**Report**

Nhóm 4

Thành viên:

23631971 – Nguyễn Xuân Đỉnh

23634031 – Lê Huỳnh Tấn Đạt

23640391 – Trần Vĩnh Cơ

**1.0 Giới thiệu Tổng quan**

Hệ thống Giám sát Tập trung Cuộc họp được thiết kế để cung cấp một giải pháp phân tích hành vi tự động, nhằm đánh giá mức độ tập trung của những người tham gia trong các cuộc họp trực tuyến hoặc trực tiếp. Vấn đề cốt lõi mà hệ thống giải quyết là việc định lượng hóa các biểu hiện mất tập trung thường thấy, chẳng hạn như cúi đầu quá lâu, quay ngang, hoặc có dấu hiệu buồn ngủ, từ đó cung cấp những phản hồi khách quan. Tài liệu này được biên soạn dành cho các kỹ sư phần mềm, nhà phát triển và các bên liên quan kỹ thuật cần một sự hiểu biết sâu sắc về cấu trúc bên trong của hệ thống.

Mục tiêu của tài liệu này là trình bày một cách chi tiết và rõ ràng về kiến trúc tổng thể, luồng xử lý dữ liệu từ đầu vào đến đầu ra, và các thuật toán thị giác máy tính cốt lõi được triển khai. Việc nắm vững các khía cạnh này là nền tảng thiết yếu để hỗ trợ các hoạt động bảo trì, nâng cấp tính năng, và tích hợp hệ thống với các nền tảng khác trong tương lai.

Để đạt được mục tiêu trên, tài liệu sẽ tuần tự đi qua các thành phần kiến trúc, yêu cầu môi trường, phân tích các tham số cấu hình, mô tả chi tiết luồng dữ liệu, và cuối cùng là đi sâu vào logic toán học của các thuật toán phân tích hành vi.

**2.0 Kiến trúc Tổng thể của Hệ thống**

Để đảm bảo tính linh hoạt, dễ bảo trì và mở rộng, hệ thống được xây dựng theo kiến trúc module hóa. Mỗi module đảm nhiệm một chức năng chuyên biệt, từ việc nạp cấu hình, xử lý luồng video, phân tích hành vi, đến ghi nhận kết quả. Sự phân tách rõ ràng này cho phép các nhà phát triển có thể tinh chỉnh hoặc thay thế từng thành phần mà không gây ảnh hưởng lớn đến toàn bộ hệ thống.

Các thành phần kiến trúc chính bao gồm:

* **Module Nạp Cấu hình (settings\_loader.py):** Đây là trung tâm điều khiển tập trung, quản lý toàn bộ các tham số vận hành của hệ thống. Bằng cách tách biệt cấu hình khỏi logic mã nguồn, module này cho phép người dùng và nhà phát triển dễ dàng tinh chỉnh các ngưỡng (ví dụ: góc cúi đầu, độ nhạy phát hiện buồn ngủ) và các thiết lập khác (ví dụ: nguồn video, thời lượng phiên) mà không cần can thiệp vào mã nguồn chính.
* **Module Xử lý Chính (meeting\_monitor.py):** Được xem là trái tim của hệ thống, module này điều phối toàn bộ luồng xử lý. Nó chịu trách nhiệm đọc luồng video, gửi từng khung hình đến mô hình nhận diện, điều phối việc theo dõi đối tượng (tracking), liên kết các đối tượng với các vị trí không gian cố định, gọi các hàm phân tích hành vi, và tổng hợp dữ liệu để hiển thị và lưu trữ.
* **Module Thuật toán Phân tích (pose\_utils.py):** Đây là bộ não tính toán của hệ thống, chứa các hàm logic phức tạp để diễn giải dữ liệu thô (tọa độ keypoints từ khung xương, landmarks trên khuôn mặt) thành các trạng thái hành vi có ý nghĩa. Các thuật toán cốt lõi như tính toán góc cúi đầu và góc quay ngang đều được định nghĩa tại đây.
* **Module Ghi nhận Dữ liệu (Logic trong meeting\_monitor.py):** Thành phần này đảm nhận việc lưu trữ kết quả phân tích. Toàn bộ logic tổng hợp dữ liệu vào các cấu trúc person\_stats và meeting\_log, cũng như việc xuất ra các file meeting\_summary.csv và meeting\_detailed\_log.csv sử dụng thư viện pandas, đều được triển khai trực tiếp bên trong meeting\_monitor.py. File focus\_logger.py tồn tại trong codebase nhưng chứa logic cũ và không được sử dụng trong quy trình xuất dữ liệu chính của hệ thống.

Tiếp theo, chúng ta sẽ xem xét các yêu cầu về môi trường và thư viện cần thiết để triển khai và vận hành hệ thống một cách ổn định.

**3.0 Yêu cầu về Môi trường và Các Thư viện Phụ thuộc**

Việc thiết lập một môi trường phần mềm chính xác và nhất quán là yếu tố tiên quyết để đảm bảo hệ thống hoạt động ổn định, hiệu quả và có thể tái tạo kết quả trên các máy khác nhau. Hệ thống được xây dựng trên nền tảng Python và phụ thuộc vào một số thư viện mã nguồn mở chuyên dụng cho thị giác máy tính và xử lý dữ liệu.

Dưới đây là bảng liệt kê các thư viện Python cần thiết và vai trò của chúng trong dự án:

|  |  |
| --- | --- |
| Thư viện (Library) | Mục đích chính trong dự án (Primary Purpose in the Project) |
| ultralytics | Cung cấp framework và mô hình YOLOv8 để thực hiện nhận diện người (object detection) và ước tính tư thế (pose estimation) trong thời gian thực. |
| opencv-python | Thư viện nền tảng cho các tác vụ thị giác máy tính, bao gồm đọc và xử lý luồng video, vẽ bounding box, hiển thị thông tin và các thao tác xử lý hình ảnh khác. |
| numpy | Cung cấp các cấu trúc dữ liệu mảng hiệu suất cao và các hàm toán học cần thiết cho việc tính toán vector, khoảng cách và các phép biến đổi trên tọa độ keypoints. |
| mediapipe | Được sử dụng để phân tích lưới khuôn mặt (face mesh) một cách chi tiết, trích xuất các điểm mốc (landmarks) quanh mắt để tính toán Tỷ lệ Khung mắt (EAR) cho thuật toán phát hiện buồn ngủ. |
| supervision | Cung cấp các công cụ tiện ích cao cấp cho thị giác máy tính, cụ thể là thuật toán theo dõi đối tượng ByteTrack (sv.ByteTrack) để duy trì định danh của mỗi người qua các khung hình. |
| pytesseract | Thư viện nhận dạng ký tự quang học (OCR), được dùng để tự động đọc và trích xuất tên của người tham gia từ vùng ảnh bên dưới bounding box của họ. |
| pandas | Thư viện mạnh mẽ cho thao tác và phân tích dữ liệu, được sử dụng để cấu trúc và lưu trữ dữ liệu tổng kết và nhật ký chi tiết ra các file CSV. |
| playsound | Một thư viện đơn giản để phát các file âm thanh, được dùng để kích hoạt âm thanh cảnh báo khi phát hiện trạng thái buồn ngủ. |
| unidecode | Chuyển đổi chuỗi văn bản có dấu (ví dụ: tiếng Việt) thành dạng không dấu, giúp chuẩn hóa tên người tham gia được trích xuất từ OCR. |
| Thư viện phụ trợ/gián tiếp | Các thư viện như mss, lap, scikit-learn có trong requirements.txt nhưng là các phụ thuộc gián tiếp hoặc không được sử dụng trong logic cốt lõi hiện tại. |

***Ghi chú:*** Hệ thống cũng sử dụng các thư viện chuẩn của Python như threading để xử lý đa luồng (phát âm thanh cảnh báo) và datetime để quản lý thời gian, nhưng chúng không yêu cầu cài đặt riêng.

***Lưu ý quan trọng:*** Hệ thống yêu cầu mô hình nhận diện tư thế đã được huấn luyện trước. Đường dẫn đến mô hình này được khai báo trong file cấu hình tại tham số MODEL\_PATH: ./models/yolov8n-pose.pt. Cần đảm bảo file mô hình này tồn tại ở đúng đường dẫn đã chỉ định.

Với môi trường đã được thiết lập, phần tiếp theo sẽ phân tích cách hệ thống được điều khiển thông qua các tham số cấu hình.

**4.0 Phân tích Module Cấu hình (settings\_loader.py)**

Module settings\_loader.py đóng vai trò chiến lược như một trung tâm điều khiển tập trung của hệ thống. Việc tách biệt toàn bộ các tham số cấu hình ra khỏi logic mã nguồn chính mang lại lợi ích to lớn: nó cho phép các kỹ sư và người vận hành dễ dàng tinh chỉnh hành vi của hệ thống, thử nghiệm các kịch bản khác nhau và thích ứng với các điều kiện môi trường cụ thể mà không cần sửa đổi mã lệnh.

Dưới đây là phân tích chi tiết về các nhóm tham số chính và tác động của chúng đến hoạt động của hệ thống:

|  |  |  |
| --- | --- | --- |
| Nhóm Tham số | Tham số | Mô tả và Tác động |
| **VIDEO & MODEL** | VIDEO\_SOURCE | Xác định nguồn cấp video đầu vào (ví dụ: 0 cho webcam mặc định, 1 cho webcam thứ hai, hoặc đường dẫn đến file video). |
|  | MODEL\_PATH | Đường dẫn tới file mô hình YOLOv8 Pose (.pt). Thay đổi file này sẽ thay đổi mô hình nền tảng cho việc nhận diện và ước tính tư thế. |
| **THỜI GIAN** | SESSION\_DURATION | Tổng thời lượng của phiên giám sát, tính bằng giây. Hệ thống sẽ tự động kết thúc và xuất báo cáo khi hết thời gian này. |
|  | OUT\_TIMEOUT | Ngưỡng thời gian (giây) để xác định một người đã rời khỏi vị trí. Nếu một vị trí không có ai trong khoảng thời gian này, hệ thống sẽ cảnh báo và bắt đầu tính thời gian vắng mặt. |
|  | SLEEP\_CONFIRM\_TIME | Thời gian (giây) mà mắt người tham gia phải nhắm liên tục (dựa trên ngưỡng EAR) trước khi hệ thống chính thức xác nhận trạng thái "SLEEPING". Tăng giá trị này giúp giảm báo động giả. |
| **GÓC NHÌN** | DOWN\_THRESHOLD | Ngưỡng góc (độ) để xác định hành vi cúi đầu. Góc được tính giữa vector đầu và phương thẳng đứng. Tăng giá trị này sẽ làm hệ thống "ít nhạy" hơn với các hành vi cúi đầu nhẹ. |
|  | TURN\_THRESHOLD | Ngưỡng góc (độ) để xác định hành vi quay ngang. Tăng giá trị này đòi hỏi người tham gia phải quay đầu một góc lớn hơn để bị ghi nhận là mất tập trung. |
| **BUỒN NGỦ** | EAR\_SLEEP\_THRESHOLD | Ngưỡng Tỷ lệ Khung mắt (EAR). Nếu EAR trung bình giảm xuống dưới ngưỡng này, hệ thống sẽ coi đó là dấu hiệu của việc nhắm mắt. |
|  | EAR\_FRAMES\_CONFIRM | Số lượng khung hình liên tiếp được sử dụng để tính giá trị EAR trung bình, giúp làm mượt dữ liệu và tránh báo động giả do chớp mắt thông thường. |
| **SPATIAL TRACKING** | SPATIAL\_TOLERANCE | Khoảng cách tối đa (pixels) để liên kết một người được phát hiện với một vị trí không gian đã được "học". Giá trị lớn hơn cho phép người tham gia di chuyển nhiều hơn trong khu vực của họ mà không bị coi là một người mới. |
|  | FRAME\_LEARNING\_PERIOD | Số lượng khung hình đầu tiên mà hệ thống dùng để học và xác định các vị trí ngồi cố định trong khung hình. |
| **HIỂN THỊ** | WINDOW\_WIDTH, WINDOW\_HEIGHT | Kích thước của cửa sổ hiển thị video đầu ra. |
|  | WARNING\_BOX\_SIZE | Kích thước (pixels) của hộp cảnh báo nhấp nháy được vẽ xung quanh một vị trí bị bỏ trống. |
|  | BLINK\_SPEED | Tần suất nhấp nháy của hộp cảnh báo vắng mặt. |
| **LƯU FILE** | SUMMARY\_FILE | Tên của file CSV chứa báo cáo tổng kết cuối phiên. |
|  | DETAIL\_LOG\_FILE | Tên của file CSV chứa nhật ký chi tiết của mọi thay đổi trạng thái trong suốt phiên giám sát. |

Các tham số này sẽ được áp dụng trong luồng xử lý dữ liệu chi tiết, sẽ được trình bày ở phần tiếp theo.

**5.0 Luồng Xử lý Dữ liệu Chi tiết**

Luồng xử lý dữ liệu của hệ thống được chia thành hai giai đoạn tuần tự và khác biệt: **Giai đoạn 1: Học Layout Không gian** và **Giai đoạn 2: Giám sát và Phân tích Thời gian thực**. Việc phân tách này rất quan trọng: giai đoạn đầu tiên thiết lập một bản đồ tham chiếu về vị trí của những người tham gia, tạo ra các "khung không gian" ổn định để giai đoạn thứ hai có thể liên kết và theo dõi hành vi của từng cá nhân một cách nhất quán.

**5.1 Giai đoạn 1: Học Layout Không gian (Spatial Layout Learning)**

Mục đích của giai đoạn này là tự động xác định các vị trí ngồi cố định của những người tham gia trong khung hình. Quá trình này diễn ra trong một khoảng thời gian ban đầu được định nghĩa bởi tham số FRAME\_LEARNING\_PERIOD.

1. **Thu thập Vị trí:** Trong FRAME\_LEARNING\_PERIOD khung hình đầu tiên, hệ thống chạy mô hình nhận diện người trên mỗi khung hình. Với mỗi người được phát hiện, hệ thống tính toán tọa độ trung tâm của bounding box (cx, cy) và lưu trữ tất cả các tọa độ này vào một danh sách tạm thời là learning\_positions.
2. **Lọc và Tạo Khung Tham chiếu:** Sau khi kết thúc giai đoạn thu thập, hàm learn\_spatial\_layout được gọi. Thuật toán này không chỉ đơn thuần lọc các điểm trùng lặp mà xây dựng một tập hợp các "mỏ neo" không gian một cách lặp đi lặp lại. Nó lấy vị trí đầu tiên làm mỏ neo, sau đó duyệt qua các vị trí còn lại. Một vị trí mới chỉ được thêm vào làm mỏ neo duy nhất nếu khoảng cách của nó tới *tất cả các mỏ neo đã được thiết lập trước đó* đều lớn hơn ngưỡng SPATIAL\_TOLERANCE. Cách tiếp cận này đảm bảo tạo ra các vùng không gian riêng biệt, có khả năng chống chịu tốt với các chuyển động nhỏ của người tham gia. Kết quả là một tập hợp các "khung không gian" (spatial\_frames) duy nhất, mỗi khung đại diện cho một vị trí ngồi trong cuộc họp.

**5.2 Giai đoạn 2: Vòng lặp Giám sát và Phân tích Thời gian thực**

Sau khi đã học xong layout không gian, hệ thống chuyển sang vòng lặp xử lý chính, nơi mỗi khung hình từ luồng video được phân tích liên tục cho đến khi kết thúc phiên.

1. **Nhận diện và Theo dõi (Detection & Tracking):** Mỗi khung hình được đưa vào mô hình YOLO (model.predict) để phát hiện tất cả những người có mặt. Kết quả phát hiện này sau đó được chuyển đến bộ theo dõi ByteTrack (tracker.update\_with\_detections). ByteTrack gán cho mỗi người một track\_id duy nhất và cố gắng duy trì ID này qua các khung hình liên tiếp, ngay cả khi người đó bị che khuất tạm thời.
2. **Liên kết Không gian (Spatial Association):** Với mỗi người được theo dõi (có track\_id), hệ thống sử dụng hàm find\_or\_create\_frame\_id. Hàm này tính toán khoảng cách từ vị trí hiện tại của người đó đến trung tâm của tất cả các spatial\_frames đã học. Nếu mỏ neo gần nhất nằm trong bán kính SPATIAL\_TOLERANCE, người đó sẽ được liên kết với frame\_id tương ứng. Nếu không có mỏ neo nào đủ gần, điều này cho thấy một người mới đã xuất hiện ở một vị trí mới, và hệ thống sẽ tự động tạo ra một spatial\_frame mới tại vị trí đó. Điều này giúp liên kết một định danh theo dõi động (track\_id) với một vị trí không gian tĩnh (frame\_id).
3. **Trích xuất Tên (Name Extraction):** Để làm cho báo cáo dễ đọc hơn, hàm extract\_name\_from\_region được gọi định kỳ. Nó cắt ra vùng ảnh ngay bên dưới bounding box của người tham gia và áp dụng pytesseract (OCR) để đọc văn bản, giả định rằng tên của người đó được hiển thị ở đó (ví dụ: trong giao diện Zoom hoặc Google Meet).
4. **Phân tích Hành vi (Behavior Analysis):** Đây là bước cốt lõi. Đối với mỗi người được theo dõi và đã được liên kết với một frame\_id, hệ thống trích xuất các keypoints tư thế và vùng mặt của họ. Dữ liệu này sau đó được chuyển đến các hàm trong pose\_utils.py và meeting\_monitor.py để đánh giá trạng thái. Hệ thống xác định một trong các trạng thái sau: SLEEPING, LOOKING {direction} (LEFT/RIGHT), LOOKING DOWN, hoặc FOCUSED.
5. **Quản lý Trạng thái Vắng mặt (Absence Management):** Hệ thống kiểm tra xem có spatial\_frames nào không được liên kết với bất kỳ người nào hiện tại không (occupied\_frames). Nếu một vị trí bị bỏ trống quá thời gian OUT\_TIMEOUT, hệ thống sẽ đánh dấu vị trí đó là trống và bắt đầu tính thời gian vắng mặt (out\_time).
6. **Tổng hợp và Ghi nhận Dữ liệu (Data Aggregation & Logging):** Các chỉ số hành vi (ví dụ: down\_time, turn\_time, sleep\_time) được cập nhật liên tục vào một từ điển trung tâm là person\_stats theo từng frame\_id. Mỗi khi trạng thái của một người thay đổi, một bản ghi chi tiết bao gồm dấu thời gian, tên, trạng thái và các chỉ số liên quan được thêm vào danh sách meeting\_log.
7. **Hiển thị Trực quan (Visualization):** Cuối cùng, OpenCV được sử dụng để vẽ các thông tin phân tích trực tiếp lên khung hình video. Điều này bao gồm việc vẽ các bounding box, hiển thị tên, trạng thái hiện tại (FOCUSED, SLEEPING, v.v.), và các thông số thống kê chi tiết về thời gian mất tập trung.

Phần tiếp theo sẽ đi sâu vào các công thức toán học và logic đằng sau các thuật toán phân tích hành vi này.

**6.0 Phân tích các Thuật toán Cốt lõi (pose\_utils.py)**

Module pose\_utils.py là nơi chứa đựng tài sản trí tuệ cốt lõi của hệ thống. Nó bao gồm các hàm chuyên biệt để biến đổi dữ liệu thô cấp thấp—chẳng hạn như tọa độ pixel của các keypoints—thành các chỉ số hành vi cấp cao, có thể diễn giải được.

**6.1 Thuật toán Phát hiện Cúi đầu (head\_down\_angle)**

Thuật toán này định lượng mức độ cúi đầu của một người bằng cách tính toán góc nghiêng của đầu so với phương thẳng đứng.

* **Phương pháp luận:**
  1. Xác định các điểm mốc chính: Mũi (keypoints[0]), vai trái (keypoints[5]), và vai phải (keypoints[6]).
  2. Tính toán điểm trung tâm giữa hai vai: mid\_shoulder.
  3. Tạo một vector đại diện cho hướng của đầu: vector\_head = nose - mid\_shoulder.
  4. Tính toán góc giữa vector\_head và một vector dọc chuẩn hướng lên trên (0, -1). Góc này cho biết đầu đang ngẩng lên hay cúi xuống so với thân.
* **Quản lý Trạng thái:**
  1. Hàm sử dụng một biến trạng thái (\_down\_state) để theo dõi việc đếm thời gian. Khi góc tính được vượt qua ngưỡng DOWN\_THRESHOLD, trạng thái downing được đặt thành True và hệ thống bắt đầu tích lũy total\_time.
  2. Sự kết hợp giữa phát hiện cúi đầu và quay ngang là một lựa chọn thiết kế quan trọng để ngăn chặn các báo động giả. Một người quay đầu sang ngang tự nhiên sẽ khiến vector đầu tạo một góc lớn so với trục dọc. Bằng cách tạm dừng bộ đếm thời gian cúi đầu (down\_time) trong khi một sự kiện quay ngang (is\_turning) đang diễn ra, hệ thống phân biệt một cách chính xác giữa hành vi nhìn xuống và nhìn sang một bên.

**6.2 Thuật toán Phát hiện Quay ngang (head\_turn\_angle)**

Thuật toán này ước tính góc quay ngang của đầu dựa trên vị trí tương đối của mũi so với hai tai.

* **Phương pháp luận:**
  1. Xác định các điểm mốc: Mũi (keypoints[0]), tai trái (keypoints[3]), và tai phải (keypoints[4]).
  2. Tính toán điểm trung tâm giữa hai tai (ear\_center\_x) và chiều rộng của đầu (head\_width).
  3. Tính độ lệch ngang của mũi so với trung tâm hai tai: horizontal\_diff.
  4. Ước tính góc quay bằng cách tính tỷ lệ abs(horizontal\_diff) / head\_width và nhân với 90 độ. Thuật toán này dựa trên giả định rằng có một mối tương quan trực tiếp giữa góc quay và sự dịch chuyển theo phương ngang của mũi so với điểm trung tâm giữa hai tai. Khi nhìn thẳng, mũi ở giữa (ratio gần 0). Khi đầu quay, mũi lệch sang một bên, làm tăng ratio một cách tương ứng. Hướng quay (left/right) được xác định dựa trên dấu của horizontal\_diff.
* **Quản lý Trạng thái:** Tương tự như phát hiện cúi đầu, hàm này sử dụng biến trạng thái \_turn\_state để bắt đầu và dừng việc đếm thời gian total\_time khi góc ước tính vượt qua TURN\_THRESHOLD.

**6.3 Thuật toán Phát hiện Buồn ngủ (detect\_drowsiness\_frame)**

Thuật toán phát hiện buồn ngủ là một hệ thống con tinh vi, với logic cốt lõi nằm trong hàm detect\_drowsiness\_frame của meeting\_monitor.py. Nó được thiết kế để phân biệt giữa chớp mắt bình thường và trạng thái nhắm mắt kéo dài, dựa trên chỉ số Tỷ lệ Khung mắt (Eye Aspect Ratio - EAR).

* **Trích xuất Điểm mốc và Tính EAR:**
  + Hệ thống sử dụng mediapipe.solutions.face\_mesh để tạo ra một lưới điểm mốc chi tiết trên khuôn mặt, cung cấp tọa độ chính xác cho các điểm quanh mắt.
  + Hàm \_ear\_ratio (trong pose\_utils.py) triển khai công thức EAR kinh điển, tính toán tỷ lệ giữa chiều cao và chiều rộng của mắt. Khi mắt nhắm, tỷ lệ này giảm đi đáng kể.
* **Logic Xử lý và Xác nhận trong detect\_drowsiness\_frame:**
  + **Làm mượt Dữ liệu bằng Bộ đệm:** Thay vì phản ứng với giá trị EAR của một khung hình duy nhất, hệ thống duy trì một bộ đệm (ear\_buffer) lưu trữ các giá trị EAR của EAR\_FRAMES\_CONFIRM khung hình gần nhất. Giá trị EAR trung bình của bộ đệm này được sử dụng để so sánh, giúp loại bỏ các dao động đột ngột do chớp mắt và mang lại sự ổn định.
  + **So sánh với Ngưỡng:** Giá trị EAR trung bình được so sánh với EAR\_SLEEP\_THRESHOLD. Nếu thấp hơn, hệ thống bắt đầu một quy trình xác nhận.
  + **Xác nhận theo Thời gian:** Đây là cơ chế cốt lõi để tránh báo động giả. Hệ thống không ngay lập tức chuyển sang trạng thái ngủ. Thay vào đó, nó yêu cầu người dùng phải duy trì trạng thái nhắm mắt (EAR trung bình dưới ngưỡng) trong một khoảng thời gian liên tục là SLEEP\_CONFIRM\_TIME (ví dụ: 5 giây). Chỉ sau khi điều kiện này được thỏa mãn, biến trạng thái is\_sleeping trong person\_stats mới được đặt thành True. Biến sleep\_start được dùng để theo dõi thời điểm bắt đầu của giai đoạn này.
  + **Cảnh báo Non-blocking:** Khi trạng thái ngủ được xác nhận, một luồng xử lý riêng biệt (threading.Thread) được khởi tạo để phát âm thanh cảnh báo. Điều này cho phép âm thanh được phát mà không làm gián đoạn hay đóng băng luồng xử lý video chính.

Phần cuối cùng của tài liệu sẽ mô tả cách hệ thống tổng hợp và xuất các kết quả phân tích này ra thành các báo cáo hữu ích.

**7.0 Quản lý Dữ liệu và Đầu ra**

Giá trị cuối cùng mà hệ thống mang lại nằm ở khả năng chuyển đổi các phân tích phức tạp thành những báo cáo dữ liệu rõ ràng, hữu ích và có thể hành động. Hệ thống tạo ra hai loại file đầu ra chính, mỗi loại phục vụ một mục đích khác nhau: một file tổng kết cho cái nhìn tổng quan và một file nhật ký cho việc phân tích chi tiết.

Dưới đây là mô tả chi tiết về hai file CSV đầu ra:

|  |  |  |
| --- | --- | --- |
| Tên File (Filename) | Mục đích (Purpose) | Cấu trúc Dữ liệu (Data Structure) |
| **meeting\_summary.csv**<br/>(từ SUMMARY\_FILE) | Cung cấp một báo cáo tổng kết toàn diện về hiệu suất tập trung của từng người tham gia sau khi phiên họp kết thúc. File này lý tưởng cho việc đánh giá nhanh và so sánh. | - **Frame\_Name, Frame\_ID:** Định danh của người tham gia.<br/>- **Sleep\_Time, Down\_Time, Turn\_Time, Out\_Time:** Tổng thời gian (giây) tích lũy cho mỗi hành vi mất tập trung.<br/>- **Total\_Distraction:** Tổng của bốn loại thời gian mất tập trung ở trên.<br/>- **Distraction\_Rate\_%:** Tỷ lệ phần trăm thời gian mất tập trung so với tổng thời gian phiên họp.<br/>- **Focus\_Status:** Kết luận cuối cùng, được xác định là "TẬP TRUNG TỐT" hoặc “ MẤT TẬP TRUNG". |
| **meeting\_detailed\_log.csv**<br/>(từ DETAIL\_LOG\_FILE) | Cung cấp một nhật ký chi tiết, ghi lại mọi thay đổi trạng thái của từng người tham gia theo từng dấu thời gian. File này hữu ích cho việc truy vết và phân tích sâu các sự kiện cụ thể. | - **timestamp:** Thời gian sự kiện xảy ra (HH:MM:SS).<br/>- **frame\_id, frame\_name:** Định danh của người tham gia.<br/>- **state:** Trạng thái tại thời điểm đó (ví dụ: FOCUSED, SLEEPING, LOOKING DOWN).<br/>- **ear:** Giá trị EAR tại thời điểm đó.<br/>- **sleep\_time, down\_time, turn\_time, out\_time:** Các giá trị thời gian tích lũy tính đến thời điểm ghi nhận. |

Logic tính toán cuối cùng trong báo cáo tổng kết dựa trên hai chỉ số chính:

* **Tỷ lệ Mất tập trung (Distraction\_Rate\_%):** Được tính bằng công thức:
* **Trạng thái Tập trung (Focus\_Status):** Hệ thống so sánh giá trị Distraction\_Rate\_% với một ngưỡng được định nghĩa trong mã nguồn (DISTRACTION\_THRESHOLD, được đặt là 0.50). Nếu tỷ lệ mất tập trung vượt quá 50%, trạng thái cuối cùng sẽ là "MẤT TẬP TRUNG"; ngược lại, sẽ là "TẬP TRUNG TỐT".

**8.0 Kết luận**

Tài liệu này đã cung cấp một cái nhìn toàn diện và sâu sắc về kiến trúc, luồng hoạt động và các thuật toán cốt lõi của hệ thống Giám sát Tập trung Cuộc họp. Hệ thống đã chứng minh được năng lực trong việc tự động hóa quá trình theo dõi và định lượng các hành vi liên quan đến sự tập trung, từ việc xác định vị trí không gian, theo dõi cá nhân, đến việc phân tích các tư thế phức tạp và trạng thái sinh lý như buồn ngủ.

Tài liệu kỹ thuật này đóng vai trò là một nguồn tài nguyên thiết yếu cho các nhà phát triển và kỹ sư hệ thống. Nó không chỉ làm rõ "cái gì" và "làm thế nào" hệ thống hoạt động, mà còn cung cấp nền tảng vững chắc cho các hoạt động bảo trì, gỡ lỗi và đặc biệt là phát triển các tính năng mới trong tương lai.

Để tiếp tục nâng cao hiệu quả của hệ thống, một số hướng phát triển tiềm năng có thể được xem xét:

* **Cải thiện độ chính xác của OCR:** Tích hợp các mô hình OCR tiên tiến hơn hoặc các kỹ thuật tiền xử lý ảnh để tăng khả năng nhận dạng tên trong các điều kiện ánh sáng và phông nền khác nhau.
* **Mở rộng bộ hành vi:** Nghiên cứu và bổ sung các thuật toán để phát hiện các dấu hiệu mất tập trung khác, chẳng hạn như sử dụng điện thoại di động hoặc tương tác với các đối tượng không liên quan.
* **Tích hợp hệ thống:** Xây dựng các API để tích hợp dữ liệu đầu ra của hệ thống với các nền tảng quản lý học tập (LMS), hệ thống quản lý cuộc họp, hoặc các công cụ phân tích dữ liệu lớn hơn.