**Literature Survey on Facial Recognition Systems for Proctoring**

**Paper 3: “Enhancing AI Proctoring System Using Various ML Models” - IEEE**

**Key Contributions of the Paper:**

* **Face Recognition and Object Detection:**
  + The paper utilizes Haar Cascade for facial feature detection and YOLO for object detection to facilitate real-time monitoring of students writing online exams. This approach allows identification of the identity and identification of unauthorized objects in the test region.
* **Frame-Based Monitoring:**
  + The algorithm deals with individual frames for the detection of exam misconduct, i.e., unauthorized devices or other individuals. This method logs noteworthy events but does not offer real-time surveillance on hidden activities.

**Limitations:**

* **Limited Facial Tracking Accuracy:**
  + Feature-based Haar Cascade is illumination-sensitive, facial pose-sensitive, and occlusion-sensitive, thus inappropriate for dynamic monitoring during online exams.
* **Frame-Based Object Detection Instead of Continuous Monitoring**
  + Since the system relies on YOLO to monitor individual frames rather than actual streams, immediate cheating behaviors (such as quickly using a cell phone) won't be captured.
* **Inefficient Communication and Processing Overhead:**
  + Lack of WebSockets or real-time server-client interaction can cause the system to rely on scheduling requests, causing lag in proctor notifications. Further, performing all YOLO inference on the server introduces processing overhead, creating scalability limited.

**Our Solution:**

* **Improved Face Tracking with OpenSeeFace & Dlib:**
  + We utilize OpenSeeFace, a cutting-edge deep-learning-driven face tracking that significantly increases the robustness to varying conditions. In contrast to Haar Cascade, OpenSeeFace correctly identifies gaze direction, head pose, and microexpressions in real-time. Dlib further adds precision to facial recognition, reducing false positives.
* **Real-Time Object and Activity Monitoring:**
  + Instead of frame-based detection, we utilize real-time YOLO stream for object detection, allowing instant detection of restricted items such as mobile phones or notes. With motion detection, this offers a complete monitoring solution.
* **Optimized Backend with WebSockets and NGINX:**
  + WebSockets for Real-Time Communication: Facilitates instant proctor notifications and seamless live monitoring without any delay from polling.
* **NGINX Load Balancing:** 
  + Efficiently distributes processing loads across multiple servers for better scalability and performance.
* **Edge Processing:** 
  + The client-side lightweight face tracking reduces the backend's computational burden.

**Paper 4: "Development of a Proctoring System for Administering Online Exams in Kazakhstan"**

**Key Contributions of the Paper:**

* **Face Recognition and Object Detection:**
  + The paper utilizes LBPH as face recognition and YOLOv3 as object detection to enable it to verify student identities, detect multiple faces, and monitor head movements while taking exams.
* **Snapshot-Based Monitoring:**
  + The system captures images periodically (e.g., 10 seconds) to capture potential exam irregularities, such as unapproved mobile phone usage or the presence of a second individual.

**Limitations:**

* **Weak Robustness:**
  + LBPH usage, although effective in controlled scenarios, is prone to illumination levels, head direction, and occlusions. This can lead to failure in face recognition in dynamic exam scenarios.
* **Discrete Monitoring Approach:**
  + Provisional in nature, periodic snapshoting has a chance to miss short-term conduct or brief seconds of malpractice, thereby diluting the whole system's credibility.
* **Missing Ongoing Facial Monitoring:**
  + The above method does not have ongoing facial monitoring of essential facial metrics such as eye movements, head direction, and gaze direction, which are all essential for identifying nascent signs of cheating.

**Our Solution:**

* **Elevated Facial Analysis with OpenSeeFace:**
  + We utilize OpenSeeFace, a cutting-edge open-source deep learning-based tracking and face recognition system for reliable performance. We utilize OpenSeeFace, a cutting-edge open-source deep learning-based face recognition and tracking system for reliable performance. In contrast to LBPH, OpenSeeFace is able to track changes in lighting, head pose, and occlusion to offer more reliable identity confirmation.
* **Continuous Monitoring:**
  + OpenSeeFace continuously monitors facial features in real-time, such as direction of gaze, eyes, and head pose. Since it is continuous analysis, the system fails to notice short or covert activity—such as the student momentarily glancing away or exhibiting illegal behaviors—will not be detected if noticed by snapshot observation at a certain interval.
* **Seamless Integration with Existing Tools:**
  + Leveraging OpenSeeFace with the help of support libraries like Dlib and OpenCV, our solution offers quick real-time processing even on CPU-based devices. This enhances not only detection efficiency but also system responsiveness in general.
* **Total Enhancement in Proctoring Reliability:**
  + With persistent and intense face scrutiny, our system provides invaluable improvement over the conventional technique. It preserves test integrity by providing real-time accurate information on any suspicious movement, thus curing for the pivotal vulnerabilities of the former system.