Yoga Pose Estimation Using Deep Learning

Introduction

Yogify (https://www.yogifi.fit/) has a successful active yoga mat product that uses sensors to guide the user in real time to optimize their yoga experience. One of the difficult problems to tackle is to find out if the practitioner is executing a pose correctly without a supervisor. In this study, we explore the possibilities of using live video data from a device such as a mobile phone that captures the images of the subject and automatically provides guidance on the correct posture. The method needs to be accurate as well as power efficient. Any deep network used should be able to run on a typical mobile smartphone.

Problem Analysis

Based on my understanding of the problem, in order to determine the correctness of a pose, we need to find out the relative positions and orientations of different parts of the body. I think it can be broken down into the following steps:

1. Capture images of the subject using a mobile camera positioned at an appropriate position
2. Estimate the 3-D positions of various parts of the body of the subject who is performing an asana
3. Compare the reconstructed 3-D pose with the ideal 3-D pose that is stored in a library for various poses
4. Prompt any corrective steps to achieve the right pose

The library of ideal poses can be created off-line by doing steps 1-2 when an expert performs the asana. Multiple experts can perform it to capture the natural variations in pose and body structure

Literature Survey

This section discusses the extant research in two areas, depth estimation and body parts segmentation

Depth Estimation

There is a wealth of literature that describes how to obtain 3-D depth information from a 2-D image. Here is the summary of the literature study that I performed:

There exist two methods to estimate depth from pictures - geometrical methods, and sensor based methods.

Geometric methods consist of Structure from Motion(SfM), and Stereo Vision Matching. SfM constructs a 3-d image from a sequence of 2-d images through feature sorrespondences and geometric constraints, so the accuracy of depth estimation relies on the correctness of the feature matching, ie, correctly corresponding the same objects in the many images taken from different angles. Stereo Vision matching, as the name suggests, uses two images from slightly different viewpoints to create a disparity map by simulating the way the human eyes work. The transformation between the two cameras is calibrated in advance.

These methods are perform moderately well in efficiently estimating the depth of sparse points, but they depend on more than one image. Obtaining depth estimation from a single image is still a significant challenge using geometric methods.

As for sensor based methods, three main methods exist:

RGB-D (Red Green Blue – Depth): This has been a widely used method for more than a decade now (example, the kinect by xbox). Its main limitation is that of outdoor shots, where it performs poorly, but for this application of detecting yoga poses indoor, it may perform well. However, it would be expensive and impractical to integrate a sensor with a mobile device to measure depth.

LIDAR (Light Detection And Ranging or Laser Imaging, Detection, And Ranging): LIDAR has become a very reliable calculator of depth from single images in the past few years, and is probably the most useful and accurate of all other depth estimation methods we have currently. However, its major limitation is the high amount of power it uses, which makes it difficult or nearly impossible to incorporate into embedded systems and, in this case, smartphones.

Deep-Learning: The popularity of the use of deep learning along with computer vision has skyrocketed in the last few years, and applications of Deep Learning have now extended well to depth estimation - both monocular and binocular. One of the major disadvantages of using deep-learning is the lack of accurately annotated images for training. Researchers opine that methods using supervised learning are far more accurate than those with semi supervised ones (Zhao et al, 2020).

Mayer et. al (2016) uses optical flow and extended its use to make disparity maps. Its exact architecture is unclear to me for now, but optical flow simply measures how much objects move between frames, and this can be used as the basis for a disparity map, which shows how much the apparent change in position would be if shot from different angles. This, evidently, is an inverse of a depth map, and consequently, depth information could be derived from it.

Laina et. al (2016) developed a new approach using ResNet to increase the number of layers in the deep learning model. As the number of layers increases, processing becomes slower and more difficult, which allows for residual layers to ease up the process. The fact that ResNet is a classification model calls for an important modification - to replace the last few layers(classification layers) to depth-estimation layers(continuous) - as we do not want to predict discrete values, but continuous ones. Further, there is an addition of up-sampling blocks as the higher layers make the resolution of the frame lower and lower. This method achieves higher accuracy due to the larger number of layers.

Chen. et al use relative depth annotations rather than ground truth. A relative depth would be good enough for yoga applications since absolute depth is not required to check the correctness of pose. Relative depths of various parts of the body would be sufficient..

An important error calculation metric is the use of log-space, as, with most methods, the accuracy of the estimated depth decreases with depth. Therefore, rather than punishing all errors equally, the use of log-space looks only at the ratio of estimated depth and actual depth to calculate the error (Estimated depth and actual depth of 40 and 50 respectively would be treated the same as 4 and 5 respectively)

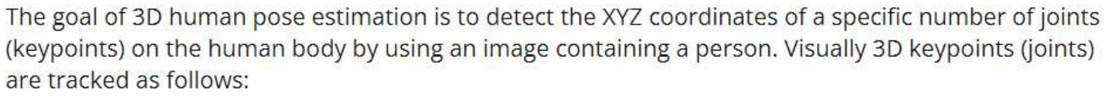
A significant amount of research has been done in the unsupervised and semi-supervised depth estimation methods too. A reliable algorithm of this kind would prove helpful as it would negate the disadvantage of lack of annotated ground truth images required for supervised learning. However, these methods have not measured up close to the accuracy obtained by supervised methods, and hence, are not in popular use.

All in all, depth estimation could be relatively unproblematic in a controlled indoor environment of a yoga practitioner, considering we can acquire enough ground truth data using LiDAR or other methods.

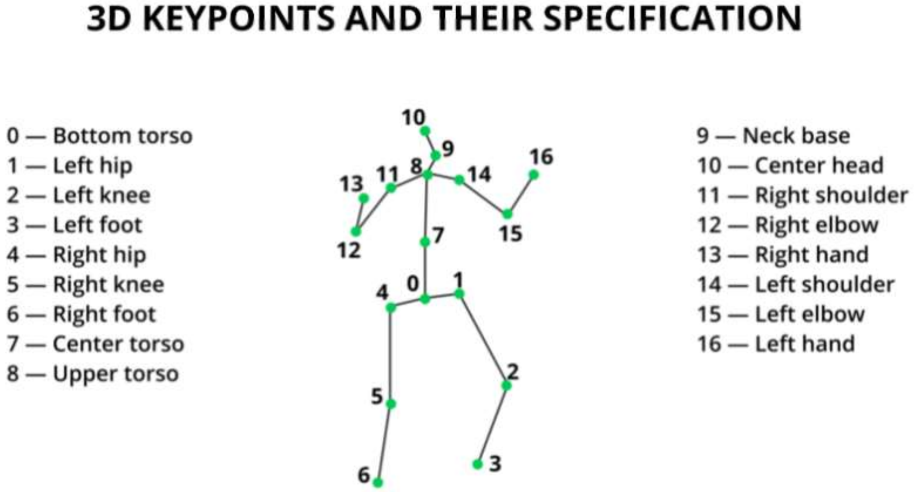
Body parts segmentation

It turns out that there is significant work done in this area as well. Human pose estimation is an important part of sports analysis.

Many machine learning techniques have been used to detect poses of the human body. Most methods treat the human body as an object composed of more than one rigid part connected by joints, allowing rotational and translational motion under three degrees of freedom.



The goal of 3D human pose estimation is to detect the XYZ coordinates of a specific number of joints (keypoints) on the human body by using an image containing a person. Visually 3D keypoints (joints) are tracked as follows:



3D KEYPOINTS AND THEIR SPECIFICATION

Source: 3D keypoints and their specification
https://mobidev.biz/wp-content/uploads/2020/07/3d-keypoints-human-pose-estimation.png

Most approaches are top down - first finding the body, then segmentation of body parts.
There are certain challenges that all machine learning methods face:

- Unknown location of human body in scene
- Self-occlusions
- Variety of environments
- Diverse body shapes
- Clothes

In the context of yoga pose estimation, self-occlusions may be the biggest challenge to overcome, considering the complexity and diversity of the many poses.

Any deep learning method has a major obstacle - difficulty in running real time due to the expensive computations required. However, yoga pose detection may not require such fast-running algorithms - it can either detect the pose only for a single frame once the pose has been executed, or can be given a few seconds to come up with the results if the procedure of the performance of the pose is important too.

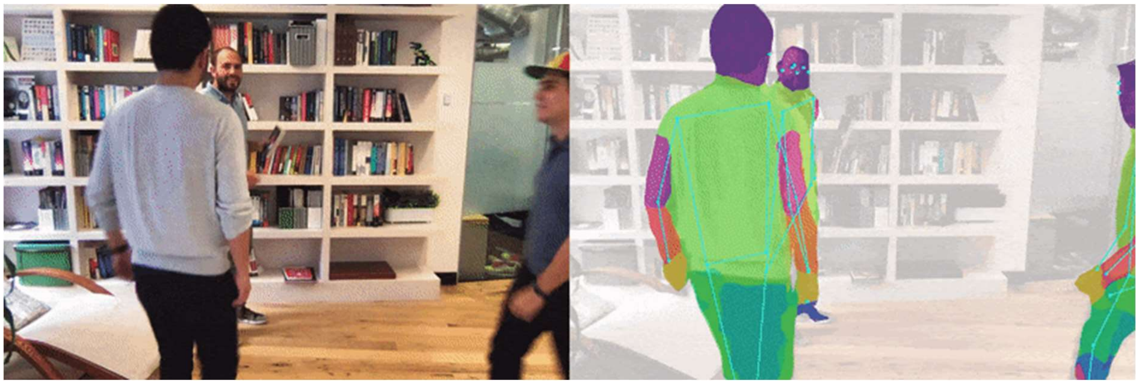
When detecting the pose of a body, two important results can be used to our advantage:

1. Of all the different body parts, one part can either be affected or unaffected by another. As it turns out, the body parts are sparsely related to each other - a matrix which represents the connection between any two body parts would be sparse.

2. Elimination of body priors - implausible body contortions - are an advantage in these algorithms (for example, an elbow bent with an angle greater than pi radians). Some deep learning algorithms also learn these contortions.

There is yet another factor to take into consideration. Since pose estimators may not be able to detect every intricate pattern of every body part, we could find out which body parts to focus on for each pose beforehand.

Popular problems such as AI fitness coach are examples of applications run by smartphones to detect the pose in real-time to check the correctness of weight-lifting poses.



(an Example of body part segmentation - Body Pix - which can be run on Mobile net)

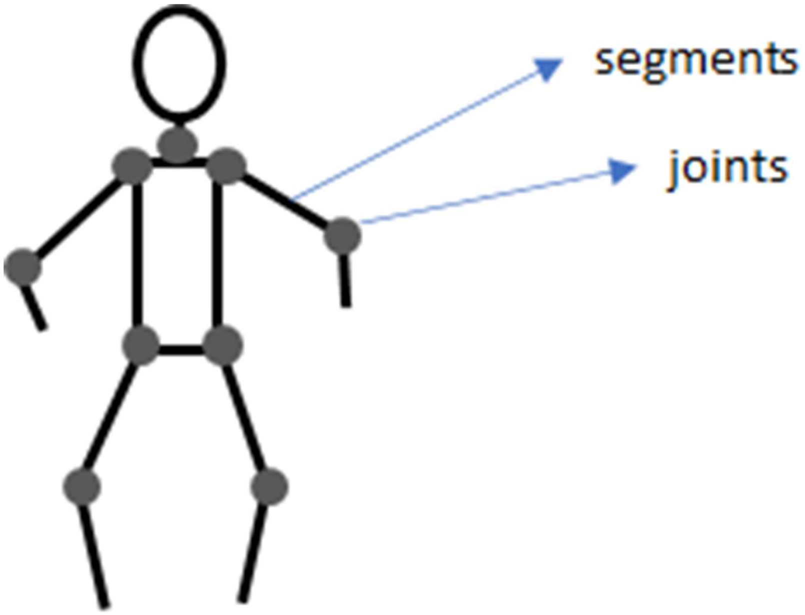
Source: https://blog.tensorflow.org/2019/11/updated-bodypix-2.html

Suggested methodology

The main task of the problem is to estimate the 3-D position of the various parts of the human body. There are two approaches that can be considered, which are described in following sections:

1. 3-D estimation using depth information
2. 3-D estimation without using depth information

In both of these methods, the model of the human body can be considered as discrete elements that are connected at the joints: shoulders, elbows, knees, hips and neck. The segment between joints can be considered as rigid for all the limbs, but not quite right for the torso, which can curve because of the backbone).



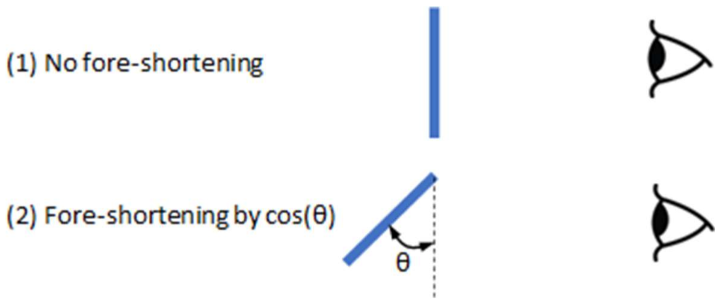
Method-1: Using depth information as well as body segmentation

In this method, the following steps are needed:

1. Depth measurement as described in section 2.1 is used to find the pixel level depth information of an image of the subject.
2. The same image is then processed using the body segmentation network described in section 2.2. This identifies the various joints and segments of the body
3. The depth values of pixels are mapped to the segments of the body. For each of the rigid segments (limbs), the depth values can be used to find the orientation of the segment in 3-D
4. By using the depth values of the segments, and by applying anatomical constraints, the 3-D model of the human body can be reconstructed
   [The algorithm for this needs to be developed]

Method-2: Using only body segmentation

This method can be considered if human body segmentation turns out to be too complex for a mobile phone. In this method, the property of limbs being well approximated by a straight line is exploited. The orientation of a straight segment in 3-D can be obtained by combining the observed 2-D angle and combining it with fore-shortening. To explain fore-shortening, consider the following: When a straight line is viewed perpendicularly by a camera, then the observed 3-D image has the maximum length. If the line is bent backwards or forwards, then its apparent length in the image is shorter. This effect is called fore-shortening. This is illustrated in the figure below:



If the expected length with no fore-shortening is known, then the value θ can be calculated. In other words, if the arm of a person appears shorter than normal, then it must be bent away or towards the camera. There is an ambiguity between the towards or away angle, which can be resolved using anatomical constraints.

If the fore-shortening angle of all the segments can be calculated, then it should be possible to reconstruct the complete 3-D model of the person.

This method would need a calibration step where the person stands erect in front of the camera, so that the length of the body segments without fore-shortening can be recorded.

Challenges

Back bends:

The pose estimation methods which treat the body as rigid rods connected at the joints do not account for the flexibility of the back, which often arches either forward or backward in yoga poses. However, the arch in the back can almost always be calculated geometrically once we know the fixed length of the back, the position of the hips, and that of the shoulders, as there is a unique curvature for any such position (the back cannot arch two different ways, with the higher back arching more, lower back less and vice versa).

Occlusion:

All parts of the body may not be visible to the camera. Some parts of the body may occlude other parts of the body. It may not be required to view all the body parts to determine the correctness of pose. Camera positioning can be chosen to minimize occlusions. Deep Learning based methods may be able to deal with some levels of occlusion by training with partial occlusion

Varying distance from the camera:

When using Method 2, the varying distance of the human body from the camera may present challenges as the apparent length of the body part in a frame is used in determining the angle it is bent. As the yoga practitioner moves a few feet back and forth the yoga mat, these distances can change. However, a calibration performed at the very beginning with the practitioner standing up-right can can give enough information to calibrate the human body at any distance from the camera by comparing the size of, say, the head, of the person in the image during the pose, to that of the same during the initial calibration. This difference in size can be used to determine the factor by which every body part will reduce in size

References

Chen, P. Y., Liu, A. H., Liu, Y. C., & Wang, Y. C. F. (2019). Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2624-2632).

Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., & Navab, N. (2016, October). Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)* (pp. 239-248). IEEE.

Mayer, N., Ilg, E., Hausser, P., Fischer, P., Cremers, D., Dosovitskiy, A., & Brox, T. (2016). A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4040-4048).

Zhao, C., Sun, Q., Zhang, C., Tang, Y., & Qian, F. (2020). Monocular depth estimation based on deep learning: An overview. *Science China Technological Sciences*, 1-16.