HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

CAO MINH QUYỀN

NGHIÊN CỨU HỆ THỐNG THEO DÕI MẮT ĐỂ PHÁT HIỆN TÀI XẾ KHÔNG TỈNH TÁO BẰNG CÁCH SỬ DỤNG DEEP LEARNING

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

HÀ NỘI - 2025

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

CAO MINH QUYỀN

NGHIÊN CỨU HỆ THỐNG THEO DÕI MẮT ĐỂ PHÁT HIỆN TÀI XẾ KHÔNG TỈNH TÁO BẰNG CÁCH SỬ DỤNG DEEP LEARNING

Chuyên ngành: Khoa học máy tính

Mã số: 8.48.01.01

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. ĐỖ TIẾN DŨNG

HÀ NỘI - 2025

MỞ ĐẦU

Giấc ngủ đóng vai trò thiết yếu đối với sức khỏe thể chất và tinh thần của con người. Khi chất lượng hoặc thời lượng giấc ngủ không được đảm bảo đầy đủ, một trong những hậu quả dễ nhận thấy nhất là tình trạng mệt mỏi (fatigue) và buồn ngủ (drowsiness) trong các hoạt động đòi hỏi sự tập trung cao như lái xe. Nhiều nghiên cứu đã chỉ ra rằng áp lực công việc, sử dụng thiết bị điện tử (máy tính, điện thoại thông minh,...) kéo dài và thói quen sinh hoạt thiếu khoa học là những nguyên nhân trực tiếp dẫn đến thiếu ngủ (sleep deprivation), từ đó làm tăng đáng kể nguy cơ buồn ngủ khi điều khiển phương tiện giao thông. Trong bối cảnh giao thông hiện đại, buồn ngủ và mất tỉnh táo khi lái xe được xem là một trong những nguyên nhân quan trọng gây ra tai nạn giao thông. Tài xế có thể có các biểu hiện như: mắt nhắm lâu, tần suất chớp mắt tăng, gật đầu, phản xạ chậm với tình huống bất ngờ… nhưng bản thân họ không ý thức được mức độ nguy hiểm của tình trạng này. Những khoảng thời gian mất tập trung chỉ kéo dài vài giây cũng đủ gây ra va chạm với thiệt hại lớn về người và tài sản. Đề án này tập trung nghiên cứu và xây dựng một hệ thống theo dõi mắt nhằm phát hiện tình trạng buồn ngủ của tài xế dựa trên hình ảnh thời gian thực. Hệ thống tận dụng ưu thế của thị giác máy tính và các mô hình học sâu để nhận diện khuôn mặt, xác định vị trí mắt, phân loại trạng thái mắt (mở/nhắm) và cuối cùng tính toán chỉ số PERCLOS để đánh giá mức độ buồn ngủ. Mục tiêu cuối cùng là đưa ra cảnh báo kịp thời, hỗ trợ người lái phòng tránh tai nạn do mất tập trung. Về mặt kỹ thuật, đề tài sử dụng kết hợp mạng nơ-ron tích chập xếp tầng đa nhiệm (Multi-task Cascaded Convolutional Networks – MTCNN) để phát hiện khuôn mặt và vị trí mắt, cùng mạng nơ-ron tích chập (Convolutional Neural Network – CNN) để phân loại trạng thái mắt. Thông tin trạng thái mắt theo chuỗi thời gian được dùng làm đầu vào cho thuật toán PERCLOS, từ đó ước lượng mức độ buồn ngủ và phát cảnh báo phù hợp. Việc khai thác khả năng xử lý của AI trong một bài toán có ý nghĩa xã hội như an toàn giao thông là động lực chính dẫn đến việc lựa chọn và thực hiện đề tài này.

CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI

1.1. Giới thiệu đề tài

Đề tài nghiên cứu hệ thống theo dõi mắt nhằm phát hiện sớm dấu hiệu buồn ngủ/mất tỉnh táo của người lái. Hệ thống hướng tới khả năng hoạt động gần thời gian thực, xuất cảnh báo khi người lái có biểu hiện nhắm mắt kéo dài hoặc tần suất nhắm mắt bất thường.

.

1.2. Lý do chọn đề tài

Đề tài được lựa chọn do: tình hình cấp thiết về an toàn giao thông; khả năng ứng dụng của thị giác máy tính và học sâu trong giám sát người lái; tiềm năng triển khai rộng rãi nhờ cảm biến camera phổ biến và chi phí thấp.

1.3. Mục đích nghiên cứu

Mục đích là xây dựng mô hình và hệ thống có khả năng: Đầu tiền là phát hiện khuôn mặt và định vị vùng mắt. Tiếp theo phân loại trạng thái mắt mở/nhắm ổn định. Sau đó tính toán mức độ buồn ngủ theo thời gian bằng PERCLOS. Cuối cùng đưa ra cảnh báo phù hợp nhằm hỗ trợ người lái.

1.4. Đối tượng nghiên cứu

Đối tượng nghiên cứu là dữ liệu hình ảnh/video khuôn mặt người lái trong cabin xe (hoặc môi trường tương tự) và các đặc trưng liên quan đến mắt (độ mở mí, chuỗi trạng thái mở/nhắm theo thời gian).

1.5. Phương pháp nghiên cứu

Đề tài sử dụng phương pháp tiếp cận tổng hợp, kết hợp cơ sở lý thuyết vớ thực nghiệm trên dữ liệu và triển khai hệ thống để đánh giá. Đầu tiên nghiên cứu các lý thuyết liên quan đến sự mất tỉnh táo khi lái xe. Tiếp theo là nghiên cứu về các kiến thức của mạng nơ-ron tích chập, kiến trúc MTCNN, cùng sác thuật toán tối ưu khác. Sau đó, sẽ thu thập và xử lý dữ liệu và dùng dữ liệu đó để xây dựng và huấn luyện mô hình. Cuối cùng, tác giả phân tích kết quả, chỉ ra các hạn chế, từ đó đưa ra các hướng phát triển kế tiếp cho hệ thống

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ NGHIÊN CỨU LIÊN QUAN

2.1. Khái niệm buồn ngủ và các biểu hiện sinh lý

Buồn ngủ khi lái xe thường đi kèm các dấu hiệu như: giảm tần suất chớp mắt bình thường, tăng thời gian nhắm mắt, nhìn lơ đãng, phản ứng chậm và có thể xuất hiện micro-sleep. Trong các dấu hiệu thị giác, trạng thái mắt là tín hiệu dễ quan sát và có tương quan mạnh với mức độ buồn ngủ.

2.2. Các phương pháp phát hiện buồn ngủ hiện có:

Các phương thức nhận biết buồn ngủ được biết đền có thể chia thành ba nhóm chính: dựa trên tín hiệu sinh lý, dựa trên hành vi điều khiển xe và dựa trên thị giác máy tính. Mỗi nhóm có những ưu và nhược điểm riêng, và việc lựa chọn hướng tiếp cận nào cần dựa trên yêu cầu ứng dụng, mức độ chấp nhận thiết bị đo đạc cũng như chi phí triển khai.

Nhóm đầu tiên dựa trên tín hiệu sinh lý (EEG, EOG, nhịp tim...): độ chính xác cao nhưng xâm lấn và khó triển khai rộng rãi.

Nhóm thứ hai dựa trên hành vi điều khiển xe (lệch làn, góc vô lăng...): không xâm lấn nhưng phụ thuộc bối cảnh và cảm biến xe.

Nhóm thứ ba dựa trên thị giác máy tính (khuôn mặt, mắt, ngáp, tư thế đầu...): phù hợp triển khai với camera, nhưng nhạy với ánh sáng/che khuất.

2.3. Thuật toán PERCLOS

Wierwille et al. [13] đã phát triển PERCLOS như một chỉ số đo lường tình trạng buồn ngủ. PERCLOSURE được định nghĩa là tỷ lệ phần trăm thời gian mà mí mắt khép lại qua đồng tử ít nhất 80%. Các giá trị PERCLOS có thể được tính bằng cách sử dụng chuỗi video liên tục của hình ảnh mắt [6]. Để tính toán được chỉ số PERCLOS, nhiều phép đo phải được hoàn thành. Đầu tiên phải nhận dạng được khuôn mặt trong khung hình, sau đó là xác định và trích xuất hình ảnh mắt. Tiếp theo sẽ xác định mắt đang ở trạng thái mở hay nhắm. Từ chuỗi kết quả phân loại, trong một khoảng thời

gian xác định gồm E_t lần đếm mắt, hệ thống đếm được E_c số lần nhắm mắt. Từ đó giá trị PERCLOS được tính như sau:

$$\text{Công thức: PERCLOS} = \frac{E_c}{E_t} \qquad (2.3)$$

Nếu giá trị PERCLOS càng cao thì thời gian mắt ở trạng thái nhắm càng nhiều, từ đó mức độ buồn ngủ càng nghiêm trọng. Nhiều thử nghiệm thực tế cho thấy khi PERCLOS vượt qua một ngưỡng nhất định, nguy cơ xảy ra sai sót, phản ứng chậm, hoặc ngủ gật trong khi lái xe tăng mạnh [6], [16].

2.4. Mạng nơ-ron

Mạng nơ-ron nhân tạo mô phỏng cơ chế học dựa trên trọng số và hàm kích hoạt. Quá trình huấn luyện tối ưu tham số bằng lan truyền ngược (backpropagation) nhằm giảm hàm mất mát, từ đó mô hình có khả năng dự đoán nhãn cho dữ liệu mới.

2.4.1. Các thành phần trong mạng nơ-ron

Một mạng nơ-ron nhân tạo được hình thành từ các đơn vị tính toán (units), lấy ý tưởng từ nơ-ron sinh học. Mỗi đơn vị nhận các tín hiệu đầu vào có thể đến từ dữ liệu bên ngoài hoặc từ đầu ra của các đơn vị ở lớp trước. Các đơn vị này được kết nối với nhau thông qua các liên kết (connections), mỗi liên kết mang một trọng số (weight) thể hiện mức ảnh hưởng của đầu vào tương ứng lên đầu ra.

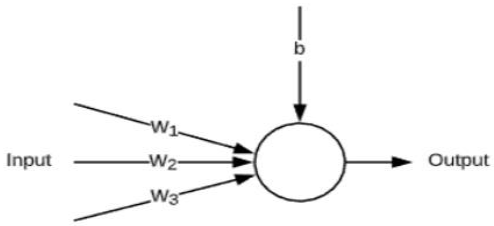
Dưới góc độ toán học, quá trình xử lý tại một nơ-ron có thể được diễn tả như sau. Giả sử x là vector đầu vào, W là vector trọng số tương ứng và b là độ lệch, giá trị hoạt hóa tuyến tính a được tính theo công thức:

$$a = W_x^T + b \qquad (2.1)$$

W_x^T là ma trận chuyển vị của trọng số. Giá trị a này chưa phải là đầu ra cuối cùng; ở bước tiếp theo, a được truyền qua một hàm kích hoạt phi tuyến tạo ra đầu ra y.

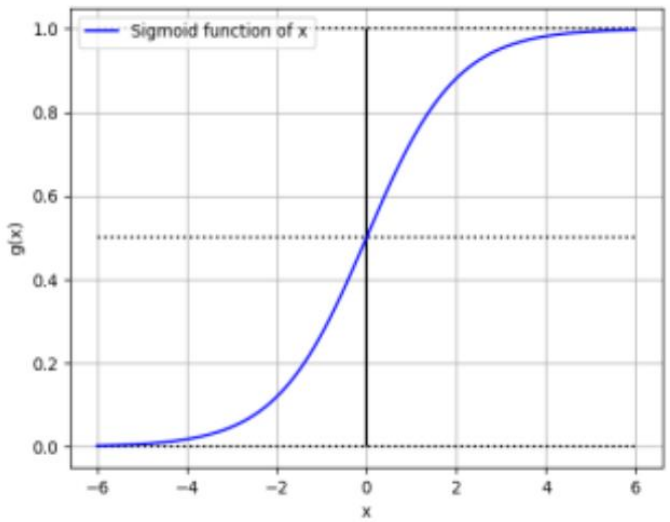
Các nơ-ron được tổ chức thành từng lớp: lớp đầu vào (input layer) nhận dữ liệu thô, một hoặc nhiều lớp ẩn (hidden layers) đảm nhiệm trích xuất và biến đổi đặc trưng, và lớp đầu ra (output layer) sinh ra kết quả cuối cùng. (Hình 2.1)

Hàm kích hoạt là một phần không thể thiếu trong từng rơ-ron, quyết định cách giá trị hoạt hóa tuyến tính a được biến đổi thành đầu ra y. Nếu bỏ qua hàm kích hoạt hoặc chỉ sử dụng các hàm tuyến tính, toàn bộ nơ-ron dù có nhiều lớp cũng vẫn chỉ như một phép biến đổi tuyến tính của đầu vào, không đủ khả năng biểu diễn các quan hệ phi tuyến phức tạp trong dữ liệu.



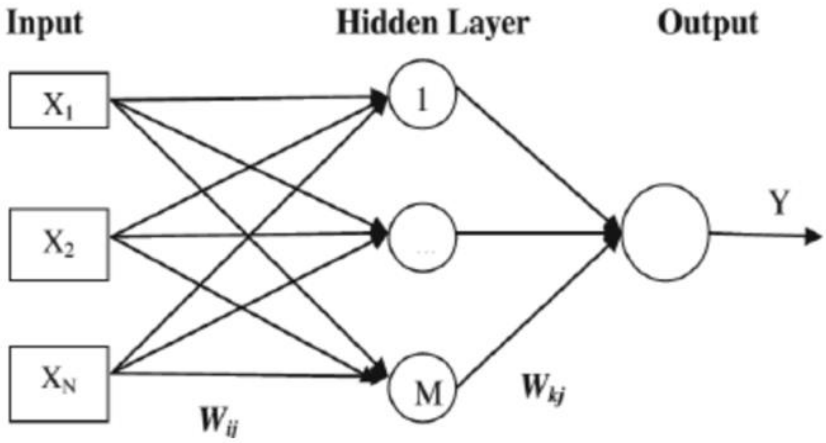
Hình 2.1: Một đơn vị trong mạng Nơ-ron

Bên cạnh ReLU, hàm sigmoid cũng là một hàm kích hoạt kinh điển, thường xuất hiện trong lớp đầu ra của các bài toán phân loại nhị phân. Hàm Sigmoid thường dùng để chỉ hàm logistic, gọi là *logistic sigmoid function*. Ngoài vai trò là hàm kích hoạt trong các lớp ẩn, sigmoid còn hữu ích trong nhiều thuật toán học máy khác, đặc biệt khi cần chuyển đổi một đại lượng thành giá trị xác suất. Khi được sử dụng ở lớp đầu ra của mô hình, hàm sigmoid giúp chuyển đổi kết quả dự đoán thành một điểm xác suất (likelihood score) dễ diễn giải và thuận lợi cho quá trình ra quyết định.



Hình 2.3: Hàm Sigmoid

Các đơn vị (units) trong mạng nơ-ron được sắp xếp thành các nhóm gọi là *lớp* (layers). Trong mỗi lớp, các đơn vị không kết nối trực tiếp với nhau; thay vào đó, chúng chỉ liên kết với các đơn vị thuộc lớp trước và lớp sau. Một mạng nơ-ron thông thường bao gồm ba loại lớp (Hình 2.4): lớp đầu vào (input layer), lớp ẩn (hidden layers) và lớp đầu ra (output layer).



Hình 2.4: Các lớp của một mạng nơ-ron

2.5. Mạng nơ-ron tích chập (CNN)

CNN đặc biệt phù hợp với dữ liệu ảnh nhờ cơ chế tích chập và chia sẻ trọng số, giúp trích xuất đặc trưng cục bộ (biên, góc, texture) và hình thành biểu diễn phân cấp.

2.5.1. Các thành phần của mạng nơ-ron tích chập

Về tổng thể, một mạng nơ-ron tích chập tiêu chuẩn thường bao gồm ba nhóm lớp chính: Lớp tích chập (Convolution layer – Conv layer), Các lớp gộp (Pooling layer), Các lớp kết nối đầy đủ (Fully Connected layer).

Những lớp này được sắp xếp thành nhiều tầng nối tiếp nhau, tạo thành một kiến trúc sâu có khả năng học biểu diễn phân cấp của dữ liệu: các lớp ở gần đầu vào tập trung vào những đặc trưng cục bộ đơn giản, còn các lớp ở giai đoạn sau tổng hợp các đặc trưng này thành những biểu diễn trừu tượng hơn.

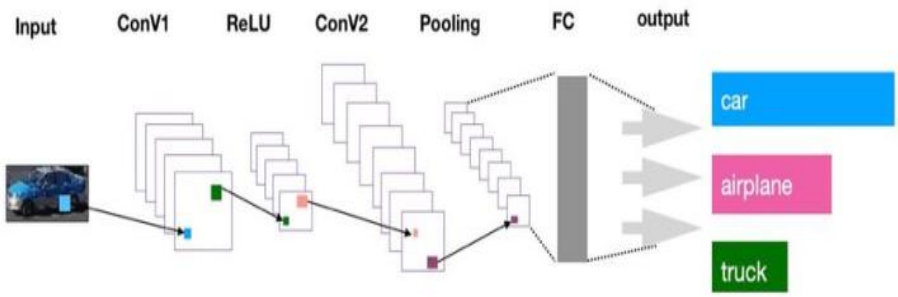
2.5.1.1. Lớp tích chập (Conv layer)

2.5.1.2. Các lớp gộp (Pooling layer)

2.5.1.3. Các lớp Kết nối đầy đủ

2.5.2. Kiến trúc mạng nơ-ron tích chập

Trong một mạng nơ-ron truyền thẳng thông thường, mỗi nơ-ron đầu vào được kết nối với mỗi nơ-ron đầu ra trong lớp tiếp theo. Kiến trúc thường xếp chồng các khối conv–ReLU–pooling, sau đó đưa qua các lớp dense để dự đoán xác suất thuộc lớp (softmax/sigmoid). Độ sâu và số lượng kernel được cân chỉnh để đạt cân bằng giữa độ chính xác và chi phí tính toán Theo Hình 2.7, chúng ta có thể thấy kiến trúc cơ bản của CNN để phát hiện ô tô trong hình ảnh.



Hình 2.7: Kiến trúc cơ bản của Mạng Nơ-ron Tích chập.

2.5.3. Huấn luyện mạng nơ-ron tích chập

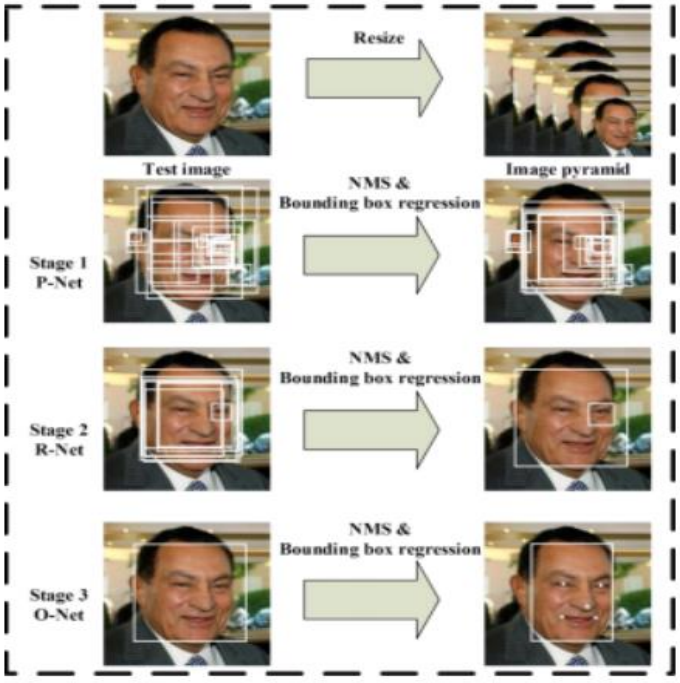
Huấn luyện mạng nơ-ron tích chập về bản chất là quá trình điều chỉnh trọng số của từng nơ-ron để thu được một mô hình có khả năng tổng quát tốt trên dữ liệu mới. Ban đầu, các trọng số của CNN được khởi tạo ngẫu nhiên. Khi một ảnh được đưa qua mạng, mô hình sinh ra dự đoán ở lớp đầu ra; dự đoán này được so sánh với nhãn đúng để tính toán hàm mất mát (loss).

2.6.Mạng Nơ-ron Tích chập xếp tầng đa nhiệm (Multi-task Cascaded Convolutional Networks)

Trong khi có nhiều phương pháp phát hiện khuôn mặt [20][19][10][26][25], nhưng MTCNN (Multi-task Cascaded Convolutional Networks - Mạng Nơ-ron Tích chập Xếp tầng Đa nhiệm) được sử dụng trong nghiên cứu này vì hai lý do. Một mặt, nó đạt được độ phát hiện chính xác cao (high detection accuracy), và mặt khác, mô hình FaceNet đã cung cấp sẵn giao diện MTCNN để phát hiện khuôn mặt.

2.6.1. Mạng Đề xuất (Proposal Network - P-NET)

P-Net là một mạng nơ-ron tích chập đầy đủ (Fully Convolutional Network - FCN). Bản đồ đặc trưng (feature map) của hàm truyền thẳng (forwarding propagation function) là một vector đặc trưng 32 chiều (32-dimensional feature vector) tại mỗi điểm. Nó được dùngđể quyết định xem các ô lưới 12x12 có chứa khuôn mặt hay không.



Hình 2.8: MTCNN:Các bước mà kiến trúc chuỗi tầng thực hiện để đảm bảo hiệu suất tốt nhất trong việc phát hiện khuôn mặt người và hồi quy hộp bao (bounding box regression).

2.6.2. Mạng Tinh chỉnh (Refine Network - R-NET)

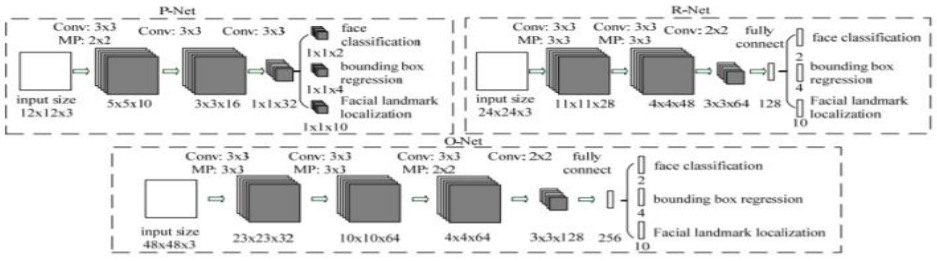
R-Net là mạng con thứ hai, hoạt động như một bộ lọc tinh, nhận đầu vào là các hộp đề xuất từ P-Net đã được cắt và chuẩn hóa kích thước. Cấu trúc của R-Net phức tạp hơn P-Net, với nhiều lớp tích chập và các lớp ẩn hơn, cho phép học được các đặc trưng phân biệt tốt hơn giữa vùng có khuôn mặt và vùng nhiễu.

2.6.3. Mạng Đầu ra (Output Network - O-NET)

O-NET giúp cải thiện độ chính xác (improve precision). Nó là một CNN đơn giản; Hộp giới hạn được tạo ra bởi bước P-Net có thể chứa hoặc không chứa khuôn mặt người. O-Net đánh giá sự hiện diện của khuôn mặt người. Khi một khuôn mặt được chứa đựng (được xác nhận), hộp giới hạn được hồi quy và trải qua cấp độ NMS.

2.6.4. Kiến trúc của MTCNN

Trong mô hình MTCNN dạng xếp tầng, mỗi giai đoạn sử dụng một kiến trúc lớp riêng với số lượng lớp và các bộ lọc tích chập (Conv filters) khác nhau, nhưng đều tạo ra ba loại đầu ra giống nhau. Đầu tiên là đầu ra phân loại khuôn mặt, trong đó hai nơ-ron được sử dụng để tạo ra điểm phân loại: một nơ-ron xác định sự hiện diện của khuôn mặt và nơ-ron còn lại thể hiện mức độ tin cậy hay thứ hạng. Tiếp theo là đầu ra hồi quy hộp giới hạn. Cuối cùng, mô hình còn thực hiện định vị các mốc khuôn mặt thông qua việc hồi quy năm điểm quan trọng gồm mắt phải, mắt trái, khóe miệng phải, khóe miệng trái và mũi, giúp tăng cường độ chính xác trong việc nhận diện và theo dõi đặc trưng khuôn mặt.



Hình 2.9: Kiến trúc của MTCNN

CHƯƠNG 3: THIẾT KẾ HỆ THỐNG

3.1.Phân tích hệ thống

3.1.1. Yêu cầu của hệ thống

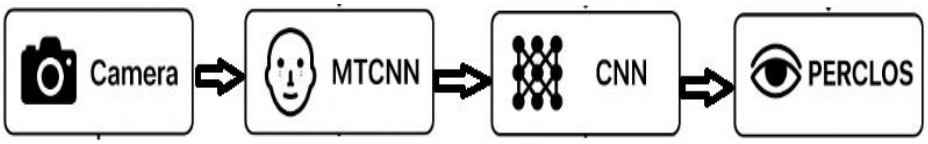
3.1.2. Phân tích lựa chọn kiến trúc hệ thống

3.1.3. Phân tích luồng xử lý và tương tác giữa các module

3.1.4. Đánh giá và nhận xét về thiết kế hệ thống

3.2.Mô hình tổng thể của hệ thống

Hệ thống được thiết kế theo pipeline xử lý khung hình (frame) từ camera đến quyết định cảnh báo. Luồng xử lý gồm 4 khối: (1) thu nhận hình ảnh; (2) phát hiện khuôn mặt và landmark bằng MTCNN; (3) trích xuất vùng mắt và phân loại mở/nhắm bằng CNN; (4) tính PERCLOS theo cửa sổ thời gian và kích hoạt cảnh báo.



Hình 3.1: Sơ đồ khối hệ thống

3.3. Mô tả chi tiết từng thành phần

Mục này trình bày đầy đủ và chi tiết kiến trúc cũng như chức năng của bốn thành phần cốt lõi trong hệ thống phát hiện buồn ngủ của người lái, bao gồm Camera, MTCNN, CNN và module tính toán PERCLOS. Mỗi thành phần có một vai trò độc lập nhưng có sự liên kết chặt chẽ để đảm bảo hệ thống vận hành trơn tru và chính xác trong thời gian thực.

3.3.1. Camera – Thiết bị thu nhận hình ảnh thời gian thực

Camera thu khung hình liên tục (thử nghiệm ở mức 24–30 fps) và cung cấp đầu vào cho hệ thống. Trong triển khai, khung hình được truy xuất bằng OpenCV; vị trí camera cần đảm bảo vùng mắt đủ rõ ở các điều kiện ánh sáng khác nhau

3.3.2. MTCNN – Phát hiện khuôn mặt và xác định landmark mắt

MTCNN phát hiện khuôn mặt trên từng khung hình, đồng thời trả về 5 landmark giúp xác định chính xác vị trí mắt. Nếu một khung hình không phát hiện được khuôn mặt, hệ thống có thể bỏ qua hoặc dùng trạng thái gần nhất để tránh nhiễu.

3.3.3. CNN – Phân loại trạng thái mắt (mở hoặc nhắm)

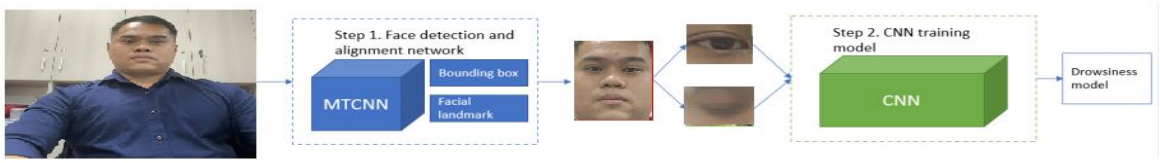
Vùng mắt sau khi cắt và chuẩn hóa được đưa vào CNN để dự đoán nhãn nhị phân. Mô hình CNN được thiết kế gọn nhẹ để phù hợp suy luận gần thời gian thực, đồng thời đảm bảo độ chính xác cao khi có nhiễu/che khuất.

3.3.4. PERCLOS – Tính toán mức độ buồn ngủ

Thành phần cuối cùng trong chuỗi xử lý là module PERCLOS, bộ phận chịu trách nhiệm chuyển đổi chuỗi trạng thái mắt theo thời gian thành một chỉ số định lượng về mức độ buồn ngủ. Chuỗi nhãn mắt theo thời gian được tổng hợp trong một cửa sổ quan sát để tính PERCLOS và suy ra mức độ buồn ngủ. Theo thiết kế ngưỡng trong đề án: PERCLOS 0–18% được xem là bình thường hoặc mệt nhẹ; 19–25% thể hiện nguy cơ buồn ngủ rõ hơn và cần cảnh báo; từ 26–30% (hoặc trên 30%) thể hiện rủi ro cao (micro-sleep/ngủ gật) và kích hoạt cảnh báo mạnh.

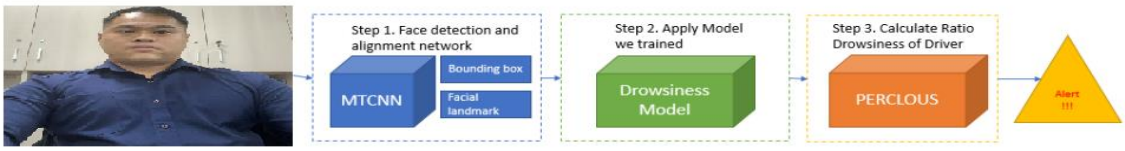
CHƯƠNG 4: XÂY DỰNG VÀ TRIỂN KHAI HỆ THỐNG

Hệ thống phát hiện buồn ngủ bằng thị giác máy tính cần được xây dựng như một chuỗi xử lý có tổ chức, trong đó mỗi bước đều phục vụ trực tiếp cho mục tiêu cuối cùng là đánh giá đúng mức độ buồn ngủ và đưa ra cảnh báo kịp thời. Trong đề tài này, giải pháp được xây dựng bao gồm hai bước. Bước đầu tiên là xây dựng tập dữ liệu và thực hiện tiền xử lý để tạo ra các ảnh mắt có chất lượng và định dạng phù hợp cho mô hình học sâu. Bước thứ hai là phát triển mô hình buồn ngủ dựa trên sự kết hợp giữa MTCNN, CNN và thuật toán PERCLOS, từ đó tạo nên một hệ thống theo dõi mắt có thể hoạt động gần thời gian thực. Hai bước này được thể hiện trong các sơ đồ khối ở Hình 4.1 và Hình 4.2, minh họa rõ luồng xử lý từ dữ liệu thô cho đến tín hiệu cảnh báo buồn ngủ gửi tới người lái.



Hình 4.1: Các bước tổng thể của Bộ dữ liệu và Tiền xử lý

Bước 1 bao gồm khuôn mặt và điểm mốc để tiếp cận, trạng thái mở mắt và bước 2 bao gồm mô hình buồn ngủ.



Hình 4.2: Các bước tổng thể của mô hình phát hiện buồn ngủ

Bước 1 nhận diện khuôn mặt bao gồm khuôn mặt và điểm mốc của mắt và bước áp dụng mô hình buồn ngủ. Bước 3: Sử dụng PERCLOS để cảnh báo buồn ngủ.

Trọng tâm của chương này là trình bày chi tiết tập dữ liệu và các bước tiền xử lý, kiến trúc mạng nơ-ron tích chập dùng để phát hiện buồn ngủ, cũng như mô tả cách triển khai hệ thống theo dõi mắt phát hiện buồn ngủ. Cuối chương, các kết quả thí

nghiệm về tốc độ xử lý, độ chính xác và khả năng triển khai thực tế sẽ được thảo luận, qua đó đánh giá mức độ đáp ứng của hệ thống đối với các yêu cầu đặt ra ở những chương trước.

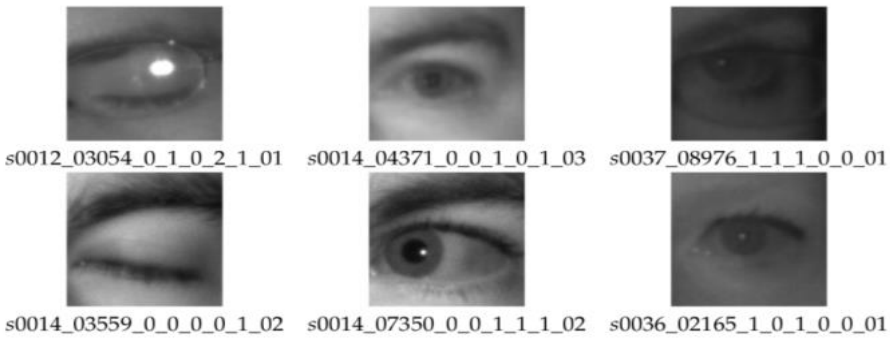
4.1. Tập dữ liệu và Tiền xử lý (Datasets and Preprocessing)

4.1.1. Tập dữ liệu (Datasets)

Dữ liệu ban đầu mà hệ thống thu được là dữ liệu thô dưới dạng ảnh hoặc video, và bản thân máy tính không thể hiểu trực tiếp ý nghĩa của những dữ liệu này nếu chúng không được tổ chức và gán nhãn một cách thích hợp.

Mô hình học sâu trong đề tài này được phát triển trên cơ sở các ảnh mắt đã được cắt và tiền xử lý. Các ảnh này xuất phát từ nguồn chính là tập dữ liệu mắt được thu thập từ Internet:

Dữ liệu Internet: bổ sung các tập dữ liệu mắt và video phổ biến như CEW (Closed Eyes In The Wild), NTHU-DDD và YawDD, UTA-RLDD (UTA Real-Life Drowsiness Dataset); đồng thời tham khảo MRL Eye Dataset để tăng độ đa dạng.



Hình 4.4: MRL Eye Dataset

Việc kết hợp tập dữ liệu sinh viên với MRL Eye Dataset giúp tăng đáng kể số lượng mẫu huấn luyện và kiểm thử, đồng thời làm phong phú thêm các kiểu mắt, điều kiện ánh sáng và góc chụp, qua đó nâng cao khả năng tổng quát hóa của mô hình phân loại mắt.

4.1.2. Tiền xử lý (Preprocessing)

Quy trình tiền xử lý được thực hiện nhất quán cho cả video và ảnh tĩnh:

- Tách khung hình từ video (độ phân giải nguồn thử nghiệm 2560×1820) với tốc độ trích xuất 24 khung hình/giây.

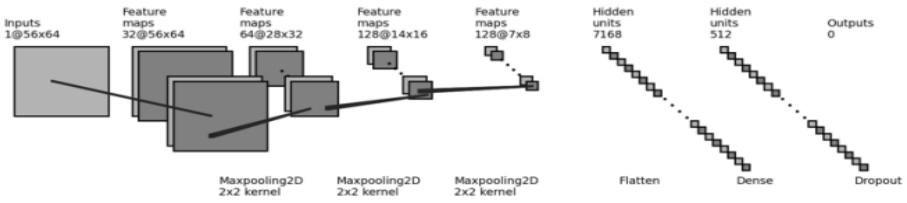
- Phát hiện khuôn mặt, cắt vùng mắt và chuẩn hóa kích thước ảnh mắt về 56×64 pixel để thống nhất đầu vào cho CNN.
- Chuyển ảnh màu sang mức xám (grayscale) nhằm giảm số kênh dữ liệu và tập trung vào đặc trưng hình học/độ sáng vùng mí mắt.
- Chuẩn hóa giá trị pixel (ví dụ đưa về [0,1]) và tổ chức dữ liệu theo thư mục nhãn "mở"/"nhắm".

- Chia dữ liệu thành train/validation/test: tập huấn luyện >35.000 khung hình mắt mở và 35.000 khung hình mắt nhắm; tập đánh giá khoảng 5.000 mở + 5.000 nhắm; tập kiểm thử 10.000 mở + 10.000 nhắm.

4.2. Mạng Nơ-ron Tích chập để Phát hiện Buồn ngủ

CNN trong đề án được xây dựng theo kiến trúc điển hình: nhiều lớp tích chập kết hợp ReLU để trích xuất đặc trưng cục bộ, xen kẽ các lớp max pooling để giảm kích thước và tăng tính bền vững. Ở phần cuối, đặc trưng được làm phẳng và đưa qua lớp kết nối đầy đủ để phân loại hai trạng thái "mắt mở" và "mắt nhắm" (softmax/sigmoid).

Trong giai đoạn huấn luyện, mô hình được tinh chỉnh các siêu tham số (tốc độ học, số epoch, batch size, số kernel...) để đạt độ hội tụ tốt. Đề án tận dụng Google Colab với GPU K80 để rút ngắn thời gian huấn luyện; kết quả học cho thấy độ chính xác vượt 90% và đường học ổn định.

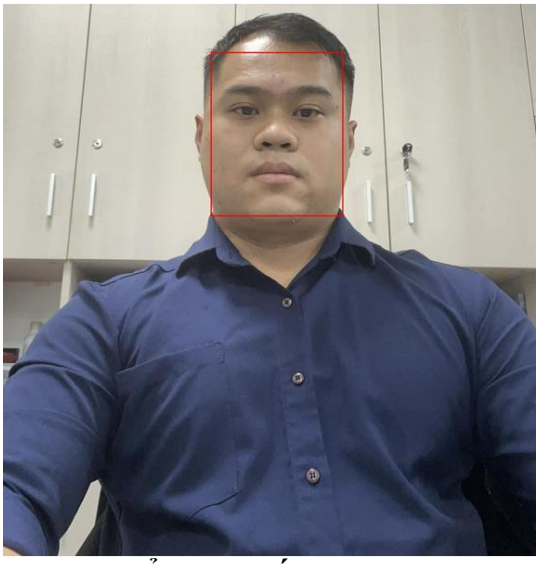


Hình 4.6: Mô hình CNN đào tạo hệ thống theo dõi mắt để phát hiện tình trạng buồn ngủ khi lái xe

4.3. Hệ thống Theo dõi Mắt Phát hiện Buồn ngủ (Drowsiness Detection Tracking Eye System)

4.3.1 Phát hiện Khuôn mặt bằng Mạng Nơ-ron Tích chập Xếp tầng Đa nhiệm vụ (MTCNN)

Trong vận hành, mỗi khung hình được đưa qua MTCNN để phát hiện khuôn mặt và xác định landmark. Vùng mắt được cắt theo vị trí landmark nhằm giảm ảnh hưởng của góc mặt và thay đổi vị trí..



Hình 4.12: MTCNN để trích xuất hộp giới hạn của khuôn mặt

4.3.2. Mạng Nơ-ron Tích chập Phát hiện Trạng thái Mắt (Convolutional Neural Networks Detect Eyes Status)

Hai ảnh mắt (trái/phải) sau tiền xử lý được CNN phân loại. Kết quả phân loại theo từng khung hình được lưu để tính PERCLOS theo cửa sổ thời gian.

4.3.3. PERCLOS và Cảnh báo (PERCLOS and Alert)

Hệ thống đếm số khung hình mắt nhắm trong cửa sổ quan sát và cập nhật PERCLOS liên tục. Dựa trên ngưỡng PERCLOS, hệ thống hiển thị trạng thái (awake/drowsy) và kích hoạt cảnh báo (âm thanh/thông báo màn hình) khi cần

4.4. Kết quả thí nghiệm

Các kết quả của thí nghiệm

Kết quả đo tốc độ xử lý của hệ thống khi: chỉ sử dụng camera, khi kết hợp với MTCNN và TensoFlow và khi kết hợp MTCNN với Pytorch.

Bảng 4.1: FPS khi chạy thử nghiệm hệ thống.

Kiểm tra FPS			
Tường hợp	Khung hình	Thời gian	FPS
Camera bình thường	100	3.8333(s)	26-27
Camera sử dụng MTCNN với TensorFlow	100	28.333(s)	3-4
Camera sử dụng MTCNN với PyTorch	100	10.667(s)	9-10

Kết quả đo hiệu năng của hệ thống với hai trường hợp không đeo kính và có đeo kính

Bảng 4.2: Kiểm tra trường hợp phát hiện buồn ngủ.

Trường hợp thử nghiệm				
Trường hợp	Accuracy	Precision	Recall	F1-score
Không đeo kình	92.852%	92.722%	93.237%	92.961%
Đeo Kính	90.469%	90.446%	90.446%	90.473%

Hai bảng sau so sánh hiệu năng của hệ thống được đề xuất với ba mô hình khác là: EAR + Dlib + ngưỡng, Haar Cascade + heuristics, MobileNetV2 transfer learning.

Bảng 4.3: So sánh các mô hình khi đối tượng không đeo kính

Mô hình	Accuracy	Precision	Recall	F1-score
MTCNN–CNN–PERCLOS	92.852%	92.722%	93.237%	92.961%
EAR + Dlib + ngưỡng	88.235%	87.308%	89.216%	88.253%
Haar Cascade + heuristics	86.436%	85.333%	87.463%	86.233%
MobileNetV2 TL cho mắt	90.223%	93.333%	85.667%	89.335%

Bảng 4.4: So sánh các mô hình khi đối tượng có đeo kính

Mô hình	Accuracy	Precision	Recall	F1-score
MTCNN–CNN–PERCLOS	90.469%	90.446%	90.446%	90.473%
EAR + Dlib + ngưỡng	81.683%	81.212%	82.677%	82.013%
Haar Cascade + heuristics	80.454%	79.587%	80.777%	80.152%
MobileNetV2 TL cho mắt	88.215%	93.127%	84.416%	88.515%

Từ hai bảng so sánh, có thể kết luận MTCNN–CNN–PERCLOS cho hiệu năng tổng thể tốt và ổn định hơn trong cả hai điều kiện. Việc kết hợp phát hiện/căn chỉnh vùng mắt bằng MTCNN, phân loại bằng CNN và tổng hợp theo thời gian bằng PERCLOS giúp giảm ảnh hưởng của nhiễu theo từng khung hình, nhờ đó phù hợp hơn với bài toán phát hiện buồn ngủ trong bối cảnh thực tế, đặc biệt khi có các yếu tố gây nhiễu như đeo kính và biến thiên ánh sáng.

KẾT LUẬN

Tóm lại, hệ thống theo dõi mắt phát hiện tình trạng buồn ngủ của người lái xe trong đề tài được xây dựng trên cơ sở kết hợp nhiều mô hình mạng nơ-ron, trong đó trọng tâm là Mạng nơ-ron (NN), Mạng nơ-ron tích chập (CNN) và Mạng nơ-ron tích chập xếp tầng đa nhiệm (MTCNN). Luận văn đã trình bày rõ kiến trúc, cơ chế học và cách áp dụng các mô hình này cho một bài toán học có giám sát, từ khâu phát hiện và căn chỉnh khuôn mặt, trích xuất vùng mắt cho đến phân loại trạng thái mắt mở/nhắm và tính toán chỉ số PERCLOS để đánh giá mức độ buồn ngủ. Trên cơ sở đó, một luồng xử lý hoàn chỉnh cho hệ thống đã được đề xuất, trong đó mỗi mô hình đảm nhận một nhiệm vụ cụ thể và phát huy được thế mạnh riêng, góp phần nâng cao hiệu quả tổng thể của hệ thống. Các kết quả thực nghiệm cho thấy mô hình kết hợp dựa trên MTCNN và CNN đạt hiệu năng tốt hơn so với việc sử dụng riêng lẻ các mô hình nơ-ron cơ bản, cả về độ chính xác phân loại trạng thái mắt và khả năng theo dõi buồn ngủ trong chuỗi thời gian. Điều này khẳng định tính phù hợp của hướng tiếp cận sử dụng kết hợp nhiều mạng nơ-ron chuyên biệt trong cùng một hệ thống. Tuy nhiên, bên cạnh những kết quả đạt được, hệ thống vẫn còn tồn tại những hạn chế nhất định cần tiếp tục được nghiên cứu và hoàn thiện trong tương lai.

Về hướng phát triển, trước hết cần chú trọng đến khả năng mở rộng và triển khai thực tế. Hệ thống hiện mới dừng lại ở mức cài đặt dưới dạng mã nguồn chạy trên máy tính cá nhân; trong các nghiên cứu tiếp theo, việc nhúng (embed) mô hình vào các nền tảng phần cứng như thiết bị di động, hệ thống giải trí trên ô tô hoặc các bộ điều khiển nhúng chuyên dụng sẽ được xem xét. Khi đó, hệ thống có thể hoạt động trực tiếp trên phương tiện, giám sát liên tục trạng thái mệt mỏi của người lái và đưa ra cảnh báo kịp thời, đồng thời có thể tích hợp thêm các chức năng hỗ trợ như gợi ý điểm dừng nghỉ, nhà nghỉ phù hợp giúp tài xế tạm thời nghỉ ngơi trước khi tiếp tục hành trình, góp phần giảm thiểu nguy cơ tai nạn giao thông. Bên cạnh khả năng triển khai, độ chính xác của hệ thống cũng là yếu tố cần được cải thiện. Trong tương lai, có thể khảo sát và thử nghiệm thêm các kiến trúc học sâu tiên tiến khác hoặc các mô hình lai ghép nhằm nâng cao hơn nữa tỷ lệ nhận diện đúng trạng thái mắt và mức độ

buồn ngủ. Ngoài ra, việc kết hợp thêm các đặc trưng khuôn mặt khác, chẳng hạn như tương quan giữa vị trí lông mày và mắt, giữa miệng và mắt hay các chỉ báo vi biểu cảm (micro-expression), cũng là một hướng tiềm năng để tăng độ tin cậy của mô hình. Sự kết hợp đa dạng các nguồn thông tin hình ảnh sẽ giúp hệ thống đánh giá chính xác hơn trạng thái tỉnh táo của người lái, qua đó nâng cao mức độ an toàn trong quá trình điều khiển phương tiện.