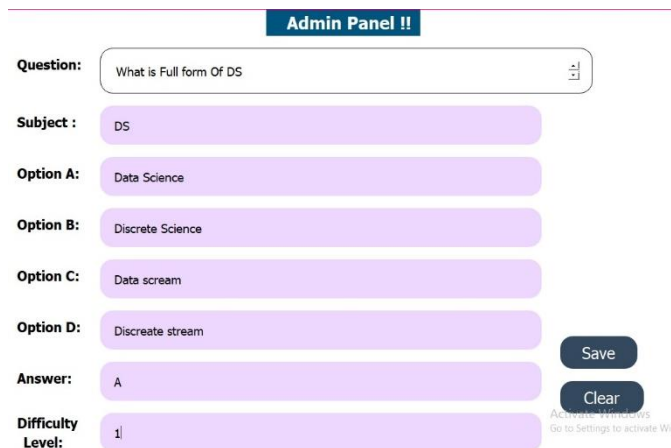


Fig -5: Admin Panel

In this way admin can add questions to dataset.

Later for student, student login is available as shown in fig.6.

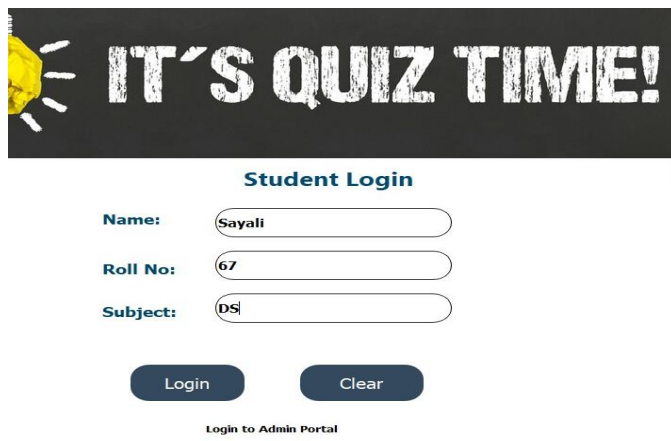


Fig -6: Student Login Page

For student, after logging in, test will be started as shown in fig 7.

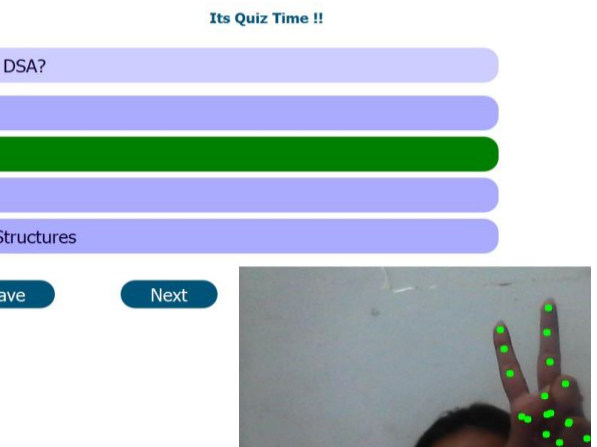
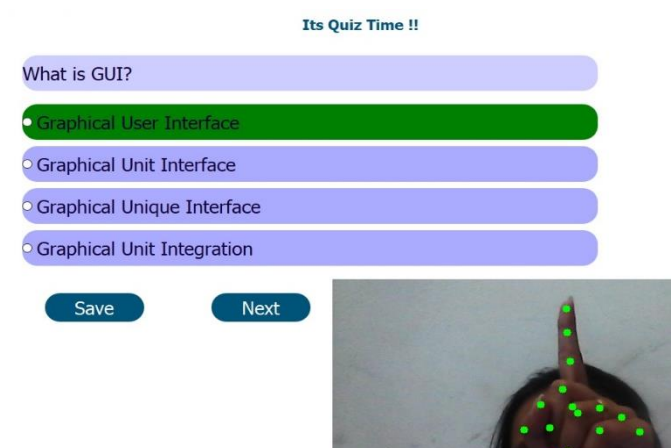


Fig -7: Answering quiz using hand gesture

In this way, user can select options using hand gestures as shown in fig 7.

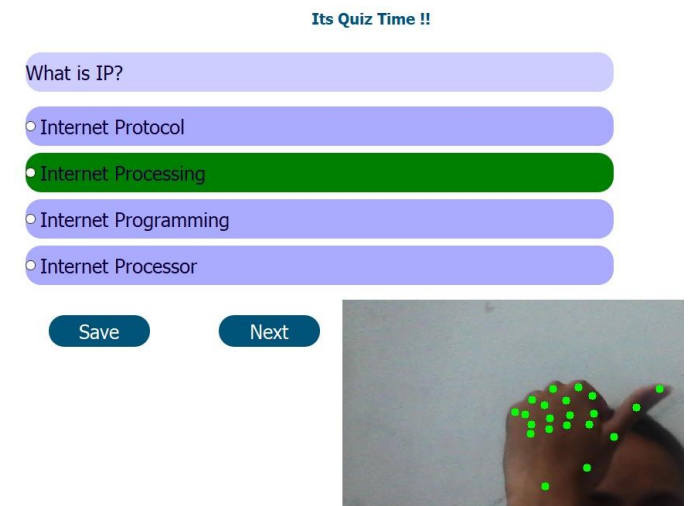


Fig -8: Saving answers using hand gesture

In this way, user can save and move on to next question. Next question will be asked using difficulty mechanism and difficulty level of current question as shown in fig 8.

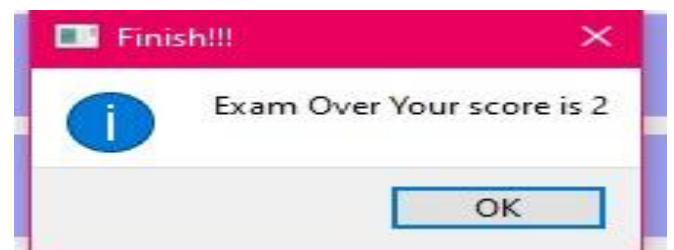


Fig -9: Displaying Score

After completing test, user can see his score on screen as shown in fig 9.

## 5. CONCLUSIONS

The integration of computer vision and webcams allows the system to recognize and process user gestures and images, providing a unique and intuitive way for users to interact with the system. The system's adaptivity is a crucial feature. It tailors the assessment questions to the user's abilities and performance, ensuring that the assessment remains engaging and challenging. By adapting questions based on user abilities, the system places the user at the center of the learning experience. This approach helps to maximize the effectiveness of the assessments. The software system's ability to evaluate user responses and adapt questions accordingly simplifies the process of assessing a user's knowledge and skills. The utilization of hand gestures and webcam interaction adds an engaging and interactive dimension to the learning experience, making it more effective and enjoyable. The system's adaptivity also provides data that can be used for continuous improvement, both in terms of the assessment content and the user experience. The software system has the potential to revolutionize how assessments are conducted by making them more adaptive, engaging, and user-centric. Its successful implementation could lead to more effective and personalized learning experiences for users across a wide range of educational contexts.

## ACKNOWLEDGEMENT

Thank all the authors who are mentioned in the reference and the respected people who helped in designing and developing of this work.

## REFERENCES

- [1] Surya Narayan Sharma — Dr. A Rengarajan "Hand Gesture Recognition using OpenCV and Python" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456- 6470, Volume-5 — Issue-2, February 2021, pp.346- 352, URL: <https://www.ijtsrd.com/papers/ijtsrd38413.pdf>
- [2] Srinivasa Rao K, et. al. International Journal of Engineering Research and Applications "Hand Gesture Recognition and Appliance Control Using Transfer Learning" ISSN: 2248-9622, Vol. 11, Issue 7, (Series-III) July 2021, pp. 37-46 <https://www.ijera.com/papers/vol11no7/Ser-3/E1107033746.pdf>
- [3] Akhilesh Allewar, Abhimanyu Sharnagate, Vaibhav Tiple, Snehal Sur, Dr. P.U.Chati , JETIR March 2022, "Misplaced Person Recognition and Detection System by using OPENCV Python" Volume 9, Issue 3 <https://www.jetir.org/papers/JETIR2203478.pdf>
- [4] Jun Xu, Hanchen wng, Jianrong zhang , "Robust hand gesture recognition based on RGBD data for natural human-Computer Interaction" IEEE.
- [5] Er. Pranay Meshram,Rajat Tarale , Vaishnavi upare , Shruti Wakodikar , Aastha Bankar , Sarika Nimaje "Web Application for Online MCQ Test" , IJARIIIE-ISSN(O)-2395-4396 Vol-6 Issue-3 2020.
- [6] Tejashree Dhawle, Urvashi Ukey, Rakshandha Choudante, "Face Detection and Recognition using OpenCV and Python", International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 10
- [7] <https://opencv.org/>
- [8] <https://pypi.org/project/mediapipe/>
- [9] <https://pypi.org/project/dlib/>
- [10] Sharma, S. N., & Rengarajan, A. (2021). "Hand gesture recognition: A comprehensive review". International Journal of Advanced Trends in Computer Science and Engineering, 10(2), 78-84.
- [11] Patel, H., Shah, S., & Patel, V. (2020). "Misplaced Person Detection and Recognition System Using Raspberry Pi". International Journal of Advanced Research in Computer Science and Software Engineering, 10(3), 47-52.
- [12] Wang, Y., Liu, S., & Zou, Y. (2019). "RGB-D based recognition method of static and dynamic hand gesture for natural human-computer interaction application with virtual environment". Multimedia Tools and Applications, 78(1), 437-456.
- [13] Meshram, P., Tarale, R., Upare, V., & Wakodikar, S. (2018). "Development of Online Multiple Choice Question PHP Framework for University Examinations". International Journal of Engineering Research & Technology (IJERT), 7(7), 430-434.
- [14] Dhawle, T., Ukey, U., & Choudante, R. (2022). "Face Detection and Identification using OpenCV and Python". International Journal of Advanced Computer Science and Applications, 13(3), 206-211.