Tóm Tắt

1. Giới thiệu đề tài

* Trong những năm gần đây, tỷ lệ người cao tuổi trong dân số toàn cầu đang gia tăng nhanh chóng, kéo theo đó là các vấn đề về sức khỏe và an toàn cho nhóm đối tượng này. Té ngã là một trong những nguyên nhân hàng đầu gây chấn thương nghiêm trọng và thậm chí tử vong ở người cao tuổi. Theo Tổ chức Y tế Thế giới (WHO), mỗi năm có hàng triệu người trên thế giới gặp phải sự cố té ngã, dẫn đến các hậu quả nghiêm trọng về sức khỏe và tâm lý.
* Để thực hiện đề tài này nhóm đã nghiên cứu qua nhiều model và nhóm đã chọn model Yolo để làm đề tài này, với nhiều điểm nổi bật về sự nhanh chóng và chính xác cùng với dung lượng nhỏ Yolo đã trở thành model thích hợp cho đề tài của nhóm.

1. Lí do chọn đề tài

* Té ngã là một vấn đề nghiêm trọng đối với người cao tuổi, dẫn đến nhiều hậu quả nghiêm trọng như gãy xương, chấn thương đầu, và thậm chí tử vong. Việc phát triển một hệ thống nhận diện té ngã có thể giúp giảm thiểu các nguy cơ này bằng cách cung cấp cảnh báo kịp thời và hỗ trợ ngay lập tức. Điều này đặc biệt quan trọng trong bối cảnh xã hội hiện nay, khi tỷ lệ người cao tuổi ngày càng tăng.
* Ngoài ra, đề tài này có thể thúc đẩy sự phát triển của các ngành công nghiệp liên quan như sản xuất cảm biến, thiết bị y tế thông minh, và công nghệ thông tin. Sự kết hợp giữa phần cứng và phần mềm trong hệ thống nhận diện té ngã có thể tạo ra những cơ hội kinh doanh và phát triển mới.
* Một lí do khác là đề tài này không chỉ mang lại lợi ích thực tiễn mà còn có giá trị giáo dục và đào tạo. Sinh viên và nhà nghiên cứu tham gia vào dự án sẽ có cơ hội học hỏi và phát triển kỹ năng trong các lĩnh vực như học máy, xử lý tín hiệu, và phát triển hệ thống nhúng, giúp họ chuẩn bị tốt hơn cho sự nghiệp trong tương lai.

1. Mục tiêu của đề tài

* Mục đích khi nhóm chọn đề tài này là vì để hỗ trợ con người trong việc phát hiện kịp thời các sự cố té ngã, đặc biệt ở người cao tuổi. Với sự hỗ trợ của hệ thống phát hiện té ngã người cao tuổi sẽ có thể nhận được sự trợ giúp nhanh chóng nhất nếu có sự cố xảy ra. Ngoài ra hệ thống còn giúp giảm bớt sự phụ thuộc vào việc giám sát thủ công, từ đó giúp tiết kiệm chi phí và nhân lực, đồng thời sẽ giúp cho người thân và gia đình biết an tâm hơn khi biết rằng người cao tuổi được giám sát và hỗ trợ liên tục.

1. Công nghệ áp dụng

* Yolo (You Only Look One) là một mô hình học sâu được sử dụng để phát hiện đối tượng trong hình ảnh và video. Yolo được phát triển với Joseph Redmon và các đồng nghiệp, và đã trở thành một trong những phương pháp phổ biến trong lĩnh vực phát hiện đối tượng, vật thể. Điểm nổi bật của Yolo là khả năng phát hiện nhiều đối tượng có nhãn khác nhau trong thời gian thực.
* Công nghệ Yolo có rất nhiều điểm nổi bật có thể kể đến như là:
* Yolo có khả năng xử lý hình ảnh rất nhanh, điều này rất quan trọng đối với các ứng dụng yêu cầu thời gian thực như giám sát an ninh, xe tự lái, và các thiết bị đeo thông minh.
* Yolo cung cấp hiệu suất cao trong việc phát hiện các đối tượng với độ chính xác tương đối tốt. Nó có thể xác định nhiều đối tượng trong cùng một hình ảnh chỉ với một lần quét.
* Yolo xử lý toàn bộ hình ảnh chỉ với một lần quét, trái ngược với các phương pháp khác yêu cầu quét nhiều lần hoặc chia hình ảnh thành các vùng nhỏ để phân tích. Điều này giúp giảm thiểu thời gian tính toán và tài nguyên xử lý.
* Ngoài Yolo có còn nhiều mô hình khác cũng được sử dụng rộng rãi ví dụ như R-CNN (Region-based Convolutional Neural Networks), HOG (Histogram of Oriented Gradients), Faster R-CNN, tuy nhiên em và nhóm đã chọn mô hình Yolo để sử dụng vì Yolo có nhiều ưu điểm nổi trội hơn các mô hình còn lại như là về tốc độ, tính đơn giản, hiệu suất và tài nguyên tính toán.

1. Mạng tích chập (CNN) :

5.1 Tích chập là gì :

- Mạng tích chập (Convolutional Neural Network), là một mạng lưới tính toán thần kính với cốt lõi là phép toán tích chập, đây là một mạng lưới rất tốt và hiệu quả với các mô hình xử lý ảnh, mạng tích chập thường được dùng trong việt trích xuất ra các đặc trưng của một bức ảnh dựa vào một ma trận trọng số gọi là fillter, fillter giống như các cạnh nối các nút của mạng nơ ron thông thường, là trọng số sẽ thay đổi trong quá trình học (training).

5.2 Công thức và cách hoạt động của một mạng tích chập cơ bản:

- Cách thức mà mạng tích chập hoạt động là dựa trên cơ chế lướt các fillter trên ma trận ảnh, cách trược cơ bản nhất của fillter sẽ giống với cơ chế cửa sổ trượt (sliding window), ảnh khi được chuyển đỗi sẽ ở dạng một ma trận ví dụ như ảnh có chiều cao 7px, chiều rộng 7px và là ảnh màu xám sau khi chuyển đổi sẽ có dạng 7x7x1 với 1 ở đây đại diện cho kênh màu. Giả sử như fillter của chúng ta có số chiều (kernel) là 1x3x3 với 1 là độ sâu, số lượng của fillter còn 3x3 là kích thước của fillter.

Tại mỗi điểm là fillter lướt qua trên ma trận ảnh nó sẽ tính toán như sau :

A = tổng (w\*x)

A là một điểm ảnh đã được tính toán lại khi fillter trược qua, và thường các fillter có số chiều (kernel) cảng lớn thì ảnh cạng bị thu hẹp, việc này cũng phần nào làm mất mát thông tim của ảnh, w chính là các phần tử trong fillter, và x là các phần tử trên ảnh với tướng ứng với vị trí mà fillter trược qua, vd tại điểm bắt đầu thì fillter bắt đầu trược từ vị trí 0,0 của ma trận ảnh khi đó tính toán sẽ như sau :

* Ngoài 2 loại cơ bản trên thì tích chập vẫn có rất nhiều biến thể cho các bài

toán khác nhau.

5.3 Sự hiệu quả của tích chập:

- Tính tối ưu và hiệu quả dường như không phải nói tới, tích chập là một phép gần như tối ưu nhất cho các vấn đề với ảnh, thực tế ngoài mạng tích chập vẫn còn một phép có độ chính xác và hội tụ nhanh hơn là capsule network, nó được cho là tối ưu hơn mạng tích chập với các bài toán nhỏ tuy nhiên với mức độ xử lý của máy mọc bây giờ thì capsule network là không khả thi do khối lượng tính toán quá lớn, một điểm nữa là với phép tích chập ta có thể đáp ứng gần như tất cả các bài toán về sử lý ánh nên hiện tại sự phát triển và đa dạng của mạng tích chập vẫn đứng nhất.

5.4 Phép tích chập được ứng dụng trong mảng nào, mô hình nào:

- Hầu hết các mô hình bên mảng xử lý mảng điều được xây dựng từ mạng tích chập có thể kể đến một vài mô hình nổi tiếng như VGG, Resnet, Yolo, Rcnn…., tích chập gần như là cốt lõi cho mọi mô hình về ảnh và trong hầu hết các mảng như nhận diện (detection), phân loại (classification), phân đoạn (segmentation), nhận diện chữ từ ảnh (OCR), nhận diện điểm (pose detection),…

1. Nhận diện đối tượng

6.1 Nhận diện đối tượng là gì:

- Nhận diện đối tượng hay detection object là một mảng trong xử lý ảnh của học

sâu (deep learning), là từ một ảnh ta không những phải nhận diện là đối tượng có tồn tại trong ảnh không còn phải chỉ ra vị trí của nó , ví dụ bài toán nhận diện con vật thì mô hình phải nhận diện được trong bức ảnh có những loài động vật nào và vị trí của từng loại đó. Đây là một trong những mảng nền tảng và kinh điển đóng vai trò là con cho những phát triển sau này như nhận diện điểm (pose detection) hay nhận diện chữ (detection OCR), và các ứng dụng trong thực tế của nó như xe tự lái, camera an ninh cảnh báo, quản lý nhà trọ chung cư,…

6.2 Lịch sử phát triển:

- Nhận diện đối tượng đã được nghiên cứu từ nhiêu năm trước tuy nhiên nói về sự

bắt đầu thật sự và sự bùng nổ phát triển thì sẽ là 2012 với sự bùng nổ hàng loạt các mô hình với độ chính xác cao như Rcnn, Fast-Rcnn và nhanh như Yolo,Ssd, từ đó đến nay các mô hình này ngày càng phát triển và vững vàng với Yolo là sự ra đời của yolov8-v9 với tốc độ tính toán nhanh và phù hợp với xử lý thời gian thực, Rcnn và các biến thể của nó , hay SSD tuy không phát triển và được chào đón bằng Yolo tuy nhiên cũng có các bước tiến.

6.3 One stage và two stage

- Hầu hết các mô hình nhận diện đối tượng bây giờ điều có thể phân loại dựa vào One stage, two stage. One stage là cách gọi các mô hình nhận diện chỉ có một mạng lưới chính và dùng mạng lưới đó trực tiếp nhận diện và phát hiện đối tượng các mô hình thuộc dạng này thường có độ chính xác không bằng two stage tuy nhiên mô hình thuộc loại này lại nhẹ hơn đáng kể, còn two stage ý chỉ các mô hình có một mạng con bên trong dùng để tính toán ra các vùng đề xuất thường gọi là mạng đề xuất khu vực hay RPN region proposal network, là một mạng nhỏ bên trong mô hình tổng có chức năng đưa các các vùng có khả năng sẽ là đối tượng. Mỗi loại điều có ưu nhược điểm của mình nên việc two hay one stage tốt hơn còn tùy vào nhiều yếu tố.

6.4 Mức độ phát triện của nhận diện đối tượng

Các mô hình ở mảng nhận diện đối tượng này hiện nãy đã vô cùng phát triển với

cực nhiều mô hình mã nguồn mỡ phù hợp tới đa số các đối tượng. Từ những mô hình nhẹ thiết kế phù hợp cho xử lý thời gian thực tới các mô hình ưu tiên độ chính xác đến tuyệt đối tất cả điều có rất nhiều mô hình được chia sẽ mở. Mảng nhận diện đối tượng này không còn ở mức nghiên cứu nữa mà đang ở mức tích cực áp dụng vào thực tế.

1. Mô hình yolo
   1. Kiến trúc Yolo:

* Yolo là một mô hình kinh điển trong các bài toán nhận diện vật thể thời gian thực, tức là đây là một mô hình dạng one stase.
* Nếu với rcnn nó có một mạng lưới con để đề xuất ra những vùng khả thi để bắt đầu tìm kiếm đối tượng thì yolo lại đơn giản hóa đi việc lựa chọn các vùng bằng cách chia hình ảnh lại thành các ô và xác định xem trong các ô đó có vật thể hay không sau đó mới tính toán vị trí của nó. Thực tế khi tính toán trong code thì bản chất của việc chia ảnh sẽ là dùng hàng loạt các lớp tích chập hay một mô hình tích chập nào đó đưa ảnh về một ma trận SxS với s là số khung muốn chia ban đầu thông số này cũng sẽ ảnh hưởng tới việc nhận diện vật thể việc hai vật thể có cùng một tâm sẽ không nhận diễn được nhưng đây chỉ là điểm yếu của các phiên bản đầu của Yolo về sau khi áp dụng các khái niệm như anchor thì vấn đề này đã được giải quyêt.
* Trước khi đi tiếp vào các thuật toán của Yolo ta sẽ nói về Cnn và Extra layers (fully connected layers), như có trình bài ở đầu mục phần hai Cnn là một mạng tích chập thường dùng để trích xuất đặc trưng ảnh còn Extra layers nó sẽ từ feature map có được từ Cnn để tính toán ra đầu ra (output), là một tổ hợp bao gồm các thông số như xywh với xy là tọa độ tâm của đối tượng, wh là chiều dài rộng của đối tượng, sát xuất một điểm trên feature map đó (cell) thuộc nhãn nào và tại cell này có tồn tại đối tượng không.
* Output sẽ có hình dạng SxSx(5\*B+C) với B là số lượng khung (box) và C là số lượng nhãn. Ví dụ với feature map 7x7, phân biệt chó mèo, có 2 khung cố định ban đầu , thì output sẽ là 7x7x12 tương ứng với mỗi điểm trên feature map (cell) ta có 12 phần tử, 5 phần tử đầu tiên quyết định xem cell này có chứa chó hay mèo không và xywh của box đầu tiên, và 5 phần tử tiếp theo quyết định cell này có chứa chó hay mèo không và xywh của box thứ hai, 2 phần tử cuối là xác xuất chứa chó hay mèo của cell này.
* Vậy tóm lại và bỏ đi các lý thuyết màu mè thì mô hình Yolo sẽ từ một ảnh ban đầu sau đó đi qua một mạng Cnn để thu nhỏ về một feature map có kích thước bằng với số khung muốn chia trên ảnh ban đầu, sau đó lại duỗi phẳng feature map ra thành một vector và cho vecto này đi qua các lớp fully connected sau cho output cuối cùng về mạng một ma trận SxSx(5\*B+C) như đã nói ở trên. Thông qua kiến trúc ta dễ dàng có thể thấy được lý do vì sau Yolo lại nhanh hơn nhiều so với các mô hình khác như Rcnn, Faster-Rcnn, cấu trúc của nó không quá phức tạp và chỉ có một mạng lưới thông từ đầu đển cuối không cần đến các mạng đề xuất vùng nó trực tiếp tính toán ra tọa độ và xác suất xuất hiện vật thể chỉ trong một lần tính toán.
  1. Giới thiệu mô hình yolov1:
* Yolov1 là mô hình cổ nhất của Yolo, tuy có rất nhiều khuyết điểm tuy nhiên không thể phủ nhận nó là một mô hình làm nền tản và cơ sở cho các mô hình Yolo sau này, mô hình Yolov1 là mô hình đầu tiên nên còn rất nhiều thiếu sót như không thể nhận diện được nhiều đối tượng trong một cell (một điểm trong output)
  1. Hàm mất mát (Loss function)

Cốt lõi trong mọi bài toán deeplearning không thể thiếu loss function, một hàm đo lường sự sai lệch giữa output và True\_output (nhãn của một điểm dữ liệu mà ta đang train). Và tùy vào các output mà hàm loss có thể được chia ra và tính toán hết sức phức tạp, với kiến trúc Yolo thì output của chúng ta lần lược là xywh của đối tượng, confident là độ tin cậy của box đó tức là tại box(j) này trong cell(i) với các vd trên thì j=2(số lượng box trong mỗi cell) còn i=7(các cell) này thì đối tượng có xuất hiện không, cuối cùng là xác suất của các nhãn tại cell này.

7.4 Độ đo IOU

- Hiểu đơn giản thì độ đo Iou giống như đo độ khớp giữa 2 mặt phẳn với 2 mặt phẳng trong Yolo sẽ là boxes mô hình dự đoán ra và boxes nhãn.

7.5 NMS (non-max subpression) :

- Đây là một thuật toán ra đời để giải quyết vấn đề một đối tượng bị trùng nhiều boxes, trường hợp này xảy ra khi đối tượng quá to so với một cell đó và iou của nó khi tính toán bị lệch qua các cell khác làm nhiều cell bị kéo vào và dự đoán cho một đối tượng

7.6 Các phiên bản của Yolo phía sau :

- Yolo hiện nay đã qua rất nhiều phiên bản, vậy ta sẽ kể qua những thay đổi đáng kể trong các phiên bản :

- Đầu tiên là Yolov2 với sự kết hợp của anchor box và đổi output thành 13 cell, và thay đổi một mạng cnn khác (Darknet 19) và đa dạng hóa kích thước đầu vào. Yolov3 với sự biến đổi vượt bật trong việc có thể nhận diện nhiều class trong một cell nhờ vào cung cấp score mỗi bounding boxes sử dụng logistic regression và sử dụng logistic classifiers cho mọi class thay vì softmax giải quyết vấn đề multill class trong một cell, cùng một vài thay đổi khác như backbone, FPN, và hàm loss cũng có thay đổi khá nhiều so với yolov1.

- Và với các phiên bản sau thì cũng xoay quanh các vấn đề như thay đổi backbone, hàm loss, tăng giảm anchor thêm vào các mạng sau backbone.

7.7 Vì sau lại chọn Yolov8:

- Đây là mô hình gần mới nhất của Yolo với mới nhất là v9, Yolo 8 vô cùng dễ sử dụng và cũng là phiên bản gần như tối ưu nhất, thực ra so với v5 nó cùng không có nhiều cải tiến tuy nhiên thì dù ít thì vẫn tốt hơn nên dại gì không chọn, và lý do không chọn Yolo9 là vì nó khó sài hơn mà thông số thì không có sự khác biệt đáng kể, đặc biệt là đề tài quá đơn giản nên có dùng yolov5 thì vẫn ổn. và không có quá nhiều khác biệt

1. Chuẩn bị và tiền xử lý dữ liệu :
   1. Chuẩn bị dữ liệu.

* Bộ dữ liệu phát hiện té ngã và không té ngã được lấy trên trang web (<http://fenix.ur.edu.pl/~mkepski/ds/uf.html>)
* Tiền xử lý dữ liệu.

+ Sạch dữ liệu gồm.

+ Loại bỏ những dữ liệu bị nhiễu và độ tương phản thấp.

+ Đánh label cho từng điểm dữ liệu trong bộ dữ liệu.

+ Sử dung YOLO8 model để cắt và lấy hành động của đối tượng

+ Lấy số đo của boxes

+ Lưu những hình ảnh và các labels vào các file để làm tệp train và test

* Các bước xử lý dữ liệu và đánh labels

+ Chia và xác định label

Té ngã: 1

Không té ngã: 0

* Xử lý với label té ngã:

+ Không lấy ảnh bấm **enter,**

+ Đánh label = 1 (té ngã) và lấy box nhập vào file labels

+ Ảnh được lưu vào file

* Xử lý với label không té ngã:

+ Đánh label = 0 (không té ngã) và lấy box nhập vào file labels

+ Ảnh được lưu vào file

1. Thực nghiệm

9.1 Thực nghiệm và kết quả trên bộ data của nhóm

- Kết quả mong muốn:

epoch = 50

loss 0.2 - 0.1

accuracy >= 0.9

* Kết quả train test:

+ Model kết thúc ở epoch = 50.

+ Loss test: loss của model ổn định, đạt mức loss = 0.2034.

+ Nhận xét loss: model đạt đến mức loss trong khoản mong muốn.

+ Accuracy test: đạt mức 0.906 ở điểm epoch = 50.

+ Nhận xét accuracy: model đã đạt được và vượt qua giá trị mong muốn.

* Lí do pretrain 1 model

+ Thứ nhất là để tiết kiệm thời gian và tài nguyên tính toán, để huấn luyện một mô hình từ đầu đòi hỏi rất nhiều thời gian và tài nguyên tính toán, đặc biệt là đối với các mô hình phức tạp như các mạng nơ-ron sâu. Pretrained model (mô hình được huấn luyện trước) đã được huấn luyện trên một tập dữ liệu lớn, do đó có thể giảm đáng kể thời gian huấn luyện.

+ Thứ hai là có thể cải thiện hiệu suất và độ chính xác , có khả năng khái quát tốt hơn vì mô hình đã được huấn luyện trước đã được huấn luyện qua một tập dữ liệu đa dạng

+ Thứ ba là khả năng tùy chỉnh và tinh chỉnh, pretrain cung cấp cho chúng ta một nền tảng mạnh mẽ để tinh chỉnh cho các nhiệm vụ cụ thể

+ Cuối cùng là dễ dàng triển khai và áp dụng vì các model này đã được tối ưu hóa để triển khai trên các nền tảng khác nhau như TensorFlow, PyTorch, và các dịch vụ đám mây. Việc này giúp cho việc tích hợp và ứng dụng mô hình vào các ứng dụng thực tế sẽ ngày càng dễ dàng hơn.

9.2 Thực nghiệm và kết quả trên bộ data bất kì.

- Kết quả mong muốn: model đạt kết quả tốt và chạy mượt mà khi ở điều kiện hình ảnh thực tế.

- Kết quả train test:

Model cho thấy sự tích cực và tập trung đúng vào đối tượng cần ở đây là người bị đột quỵ và té ngã.

1. Kết luận và hướng phát triển

10.1 Kết luận

- Sau quá trình phát triển và đánh giá, Model AI đã thể hiện khả năng khá tốt trong việc giải quyết bài toán được đề ra. Kết quả đạt được đáp ứng đúng mục tiêu và đáp ứng yêu cầu của đồ án. Cụ thể:

+ Độ chính xác: Model đã đạt độ chính xác cao

+ Hiệu suất: Thời gian thực thi và tốc độ dự đoán của model đều ở mức tốt.

+ Áp dụng thực tế: Model đã được triển khai thành công trong môi trường thực tế đem lại giá trị cho người dùng.

10.2 Hướng phát triển

- Những phương pháp thay đổi sâu hơn trong cấu hình của mô hình có thể làm tăng độ chính xác của model. Cụ thể:

+ Phải xử lý tốt ở bước tiền xử lý dữ liệu

+ Tinh chỉnh siêu tham số (Hyperparameter Tuning

+ Kiểm tra và loại bỏ overfitting

+ Tối ưu hóa thuật toán huấn luyện

+ Sử dụng kiến trúc mạng phù hợpHiện tại năm 2024 đã có mô hình mới hơn là YOLOv9 có những điểm vượt trội hơn so với YOLOv8 ở lĩnh vực thị giác máy tính. Cụ thể:

+ Kiến trúc cải tiến: YOLOv9 tích hợp các cải tiến như Programmable Gradient Information (PGI) và Generalized Efficient Layer Aggregation Network (GELAN). PGI giúp tránh mất dữ liệu trong quá trình cập nhật gradient, trong khi GELAN tối ưu hóa các mô hình nhẹ thông qua lập kế hoạch gradient.

+ Độ chính xác và hiệu suất: tăng khả năng học của mô hình mà còn đảm bảo giữ lại thông tin quan trọng trong quá trình nhận biết.

+ Những kết quả khi ảnh bị che, mờ cũng có thể cải thiện khi tăng cường dữ liệu