XÂY DỰNG HỆ THỐNG NHẬN DIỆN DÁNG ĐIỆU TRONG GIÁM SÁT SỨC KHỎE VẬN ĐỘNG THỂ THAO

Nguyễn Hữu Phát*, Nguyễn Thị Ngọc†, Phạm Ngọc Thiện†
*Bộ môn Mạch và Xử lý tín hiệu, Viện Điện tử viễn thông, Đại học Bách Khoa Hà Nội
† Viện Điện tử viễn thông, Đại học Bách Khoa Hà Nội

Tóm tắt: Bài báo đề xuất một hệ thống nhận diện dáng điệu, theo dõi hành vi con người áp dụng cho giám sát y tế, sức khỏe trong vận động thể dục, thể thao. Phương pháp mà chúng tôi sử dụng là ước tính dáng điệu của con người bằng thuật toán Openpose có backbone (xương sống) là mạng MobilenetV2, theo dõi và phân loại các dáng điệu đó bằng mạng LSTM. 4 dáng điệu được đề xuất huấn luyến bao gồm đá chân, đánh đấm, nhảy và thể dục khuyu chân. Đầu ra của hệ thống là trích xuất ra bộ khung xương của con người và nhãn tương ứng cho các dáng điệu. Kết quả hệ thống có độ chính xác khoảng 91% và có khả năng ứng dụng giám sát y tế, thể dục thể thao hoặc điều khiển thiết bị điện tử trong nhà thông minh.

Từ khóa: Openpose, VGG19, Mobilenet, Mobilenet-thin, LSTM, pose estimate, extract skeleton, pose human.

I. ĐẶT VẤN ĐỀ

Nền công nghiệp AI sử dụng máy móc thay thế sức lao động của con người được nghiên cứu và phát triển ra nhiều máy móc hữu ích, áp dụng rộng rãi trong thực tế như điều khiển nhà thông minh bằng giọng nói, chấm công bằng nhận diện khuôn mặt, xe tự lái,… Một ứng dụng để điều khiển máy móc mà chúng tôi đang nghiên cứu hiện tại chưa có phương pháp nào tối ưu tuyệt đối để vận dụng rộng rãi trong thực tế đó là điều khiển thiết bị điện tử trong nhà bằng cử chỉ, hành động, dáng điệu của con người, đồng thời giám sát, giám sát hành vi của con người đưa ra các cảnh báo nguy hiểm trong y tế, theo dõi các động tác trong thể dục thể thao. Hệ thống nhận diện dáng điệu mà chúng tôi xây dựng mang đến lợi ích tuyệt vời như vậy. Lấy ví dụ một số ví dụ điển hình như trong y tế theo dõi người già bị ngã, bị đột quỵ, trẻ nhỏ ngã hoặc gặp nguy hiểm, đưa ra các cảnh báo cho người thân hoặc bệnh viện; trong hình sự nhận biết được hành vi trộm cắp, đe dọa; trong thể thao nhận biết được vận động viên sai luật,…

Thực tế đã có một số nghiên cứu nhận diện cử chỉ, hành động đã được áp dụng cho các ứng dụng trên nhưng đem lại kết quả không cao hoặc chi phí phần cứng lớn. Hệ thống nhận dạng dáng điệu được coi là nâng cao của nhận dạng cử chỉ, hành động. Đầu vào của chúng tôi yêu cầu cần dữ liệu là các khung hình liên tiếp, chưa toàn bộ cơ thể con người. Qua thử nghiệm, hệ thống phát hiện khá chính xác con người và đưa ra nhãn chính xác.

Sự khác biệt trong hệ thống của chúng tôi bắt nguồn từ dữ liệu vào là các khung hình cắt từ các video và chứa toàn bộ cơ thể con người. Hệ thống này muốn đạt được kết quả tốt nhất thì dữ liệu cần huấn luyện là vô cùng lớn. Hãy hình dung hệ thống của chúng tôi gồm 2 phần lớn. Đầu tiên là phát hiện và trích xuất bộ xương của con người, chúng tôi sử dụng thuật toán Openpose, chúng tôi thay thế backbone nguyên thủy là mạng VGG19 bằng mạng Mobilenet giảm khối lượng tính toán, giảm số lượng tham số đi 40 lần. Tiếp theo, hệ thống sẽ sử dụng mạng LSTM để học các đặc trưng của bộ xương được trích xuất đồng thời phân loại nhã dáng điệu, đây là mạng bộ nhớ dài ngắn sử dụng dữ liệu video để huấn luyện, nắm bắt được ưu điểm này nên chúng tôi quyết định áp dụng vào hệ thống nhận diện dáng điệu.

II. CÁC NGHIÊN CỨU LIÊN QUAN

Hiện nay, chủ đề về nhận diện hành động của con người không còn xa lạ đối với những ai có đam mê nghiên cứu về lĩnh vực thị giác máy tính. Có rất nhiều phương pháp được đề xuất nhưng mới chỉ dừng lại ở mức nghiên cứu hoặc mô phỏng trên máy tính.

Trước đây, phương pháp đơn giản để phát hiện hành vi của con người dựa trên phương pháp trừ nền để tìm kiếm đối tượng chuyển động [1]–[3], do đó dữ liệu huấn luyện cần chất lượng tốt, ánh sang tốt, tiền xử lý mất nhiều công đoạn và phải áp dụng các phương pháp tối ưu cho việc lọc,, trừ nền,…Các phương pháp, mô hình mạng Deep Learning được ra đời như Yolo [4] trong chuyển động video, phát hiện con người và sử dụng hộp giới hạn để theo dõi và nhận diện hành động hay Mobilenet V2 kết hợp SSD [5] với ảnh tĩnh đạt độ chính xác trên 90% khi chạy trên cấu hình yếu như điện thoại. Những phương pháp này mặc dù đã cải thiện tốt hơn tuy nhiên tùy thuộc vào dữ liệu đào tạo mà hiệu suất có thể kém và model nặng, ảnh hưởng trực tiếp đến hiệu năng và khả năng áp dụng vào thực tế cảu bài toán.

Các phương pháp sử dụng camera RGB-D Kinect hoặc sử dụng sensor đạt được độ chính xác khá cao nhưng tốn kém về mặt phần cứng [6][7]. Một số thuật toán xử lý ảnh được áp dụng phổ biến cho bài toán ước tính hành vi của con người như SimplePose [8], AlphaPose [9], OpenPose [10] sử dụng các bộ dữ liệu lớn như COCO, MPII,… Các thuật toán này được nghiên cứu và áp dụng

Tác giả liên hệ: Nguyễn Hữu Phát
Email: phat.nguyenhuu@hust.edu.vn
Đến tòa soạn: 6/2021, chỉnh sửa: 7/2021, chấp nhận đăng: 7/2021

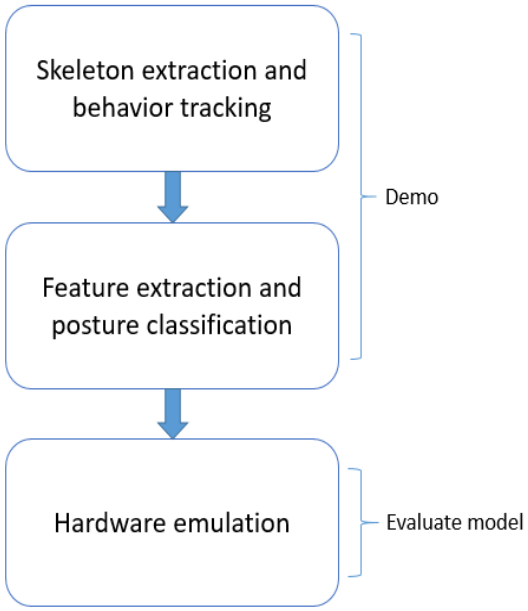
để trích xuất ra bộ xương từ việc tìm ra các điểm trên cơ thể đến ghép nối một cách phù hợp, sau đó ước tính tư thế con người, thậm chí có thể thực hiện trong thời gian thực. Để cải thiện tốc độ cũng như độ chính xác cho hệ thống phát hiện dáng điệu con người, một số nghiên cứu tiên phong [11] [12] đã cho ra đời phương pháp tìm ra các keypoint tạo ra bộ xương, hệ thống sẽ học các đặc trưng của bộ xương đó để nhận diện các tư thế, hành vi con người. Ban đầu, các keypoint được tìm kiếm theo phương pháp top-down [9] hệ thống dò tìm đối tượng sau đó trích xuất bộ xương và ước tính tư thế, theo thực nghiệm phương pháp này sẽ có phát hiện dư thừa do nhiều tư thế được phát hiện cho một người vì có nhiều hộp giới hạn lỗi bao quanh, phương pháp này có thời gian chạy tương ứng với số người xuất hiện trong khung hình, do đó phương pháp sẽ hiệu quả nếu chỉ có ít người. Trong khi [10] phương pháp down-top trong Openpose lại tách thời gian chạy hệ thống với số người xuất hiện trong khung hình, điều này rất giúp hệ thống vừa chạy trên thời gian thực, vừa trích xuất được nhiều bộ xương. Đây chính là điểm mấu chốt dẫn đến quyết định lựa chọn Openpose cho ước tính tư thế con người của chúng tôi. Tuy nhiên, ban đầu tác giả [10] sử dụng backbone của Openpose là mạng VGG19, mạng CNN này được đánh giá là một mạng học sâu, bao gồm một chuỗi các tầng, một tầng tích chập (với phần đệm để duy trì độ phân giải), một tầng phi tuyến như ReLU, một tầng gộp như tầng gộp cực đại. Một khối VGG gồm một chuỗi các tầng tích chập, tiếp nối bởi một tầng gộp cực đại để giảm chiều không gian. Trong bài báo gốc của VGG [13], tác giả sử dụng tích chập với các hạt nhân 3×33×3 và tầng gộp cực đại 2×22×2 với sải bước bằng 22 (giảm một nửa độ phân giải sau mỗi khối). Số lượng tham số mà mạng VGG19 được tính toán khoảng 138 triệu tham số. Do hệ thống của chúng tôi cần tối ưu về hiệu suất nên chúng tôi thay backbone của Openpose bằng mạng Mobilenet V2 có tham số giảm hơn 40 lần so với VGG19.

Sau khi sử dụng Openpose để trích xuất ra bộ xương, chúng tôi sẽ tích hợp LSTM vào hệ thống nhằm trích xuất ra các đặc trưng và phân loại nhãn dáng điệu. Đưa LSTM vào hệ thống, chúng tôi sẽ cải thiện được sự sai khác giữa các hành động tĩnh và động có dáng điệu dễ gây nhầm lẫn. Chi tiết về hệ thống chúng tôi sẽ trình bày ở phần III và đánh giá kết quả ở phần IV, bao gồm cả kết quả đánh giá trên phần cứng thật mà chúng tôi mô phỏng trên máy tính nhúng Jetson Nano Kit.

III. GIẢI PHÁP THỰC HIỆN

A. *Tổng quan về hệ thống*

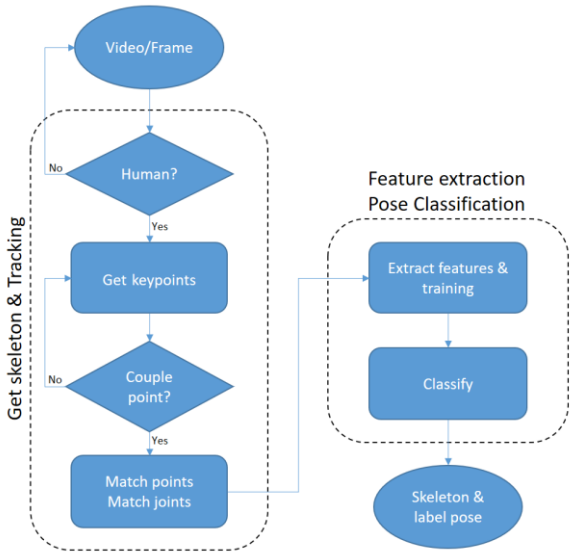
Hình 1 là mô hình tổng quan các khối chính của hệ thống nhận diện. Đầu tiên, dữ liệu đầu vào đi qua mô hình Openpose, thuật toán giúp phát hiện con người, tiến hành phân tích và trích xuất các điểm lưu lại trên cơ thể, ghép nối tạo thành các khớp và tạo bộ xương hoàn chỉnh, đồng thời thuật toán cũng theo dõi sự thay đổi qua các khung hình để việc trích xuất chuẩn xác hơn. Các giá trị bộ xương đã trích xuất được mạng LSTM học các đặc trưng, huấn luyện và tiến hành phân loại. Sau khi huấn luyện và thử nghiệm với bộ dữ liệu test, hệ thống được đánh giá là có khả năng chạy trên phần cứng cấu hình thấp hơn. Vì vậy kết quả huấn luyện được đưa vào máy tính nhúng Jetson Nano để thử nghiệm và cho ra kết quả khá khả quan, đánh giá chi tiết ở phần IV.



Hình 1. Mô hình tổng quan hệ thống thực hiện.

Mọi hệ thống được nghiên cứu và xây dựng nếu chỉ dừng ở mô phỏng sẽ là vô tác dụng, do đó chúng tôi sẽ đánh giá kết quả mô phỏng và thực thi thử nghiệm trên phần cứng, từ đây chúng tôi khẳng định hệ thống này không chỉ dừng ở bước nghiên cứu mà có thể đưa vào mô phỏng thực tế như điều khiển thiết bị điện tử thật thông qua giám sát camera thường.

B. *Chi tiết hệ thống*



Hình 2. Sơ đồ hoạt động chi tiết của mô hình

Hình 2 khái quát chi tiết cách thức huấn luyện mô hình nhận diện dáng điệu của hệ thống.

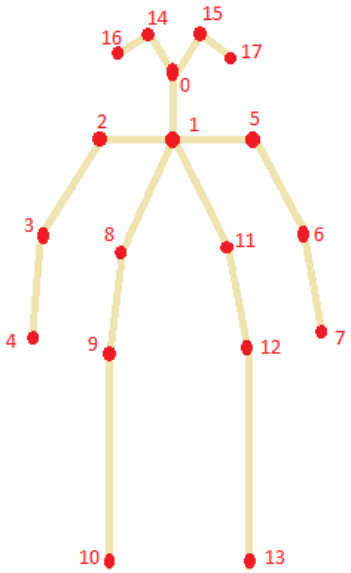
1. *Openpose và Mobilenet-thin*

Khối thứ nhất làm nhiệm vụ trích xuất bộ xương đồng thời theo dõi đối tượng để nối các khớp xương từ các điểm chính trên cơ thể đã được tìm thấy, xem Hình 3. Thuật toán Openpose thực hiện toàn bộ công đoạn này.

Hình 4 [14] mô tả kiến trúc của thuật toán Openpose. Các khung hình lần lượt được đưa vào, Openpose,

backbone là mạng VGG-19 phát hiện con người, các nhánh Stage 1 – Stage t để đào tạo model nhận diện các điểm của bộ xương. Sau đó, 38 Part Afinity Fields (PAF) cho biết mức độ liên kết các khớp và dự đoán liên tục.

Từ việc đánh giá kích thước mạng, tham số và độ chính xác được liệt kê ở Bảng I. Hệ thống của chúng tôi quyết định sử dụng backbone Openpose là mạng Mobilenet.

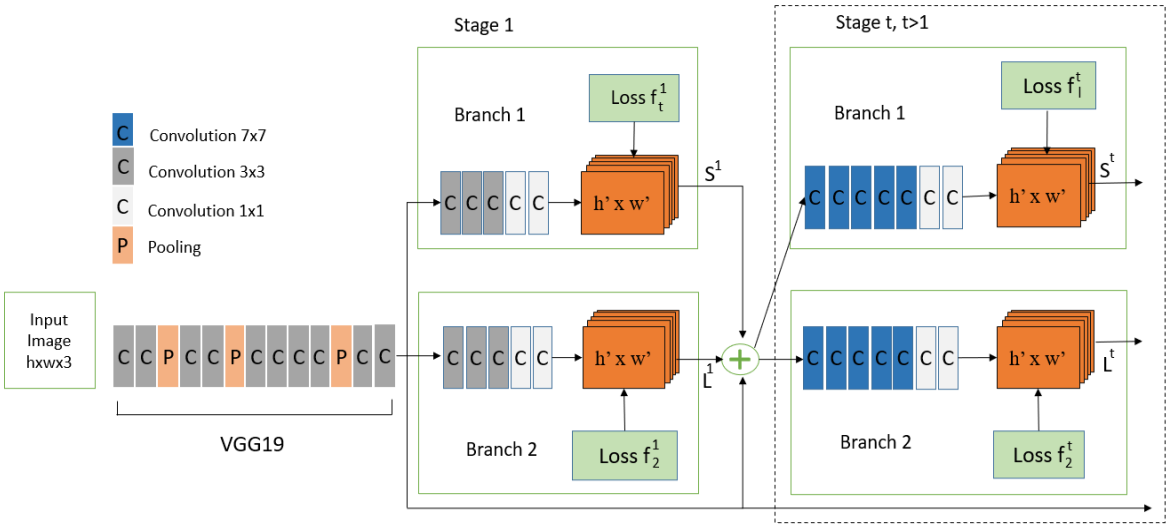


Hình 3. 18 điểm trên cơ thể và bộ xương tương ứng mà Openpose cần trích xuất

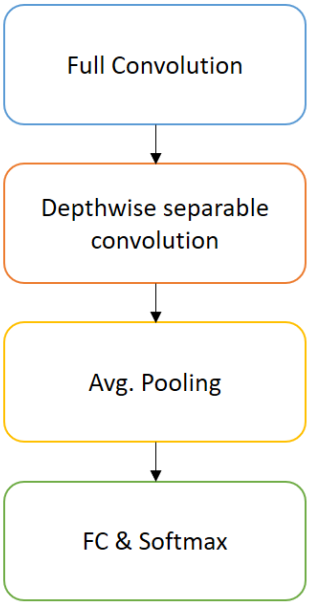
Mobilenet sử dụng lớp tích chập Depthwise Separable Convolution, xem Hình 5 [15]. Ý tưởng của Depthwise Separable Convolution là chia phép convolution làm 2 phần: Depthwise convolution & Pointwise convolution. Depthwise convolution là một loại tích chập áp dụng một bộ lọc tích chập duy nhất cho mỗi kênh đầu vào. Trong phép tích chập 2D thông thường được thực hiện trên nhiều kênh đầu vào, bộ lọc cũng sâu như đầu vào và cho phép chúng ta tự do trộn các kênh để tạo ra từng phần tử trong đầu ra. Ngược lại, sự biến đổi theo chiều sâu giữ cho mỗi kênh riêng biệt. Pointwise Convolution là một kiểu tích chập sử dụng một nhân 1x1: một nhận lặp lại qua từng điểm đơn lẻ. Kernel này có độ sâu bằng nhiều kênh mà hình ảnh đầu vào có. MobileNet dùng rất nhiều lớp Depthwise Separable Convolution để giảm số lượng parameter đi nhiều lần khoảng 9 lần.

Tuy nhiên sau khi khảo sát thấy mạng Mobilenet-thin được nghiên cứu dựa trên mạng Mobilenet cho tốc độ tính toán nhanh hơn gấp đôi và độ chính xác gần bằng nhau, xem Bảng 4 [15], chúng tôi quyết định sử dụng mạng Mobilenet-thin cho backbone của Openpose. Hình 6 là cấu trúc mạng của Mobilenet-thin. Drop-Activation là hàm kích hoạt phi tuyến, cụ thể là hàm ReLU6. ReLU6 cung cấp hệ số mở rộng, đây là một hàm bẻ gãy sự tuyến tính của các Neuron, nó mạnh mẽ khi được sử dụng với tính toán có độ chính xác thấp [16]. Batch Normalization là lớp chuẩn hóa hàng loạt để giảm thiểu Overfitting [17]. Lớp chập theo chiều sâu giúp lọc đầu vào cuối cùng là lớp chập 1x1 giúp số lượng kênh nhỏ hơn, lớp này còn được gọi là lớp nút cổ chai – Projection Layer. Lớp này giảm dữ liệu chảy qua mạng. Mạng Mobilenet-thin cắt giảm đi 1/3 số lớp của Mobilenet, mô tả ở Bảng Bảng III [15], đầu vào là ảnh có kích thước 32x32x3, bộ dữ liệu sử dụng là CIFAR-10.

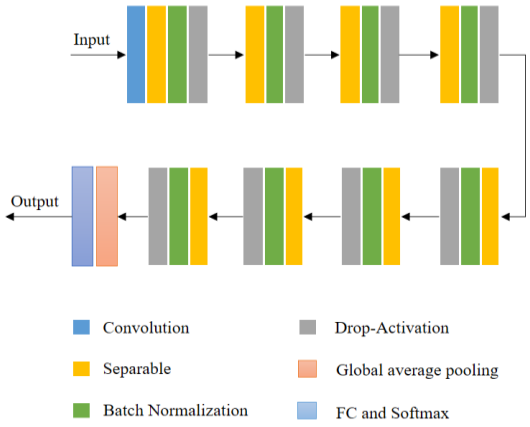
Với kết quả khảo sát được ở Bảng II, hệ thống của chúng tôi sẽ sử dụng Pretrained model (mô hình mạng đã được huấn luyện dựa trên bộ dữ liệu lớn và được đánh giá công khai) của mạng Mobilenet-thin để thực hiện huấn luyện hệ thống của chúng tôi. Điểm tương đồng về dữ liệu đào tạo được mô tả ở Hình 7 và Hình 8. Theo [15] tác giả sử dụng mạng Mobilenet-thin huấn luyện bộ dữ liệu CIFAR-10, bài toán cụ thể là nhận diện đối tượng (object detection). Ở hệ thống này, chúng tôi sẽ áp dụng bài toán cơ sở object detection để thực hiện bài toán phức tạp hơn, cụ thể đối tượng mạng cần thực hiện phát hiện và nhận diện là con người, đồng thời tìm kiếm các điểm trên cơ thể để trích xuất bộ xương.



Hình 4. Kiến trúc Openpose



Hình 5. Mô hình mạng cơ sở của Mobilenet



Hình 6. Mô tả cấu trúc mạng Mobilenet-thin

Bảng I. Đánh giá tham số và độ chính xác của một số mạng CNN trên cùng tập dữ liệu ImageNet từ Keras

Mạng	Kích thước	Độ chính xác	Tham số
Xception	88MB	0.945	22,910,480
VGG16	528MB	0.9001	138,357,544
VGG19	549MB	0.900	143,667,240
ResNet50	99MB	0.921	25,636,712
InceptionV3	92MB	0.937	23,851,784
MobileNet	16MB	0.895	4,253,864

Bảng II. So sánh mạng Mobilenet và Mobilenet-thin huấn luyện trên tập dữ liệu CIFAR-10 với 200 Epochs [15]

Mạng	Kích thước	Độ chính xác	Tham số	Thời gian tính toán trên mỗi epoch
Mobilenet	39.1MB	84.3%	3,239,114	31s
Mobilenet-thin	9.9MB	85.61%	814,826	14s

Bảng III. Kiến trúc mạng Mobilenet-thin

Layer/Stride	Ouput Shape	Parameter
Input layer	32, 32, 3	0
Conv2d/2s	16,16,32	864
Separable conv2d/s1	16, 16, 32	1312
Separable conv2d/s2	8, 8, 64	2336
Separable conv2d/s1	8, 8, 128	8768
Separable conv2d/s2	4, 4, 128	17536
Separable conv2d/s1	4, 4, 256	33920
Separable conv2d/s2	2, 2, 256	67840
Separable conv2d/s2	1, 1, 512	133376
Separable conv2d/s1	1, 1, 1024	528896
Global average pool/s1	1, 1, 1024	0
FC & Softmax/s1	1, 1, 10	10250

Hình 7. Dữ liệu CIFAR-10

Hình 8. Dữ liệu của chúng tôi thử nghiệm

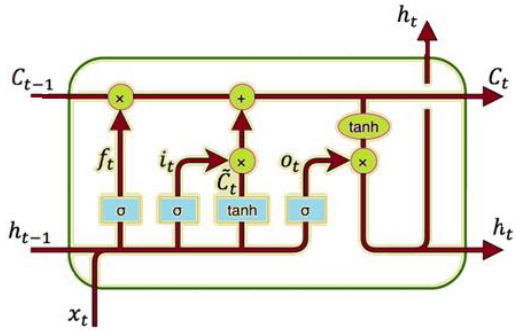
Các bộ xương mà Openpose trích xuất ra một ma trận chứa vị trí 18 điểm trên bộ xương, có giá trị trong khoảng từ 0 - 1 và file nhãn tương ứng với dáng điệu của bộ xương.

2. LSTM

RNN (Recurrent neural network) là mô hình Deap Learning, có thể xử lý thông tin dạng chuỗi (sequence/ time-series). Trong bài toán dự đoán hành động đầu vào là video, RNN có thể mang thông tin của frame (ảnh) từ state trước tới các state sau, rồi ở state cuối là sự kết hợp của tất cả các ảnh để dự đoán hành động trong video.

Mạng bộ nhớ dài-ngắn (Long Short Term Memory networks), viết tắt LSTM - là một dạng đặc biệt của RNN, nó có khả năng học được các phụ thuộc xa. LSTM được thiết kế để tránh được vấn đề phụ thuộc xa (long-term dependency). Việc nhớ thông tin trong suốt thời gian dài là đặc tính mặc định của chúng, chứ ta không cần phải huấn luyện nó để có thể nhớ được. Tức là ngay nội tại của nó đã có thể ghi nhớ được mà không cần bất kì can thiệp nào.

Mọi mạng hồi quy đều có dạng là một chuỗi các mô-đun lặp đi lặp lại của mạng nơ-ron. Với mạng RNN chuẩn, các mô-dun này có cấu trúc rất đơn giản, thường là một tầng *tanh*.



Hình 9. Cấu trúc bên trong một state của LSTM

LSTM cũng có kiến trúc dạng chuỗi như vậy, nhưng các mô-đun trong nó có cấu trúc khác với mạng RNN chuẩn. Thay vì chỉ có một tầng mạng nơ-ron, chúng có tới 4 tầng tương tác với nhau một cách rất đặc biệt, mô tả trong Hình 9.

Các đặc trưng của bộ xương sẽ đưa vào *x*, các trạng thái tiếp theo [18][19]:

- Input: c_{t-1}, h_{t-1}, x_t. Trong đó x_t là input ở state thứ t của model. c_{t-1}, h_{t-1} là output của layer trước. **h** là kết quả hàm tanh.
- Output: c_t, h_t ta gọi c là cell state, h là hidden state.
- f_t, i_t , o_t tương ứng với **f**orget gate, **i**nput gate và **o**utput gate.

Điểm mới của LSTM so với mạng RNN nói chung là băng chuyền từ c_{t-1} đến c_t giúp thông tin nào quan trọng và cần dùng ở sau sẽ được gửi vào và sử dụng khi cần thiết nên LSTM có thể mang thông tin đi từ xa (nhớ dài hạn), kết hợp với các đặc biệt kế thừa từ RNN là bộ nhớ nhắn hạn, đây là lý do tại sao LSTM là mạng dài – ngắn hạn. Đặc điểm này của LSTM thích hợp với dữ liệu đầu vào của chúng tôi là chuỗi khung hình.

Ở khối thứ hai, sau khi mạng học hầu hết các đặc trưng sau từng state thì dữ liệu được đưa vào hàm kích hoạt Softmax nằm ở layer cuối cùng của mạng LSTM để phân lớp nhãn dáng điệu. Hàm softmax sẽ tính khả năng xuất hiện của một class trong tổng số tất cả các class có thể xuất hiện. Sau đó, xác suất này sẽ được sử dụng để xác định class mục tiêu cho các input. Xác suất sẽ luôn nằm trong khoảng [0:1]; tổng tất cả các xác suất bằng 1 [20].

IV. KẾT QUẢ ĐẠT ĐƯỢC

A. Dữ liệu đào tạo

Dữ liệu được thu thập dựa trên thực tế, 30% dữ liệu được lấy từ nguồn Youtube, 70% dữ liệu chúng tôi tự quay. Mục đích để hệ thống hoạt động chính xác nhất khi thử nghiệm thực tế.

Bộ dữ liệu gồm 4 dáng điệu: kick (đá chân lên cao), punch (đấm bốc), jumping jack (nhảy kết hợp đập tay) và exercise foot (thể dục dãn cơ chân), mô tả Hình 10, Hình 11.

Dữ liệu video được cắt thành các frame, gán nhãn và phân loại theo mục. Tổng số khung hình chúng tôi có khoảng hơn 8200 khung hình cho tập train và 900 khung hình cho tập test, Bảng IV mô tả chi tiết số lượng khung hình của từng dáng điệu.

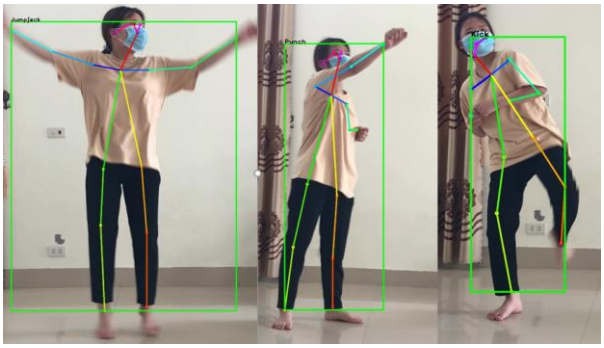
Bảng IV. Số lượng dữ liệu thu thập

Dữ liệu	Huấn luyện	Thử nghiệm
Exercise Foot	1800	180
Jumping Jack	2900	290
Kick	2300	250
Punch	1200	180

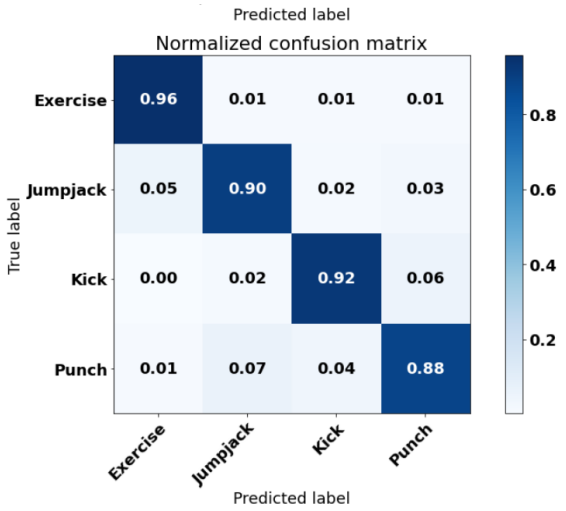
Hình 10. Minh họa bộ dữ liệu đào tạo

Hình 11. Minh họa bộ dữ liệu thử nghiệm

B. Kết quả huấn luyện mô hình



Hình 12. Kết quả phân loại dáng điệu trên tập test



Hình 13. Ma trận nhầm lẫn giữa các dáng điệu

Hình 13 mô tả ma trận nhầm lẫn giữa các dáng điệu mà chúng tôi đạt được sau khi huấn luyện xong mô hình, chúng tôi nhận thấy rằng dáng điệu Punch có sự nhầm lẫn cao nhất với các dáng còn lại do tư thế thực hiện hành động của dáng điệu này chủ yếu tay hoạt động và ở tư thế đứng. Hình 12 là một vài minh họa chúng tôi thực hiện test trên bộ dữ liệu test. Kết quả chúng tôi đạt được khá khả quan, cụ thể độ chính xác khoảng 91.2%, tốc độ xử lý khung hình trên giây khoảng 13.5fps. Các trường hợp bộ xương không chính xác như đối tượng bị thiếu bộ phận, ví dụ như cổ, xương đùi.

Chúng tôi đưa ra những so sánh nhất định với một số nghiên cứu liên quan, trong đó có đánh giá về độ chính xác, hiệu năng hoạt động và phần cứng sử dụng để đào tạo mô hình, Bảng 6 mô tả những số liệu này.

Bảng V. So sánh các phương pháp nghiên cứu

Phương pháp	**Độ chính xác (%)**	**Tốc độ xử lý (hình/s)**	**Thời gian dự đoán mỗi khung hình**	**Phần cứng sử dụng**
Phát hiện ngã bằng Threshold+SVM [21]	86	N/A	N/A	CPU
Phát hiện ngã bằng Openpose + CNN [22]	91.7	N/A	45s	GPU
Nhận dạng dáng điệu bằng Openpose + LSTM	91.2	13.5	2s	GPU

Phát hiện ngã là một ứng dụng trong giám sát sức khỏe. Phương pháp trích xuất con người bằng Threshold cho hiệu suất kém nên việc phân loại đạt độ chính xác thấp. Lý do chính là sự thay đổi môi trường sẽ làm ảnh hưởng đến hiệu quả nhận diện, ví dụ điển hình là sự thay đổi ánh sáng hay sự thay đổi hậu cảnh dù nhỏ nhưng vẫn khiến cho việc nhận diện đối tượng vô cùng khó khăn. Khi việc dự đoán sai thì dẫn đến việc phân loại gặp trở ngại lớn. Do đó, phương pháp Threshold phân loại bằng SVM trong [21] chỉ đạt kết quả nhận diện khoảng 86%.

Cùng phương pháp ước tính tư thế con người sử dụng Openpose nhưng thuật toán phân loại khác nhau nên cho ra những đánh giá khác, cụ thể QingzhenXu [22] sử dụng mạng Inception-ResNet-v2 – một mạng CNN để trích xuất đặc trưng và phân loại, tham số mạng khoảng hơn 54 triệu [23], mạng học khá sâu, đây chính là lý do tại sao độ chính xác mà tác giả đạt được cao hơn phương pháp của chúng tôi 0.5%. Nhưng để so sánh về khối lượng tính toán của mạng thì chúng tôi chắc chắn mạng LSTM mà chúng tôi sử dụng cho hệ thống nhận diện có số lượng tham số nhỏ hơn gấp 27 lần, khoảng 2 triệu tham số. Với đặc thù ứng dụng của hệ thống giám sát sức khỏe mà chúng tôi xây dựng là quan tâm đến hiệu năng hoạt động của hệ thống nên độ chính xác thấp hơn. Nhưng do khối lượng tính toán nhỏ nên so sánh về thời gian dự đoán khung hình/giây, hệ thống của chúng tôi nhanh gấp 22 lần và chúng tôi đánh giá được tốc độ xử lý khoảng 13.5 fps, trong khi phần cứng sử dụng để đào tạo mô hình có cấu hình: Core i5 8300H, RAM 16GB, GTX 1050 Ti. Đây chính là điểm nổi bật mà hệ thống chúng tôi đạt được.

C. Đánh giá hệ thống trên phần cứng

Sau khi có kết quả mô phỏng, chúng tôi nhận thấy rằng với kết quả mà chúng tôi đạt được có khả năng nhúng được vào phần cứng thật. Vì vậy, chúng tôi đã tiến hành đánh giá hệ thống trên thiết máy tính nhúng Jetson Nano Kit có thông số như trên Bảng VI.

Bảng VI. Thông số Kit Jetson Nano

GPU	128-core Maxwell
CPU	Quad-core ARM A57 143GHz
Bộ nhớ	4GB 64-bit LPDDR4 25.6 Gb/s
Hiển thị	HDMI/Display port
DC	5V 4A

Bảng VII. Đánh giá hệ thống trên Jetson Nano

Công suất trung bình	6166mW
CPU	44% - 1.5GHz
Ram	3GB/4GB
Bộ nhớ swap	1.6GB/8.1GB
FPS	2.2

Kết quả đánh giá của hệ thống thể hiện trên Bảng VII. Theo đó, hệ thống sử dụng khoảng 44% CPU ở tần số 1.5GHz, tiêu tốn khoảng 3GB/4GB của Kit, tốc độ xử lý một khung hình theo giây là 2.2fps, bộ nhớ swap chiếm 1.6GB/8GB. Từ các kết quả nhận được khá khả quan, chúng tôi đánh giá rằng hệ thống nhận diện dáng điệu mà chúng tôi xây dựng có khả năng áp dụng vào thực tế.

D. Hạn chế và cải thiện định hướng nghiên cứu

Mặc dù nhận được kết quả khá tích cực cho hệ thống, tuy nhiên ở một vài trường hợp thử nghiệm, hệ thống cho ra kết quả không tốt, ví dụ như ảnh hưởng từ môi trường như ánh sáng yếu, cảnh vật gần giống với người như cành cây, ma nơ canh, khung treo quần áo,… hệ thống có thể nhận nhầm thành con người. Ngoài ra

trong trường hợp đối tượng cần nhận diện là con người nhưng không đủ các bộ phận trên khung hình thì hệ thống cũng nhận diện sai hoặc không, trường hợp lớn hơn là khung hình có chứa trên 5 người, hệ thống khó có thể nhận diện được tất cả. Để cải thiện những hạn chế này, hướng tiếp theo chúng tôi sẽ thực hiện các bước như bổ sung dữ liệu, quan tâm đến mọi trường hợp xấu mà chúng tôi nêu ra; thực hiện thêm với nhiều dáng điệu hơn kết hợp thử nghiệm với nhiều đối tượng trong một khung hình; tăng tốc độ khung hình trên giây, cải thiện độ chính xác bằng cách tăng độ phân giải của video.

V. KẾT LUẬN

Bài báo tập trung vào nghiên cứu việc sử dụng CNN để ước tính tư thế con người và phân loại dáng điệu. Trong bài báo này chúng tôi đã nhận diện được các dáng điệu với độ chính xác trên 90% bằng cách ước tính khung xương sử dụng Openpose với Backbone mạng Mobilenet-thin và sử dụng LSTM để học các đặc trưng đồng thời phân loại các dáng điệu. Hệ thống của chúng tôi có khả năng chạy được trên phần cứng thật dựa vào kết quả chúng tôi thử nghiệm được trên Jetson Nano Kit.

Tuy nhiên hệ thống vẫn còn nhược điểm như kết quả nhận diện các dáng điệu chưa cao và tốc độ khung hình trên giây ở mức chấp nhận được. Lí do chính là do dữ liệu chúng tôi tự thu thập có chất lượng kém, ánh sáng yếu, thiết bị quay bị hạn chế. Trong tương lai, nếu được cung cấp một bộ dữ liệu chất lượng hơn, có thể hệ thống của chúng tôi sẽ cải thiện được các hạn chế nêu trên.Từ đó, ứng dụng kết quả nhận diện dáng điệu cho việc giám sát y tế như hướng dẫn tập thể dục, hướng dẫn tập võ,…

LỜI CẢM ƠN

Nghiên cứu này được thực hiện trong khuôn khổ đề tài do Bộ Giáo dục và Đào tạo, Việt Nam tài trợ với tiêu đề "Nghiên cứu phát triển hệ thống nhận dạng cử chỉ, hành động ứng dụng trí tuệ nhân tạo trong nhà thông minh" theo đề tài cấp bộ mã số B2020-BKA-06. Cảm ơn Bộ KHCN đã tài trợ trong quá trình thực hiện bài báo này.

TÀI LIỆU THAM KHẢO

[1] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, “Real-time foreground–background segmentation using codebook model,” Real-Time Imaging, vol. 11, no. 3, pp. 172–185, 2005, special Issue onVideo Object Processing.

[2] M.-C. Chen and Y.-M. Liu, “An indoor video surveillance system with intelligent fall detection capability,” Mathematical Problems in Engineering, vol. 2013, pp. 1–8, 11 2013.

[3] Y. Yuan, Z. Miao, and S. Hu, “Real-time human behavior recognition in intelligent environment,” in 2006 8th international Conference on Signal Processing, vol. 3, 2006.

[4] S. Shinde, A. Kothari, and V. Gupta, “Yolo based human action recognition and localization,” Procedia Computer Science, vol. 133, pp. 831–838, 2018, international Conference on Robotics and Smart Manufacturing (RoSMa2018).

[5] P. N. Huu, H. N. T. Thu, and Q. T. Minh, “Proposing a recognition system of gestures using mobilenetv2 combining single shot detector network for smart-home applications,” Journal of Electrical and Computer Engineering, vol. 2021, pp. 1–18, Nov. 2021.

[6] M. E. Amine Elforaici, I. Chaaraoui, W. Bouachir, Y. Ouakrim, and N. Mezghani, “Posture recognition using an rgb-d camera: Exploring 3d body modeling and deep learning approaches,” in 2018 IEEE Life Sciences Conference (LSC), 2018, pp. 69–72.

[7] Z. Huang, Y. Liu, Y. Fang, and B. K. P. Horn, “Video-based fall detection for seniors with human pose estimation,” in 2018 4th International Conference on Universal Village (UV), 2018, pp. 1–4.

[8] J. Li, W. Su, and Z. Wang, “Simple pose: Rethinking and improving a bottom-up approach for multi-person pose estimation,” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 11 354–11 361, April 2020.

[9] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, “Rmpe: Regional multi-person pose estimation,” in 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2353–2362.

[10] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, “Openpose: Realtime multi-person 2d pose estimation using part affinity fields,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 1, pp. 172–186, 2021.

[11] A. Toshev and C. Szegedy, “Deeppose: Human pose estimation via deep neural networks,” in 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1653–1660.

[12] A. Jain, J. Tompson, Y. Lecun, and C. Bregler, “Modeep: A deep learning framework using motion features for human pose estimation,” Sept. 2014, pp. 302–315.

[13] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv 1409.1556, Sept. 2014.

[14] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, “Realtime multi-person 2d pose estimation using part affinity fields,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1302–1310.

[15] D. Sinha and M. El-Sharkawy, “Thin mobilenet: An enhanced mobilenet architecture,” 10 2019, pp. 0280–0285.

[16] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” ArXiv, vol. abs/1704.04861, 2017.

[17] S. Ioffe, “Batch renormalization: Towards reducing minibatch dependence in batch-normalized models,” Feb. 2017.

[18] H. Sak, A. Senior, and F. Beaufays, “Long short-term memory recurrent neural network architectures for large scale acoustic modeling,” Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, pp. 338–342, Jan. 2014.

[19] J. Kim, J. Kim, T.-T.-H. Le, and H. Kim, “Long short term memory recurrent neural network classifier for intrusion detection,” Feb. 2016, pp. 1–5.

[20] K. Duan, S. Keerthi, W. Chu, S. Shevade, and A.-N. Poo, “Multi-category classification by soft-max combination of binary classifiers,” June 2003, pp. 125–134.

[21] A. Shahzad and K. Kim, “Falldroid: An automated smart-phone-based fall detection system using multiple kernel learning,” IEEE Transactions on Industrial Informatics, vol. PP, pp. 35–44, May 2018.

[22] Q. Xu, G. Huang, M. Yu, and Y. Guo, “Fall prediction based on key points of human bones,” Physica A: Statistical Mechanics and its Applications, vol. 540, p. 123205, 2020.

PROPOSING POSURE DETECTION SYSTEM FOR MONITORING SPORT HEALTH

Abstract: The article proposes a system of posture recognition applying to medical and health monitoring in exercise and sports. The method that we use is to

estimate human postures by the Openpos algorithm whose backbone is MobilenetV2 network. Besides, the system tracks and classifies those postures by LSTM network. The four poses that are used for training include kicking, punching, jumping, and elbow training. The output of the system is to extract the human skeleton and corresponding label for poses. The results show that an accuracy of system is 91% that is capable of applying to medical monitoring, sports or controlling electronic devices in smart homes.

Keywords: Openpose, VGG19, Mobilenet, Mobilenet-thin, LSTM, pose estimate, extract skeleton, pose human

Nguyen Huu Phat, nhận bằng kỹ sư2003), thạc sỹ (2005) ngành Điện tử và Viễn thông tại Đại học Bách Khoa Hà Nội (HUST), Việt Nam và bằng tiến sĩ (2012) về Khoa học Máy tính tại Viện Công nghệ Shibaura, Nhật Bản. Hiện tại, đang là giảng viên tại Viện Điện tử Viễn thông, HUST, Việt Nam. Các nghiên cứu gồm xử lý hình ảnh và video, mạng không dây, big data, hệ thống giao thông thông minh (ITS), và internet của vạn vật (IoT). Ông đã nhận được giải thưởng bài báo hội nghị tốt nhất trong SoftCOM (2011), giải thưởng tài trợ sinh viên tốt nhất trong APNOMS (2011), giải thưởng danh dự của Viện Công nghệ Shibaura (SIT).

Nguyễn Thị Ngọc, Hiện tại là sinh viên Viện Điện tử Viễn thông, Trường Đại Học Bách Khoa Hà Nội. Hướng nghiên cứu gồm xử lý hình ảnh và video kỹ thuật số và các ứng dụng nhà thông minh.

Phạm Ngọc Thiện Hiện tại là sinh viên Viện Điện tử Viễn thông, Trường Đại Học Bách Khoa Hà Nội. Hiện đang công tác tại trung tâm nghiên cứu Viện Điện Tử Viễn Thông, Đạo Học Bách Khoa Hà Nội. Hướng nghiên cứu của anh gồm hệ thống nhúng, xử lý hình ảnh và video, số và các ứng dụng thông minh.