Fitness Activity Recognition

Demo 2
5/6/2022

Brady Hong, Alqama Sams, Ahmed Ceif, Alex Lidiak, Devansh Sharma, Sarthak Gupta, Raghav Kachroo headed by Mike Chung

SPEC: Seeing People in the Wild with an Estimated Camera

What is it?

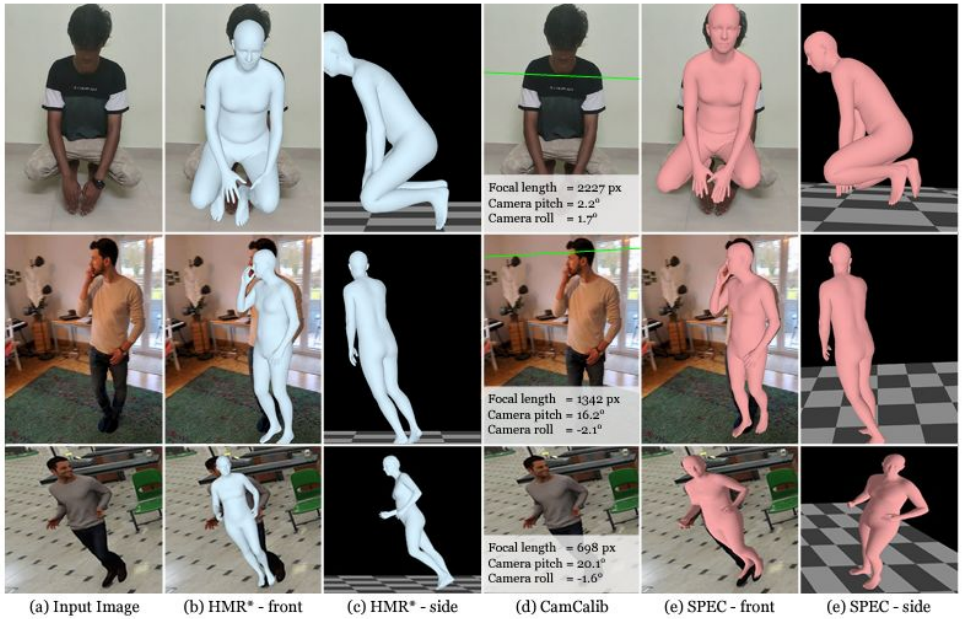
- Images are input
- Estimates the perspective camera
- Reconstruct 3-d bodies
- Front & Side

How it can help?

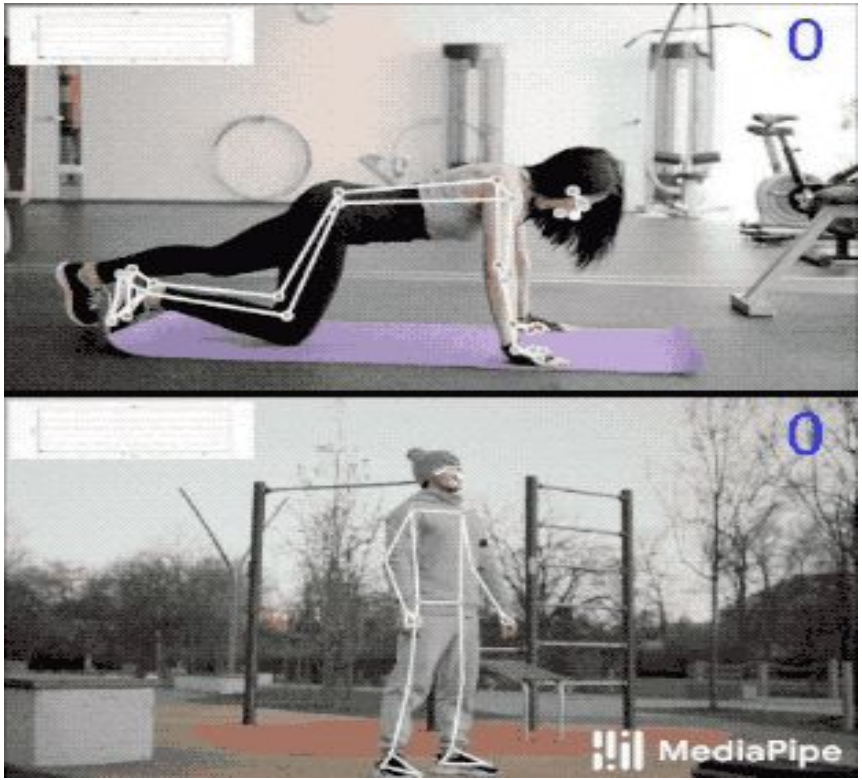
- Can allow us dive deeper into specific poses
- Maybe useful for creating new datasets

Limitations:

- High complexity may result in decreased speed.

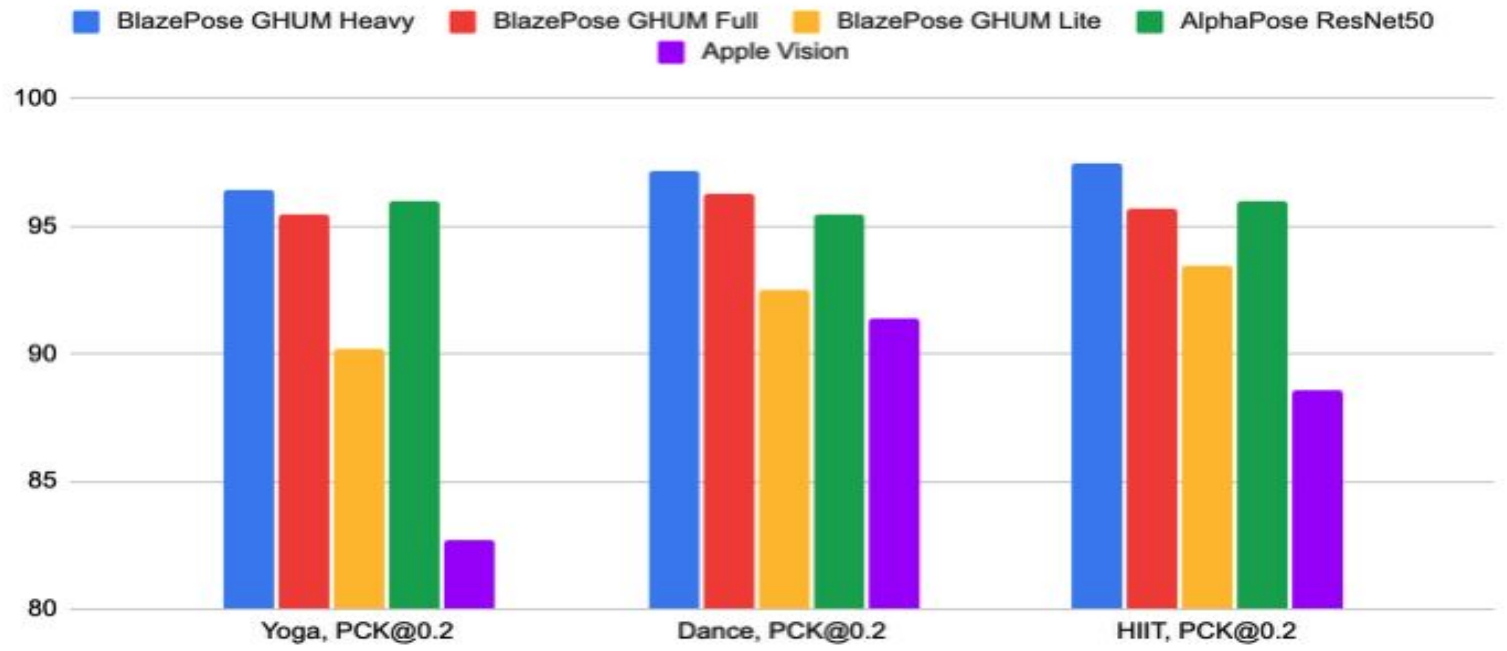


MediaPipe Pose(BlazePose)

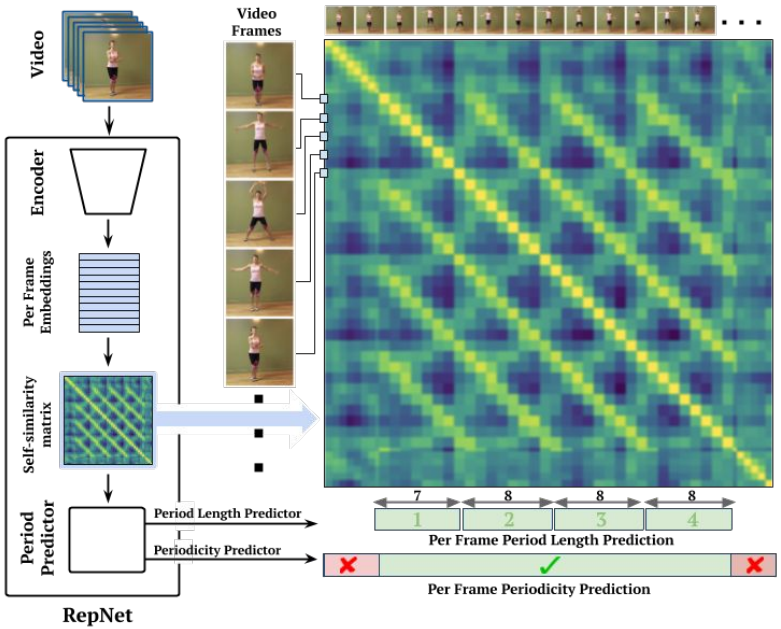


- **MediaPipe Pose(BlazePose)** extracts 3D coordinates for 33 pose estimation joint (Previous method **MoveNet** extracts 2D coordinates for 17 pose estimation joints)
- Can **calculate the joint angles** and can apply them to classify motions like yoga pose and weight training
- Can perform **repetition counting**

MediaPipe Pose(BlazePose)



Google RepNet



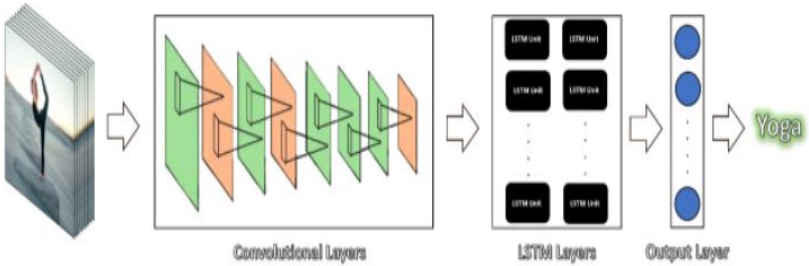
RepNet is a model that takes as input a video that contains periodic action of a variety of classes (including those unseen during training) and returns the period of repetitions found therein.

The model consists of three parts: a frame encoder, an intermediate representation, called a temporal self-similarity matrix, and a period predictor.

Identifying spatio-temporal relations between frames using LSTM

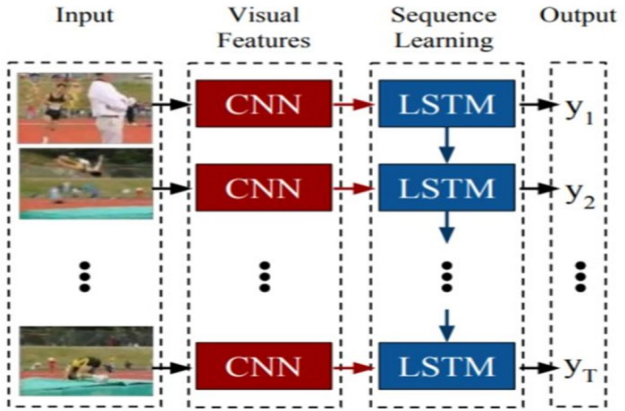
1. LRCN (Long-term Recurrent Convolutional Neural Network):

- LRCN combines CNN and LSTM layers in a single model.
- Convolutional layers are used for spatial feature extraction from the frames, and the extracted spatial features are fed to LSTM layer at each time-steps for temporal sequence modelling.
- The network learns spatio-temporal features directly in an end-to-end training, resulting in a robust model.

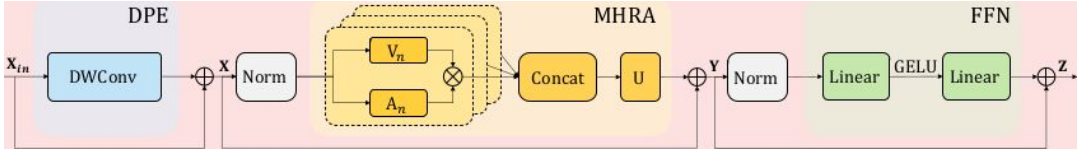


2. ConvLSTM:

- Variant of an LSTM network that contains convolutions operations in the network.
- Capable of identifying spatial features of the data while keeping into account the temporal relation.
- This approach effectively captures the spatial relations in the individual frames and the temporal relations across the different frames.



UniFormer

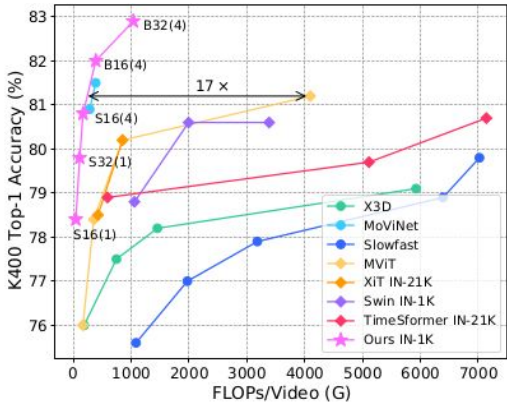


Similar to some of the above approaches, use the efficiency and spatial modeling strengths of convolutions, then add the temporal modeling strengths of another model, the Transformer. But this model treats spatial and temporal components equivalently!

UniFormer Block (above):

1. 3D depthwise convolution (DPE) to efficiently encode redundant spatiotemporal information present in local frames into **tokens**.
2. A multi-head relation aggregator (MHRA) that concatenates resulting tokens weighted by a **Token Affinity An**. **An** can be **local** or **global**.
3. Feed Forward Network (FFN) for pointwise enhancements of tokens.

The full model uses 4 blocks of convolutions with {64, 128, 320, 512} channels each followed by multiple UniFormer blocks {3, 4, 8, 3} or {5, 8, 20, 7} and a final pooling plus fully connected layer for outputting class predictions. The first two UniFormer blocks utilize the **local** affinity, while the latter two use the **global**. This design efficiently tackles video redundancies in shallow layers while using global similarities in deeper layers to learn global spatiotemporal features.



Performance comparison on the *Kinetics-400* action dataset. UniFormer variants all outperform alternatives in efficiency and accuracy.

MoveNet

One of the amazing thing about this model is it is actually very fast and does not require high specs gpu

MoveNet.SinglePose

Model Details

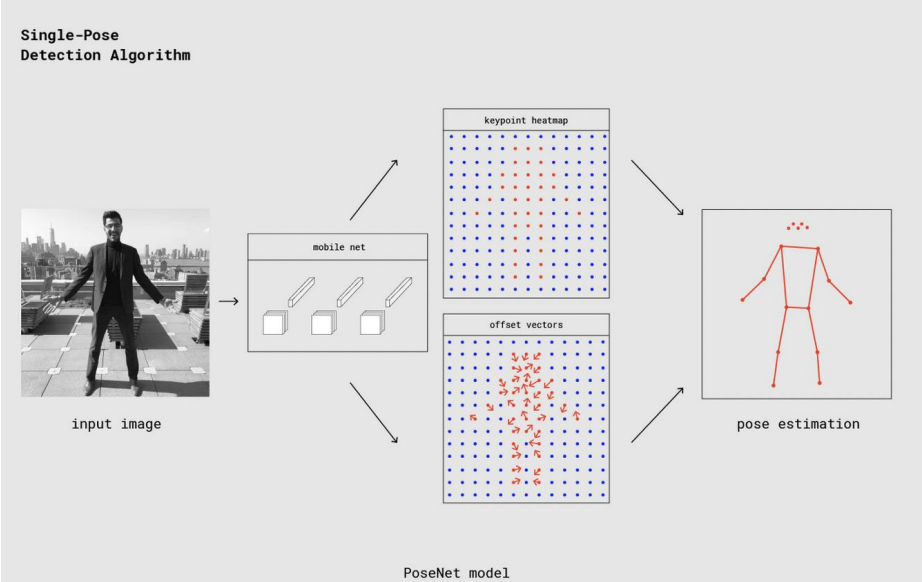
A convolutional neural network model that runs on RGB images and predicts human joint locations of a single person. The model is designed to be run in the browser using Tensorflow.js or on devices using TF Lite in real-time, targeting movement/fitness activities. Two variants are presented:

- **MoveNet.SinglePose.Lightning:** A lower capacity model that can run >50FPS on most modern laptops while achieving good performance.
- **MoveNet.SinglePose.Thunder:** A higher capacity model that performs better prediction quality while still achieving real-time (>30FPS) speed. Naturally, thunder will lag behind the lightning, but it will pack more of a punch.

PoseNet

Biggest selling point of PoseNet is multi-person pose detection, which is out of scope for our project.

Included with the Tensorflow Pose Detection library, which also includes MoveNet and BlazePose. BlazePose offers a 33 keypoints compares to the 17 of the COCO points offered in MoveNet and PoseNet, providing additional keypoints for face, hands and feet. MoveNet on the other hand provides a high level of accuracy and speed while also allowing for 50+ FPS on most modern laptops and phones. TensorFlow Lite also allows for deployment of MoveNet and PoseNet but even in those scenarios PoseNet seems to be outperformed by MoveNet, making the argument worse for the viability of PoseNet.



Conveniently, the PoseNet model is image size invariant, which means it can predict pose positions in the same scale as the original image regardless of whether the image is downscaled. This means PoseNet can be configured to have a *higher accuracy at the expense of performance* by setting the **output stride** we've referred to above at runtime. An output stride of 32 will result in the fastest performance but lowest accuracy, while 8 will result in the highest accuracy but slowest performance. Image scale factor can also be similarly tinkered with.

Next Steps

Start exploring the implementation of the finalized models, which are BlazePose, RepNet and MoveNet.

Further explore alternative feature engineering and classification processes.

If on track, contribute towards collecting and annotating further videos.