International Journal of Future Generation Communication and Networking
Vol. 13, No. 2, 2020 pp.1556-1561

Deep Learning Approach for generating 2D Pose Estimation from Video for Motion Capture Animation

Mr. Mohit Tiwari[1], Ms. Tripti Tiwari[2], Gopika Rajendran[3], Roberto Suson[4]

[1] *Bharati Vidyapeeth's College of Engineering, New Delhi, India*

[2] *Bharati Vidyapeeth (Deemed to be University) Institute of Management & Research, New Delhi, India*

[3] *Department of Computer Science, College of Engineering Kidangoor, Kerala, India*

[4] *Cebu Technological University, Philippines*

Abstract

As one of the most critical and frustrating computer vision issues, vision-based monocular humans pose estimation efforts to obtain the location of the human body from images or video. Rapid trends in deep learning techniques ushered meaningful strides and noteworthy advancements in human pose estimation. We propose a quick and efficient approach to detecting a human's 2D pose from a video. The approach uses an Affinity Vector Field (AVF) that learns how the body parts are related. The architecture codes for global interpretation, enabling a bottom-up approach that retains high precision in real time. The architecture was designed to use two branches of a certain method of sequential prediction to learn and link part locations together. Effectively, our approach was carried out over the human pose dataset of MPII.

Keywords: *Computer Vision, Deep Learning, Human Pose Estimation*

1. Introduction

The human pose estimation (HPE) task [1] developed over years is designed to obtain the human body posture from the sensor inputs given. Cameras also employ vision-based strategies to deliver such a solution. With deep learning in recent years, good performance has been shown on many tasks in the computer version, such as image classification, object detection [2]. HPE is also making significant progress in the use of deep learning technologies. HPE becomes a very popular research subject and can be applied to many applications such as action recognition, films and animation, virtual reality, computer-human interaction, video monitoring, medical assistance, etc..

Human 2D pose estimation is defined as the task with locating anatomical key points or "parts" which rely primarily on the identification of body parts [3]. Usually, HPE methods can be categorized as top-down and bottom-up methods based on the initial point of the prediction referred to as high-level inference or low-level pixel evidence. Top-down methods start with high-level inference for discovering the person in the picture and then generating the person's location in boundary boxes. But when the detector fails, it suffers from early obligation and there is no hope for recovery. At the other side, bottom-up processes initially identify the body parts of the person in the frame and then organize it either through fitting the body models or by algorithms. The parts of the body may be joints, limbs or small patches of templates based on specific strategies. But bottom-up strategies do not explicitly use global background from other parts of the body and human.

ISSN: 2233-7857 IJFGCN

In this work, we emphasize an approach that can be viewed as a marker-less 2D motion capture tool designed to infer human body movements in video. The bottom-up representation of the association scores via AVF is a set of 2D vector encoding the location and direction of the parts of the body over the frame domain to infer human pose in video. The key components of PE are heat maps, a visual representation that stores the confidence of the network, as some pixels contain a certain part of it. Thus the system produces 2D keypoints (X, Y coordinate values) for a given video as in Figure 1. We illustrate that these bottom-up depictions of detection and association simultaneously interpret the global context well enough to achieve high-quality results.

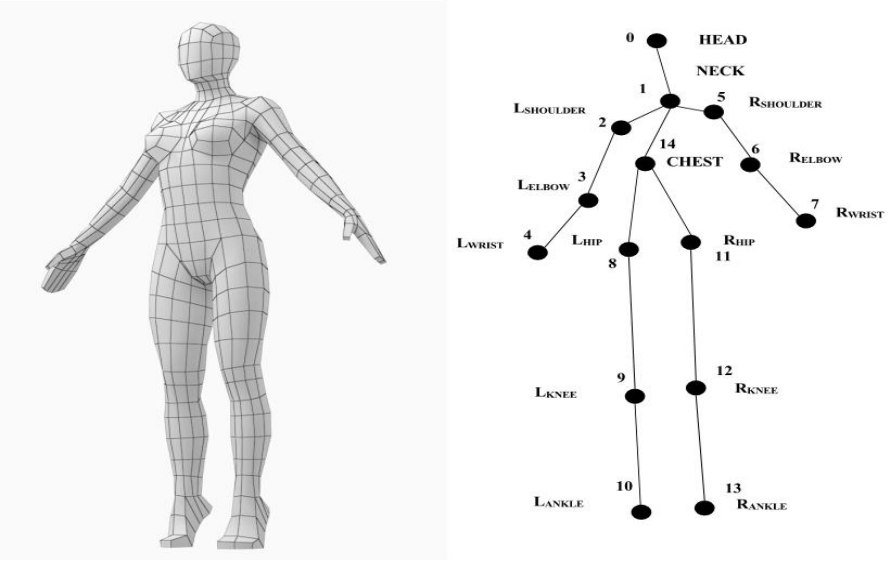


Figure 1: Virtual Human Model and its articulated stick mode for joints generated by processing the image given to the system

The structure of paper is organized as follows: Section 2 describes the methodology for estimating human pose from a video. The result and discussion are depicted in the section 3 shows the output obtained from our work.

2. Methodology

Figure 2 shows the entire architecture of the system. The method provides a framework for competitively performing pose estimation on multiple public benchmarks. It uses bottom up approach to determine human poses directly from pixel-level image proof. For this purpose, a two-branch fully convolutional neural network able to solve the problem using a multi-stage classifier where each stage improves the results of the previous one. A feed-forward network that sequentially foresees a set of 2D heatmaps ***Cs*** of body part locations and a set of 2D vector fields ***Af*** of affinities coding the degree of association between parts

ISSN: 2233-7857 IJFGCN

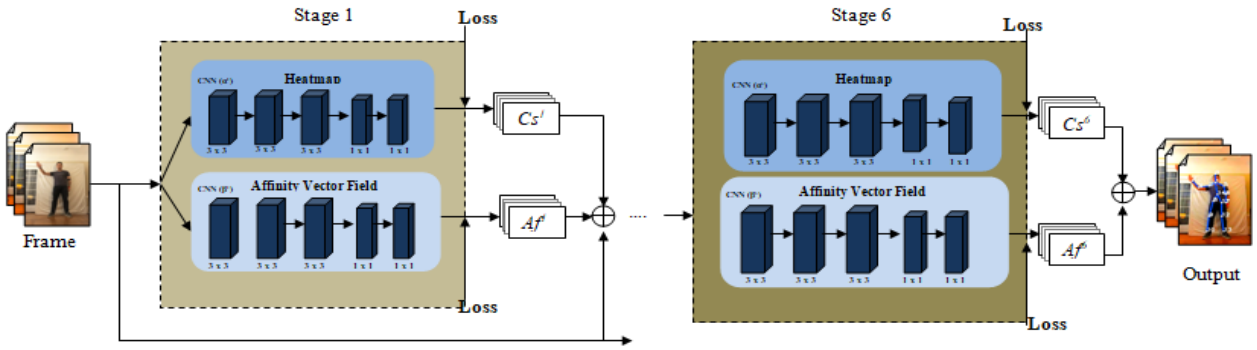
.

The last operation of the neural network returns a matrix consisting of 57 vectors as in the equation below. However this last operation is just a concatenation of two different vector as heatmaps and AVFs. It's important to understand those two feature vector.

$$\boldsymbol{HeatMap}: \mathbb{R}^2 \rightarrow \mathbb{R} \quad \boldsymbol{Affinity\ Vector}: \mathbb{R}^2 \rightarrow \mathbb{R}^2 \qquad (1)$$

The network is divided into two branches: The top branch predicts maps of confidence on each body part and the bottom branch predicts fields of part affinity. The system takes input video frames of size *width (W) x height (H)* and feeds these frames to CNN Network. Thus the system generates the anatomical key points in 2D positions for each frame as output of humans respectively. The first stage takes input image and predicts each keypoint's possible locations in the image with a confidence score (called the heatmap). Preferably, if a person appears on the frame, each confidence map should have a single peak when the corresponding part is visible. The location value $\boldsymbol{L_{Cs}}$ in $\boldsymbol{Cs}$ is defined as

Figure 2: Architecture of the two-branch multi-stage CNN. Each stage in the rst branch predicts confidence maps Cs, and each stage in the second branch predicts AVFs Af.



$$\boldsymbol{Cs^*(L_{Cs}) = e^{-\|L_{Cs} - X\|/\sigma^2}} \qquad (2)$$

where $\boldsymbol{\sigma}$ controls the distribution over the peak and $\boldsymbol{X} \in \mathbb{R}^2$ be the ground-truth position of each body part for the human in the frame. It then uses the fact that some joints are attached via limbs to introduce another CNN [4] branch which predicts 2D vector fields called Affinity Vector Fields (AVFs). A limb is created by connecting two parts and AVF encodes the path from one part to another; each limb is called a field of affinity between the parts of the body. If a point $\mathbf{L_{Af}}$ lies in the limb then its value $\boldsymbol{Af_{Limb}}$ in the AVF is a unit vector pointing from the starting point of the joint to the ending point of that limb; if it is outside the limb, the value is zero.

$$\boldsymbol{Af_{Limb}(L_{Af}) = \begin{cases} \vec{v}, & if\ limb\ on\ L_{Af} \\ 0, & otherwise \end{cases}} \quad \textbf{and} \quad \boldsymbol{\overrightarrow{v} = (X_2 - X_1)/\|X_2 - X_1\|_2} \qquad (3)$$

Thus the network produces a set of confidence score $\boldsymbol{Cs} \in \mathbb{R}$ for each keypoint and a set of affinity vector field $\boldsymbol{Af} \in \mathbb{R}^2$ over the successive stages, t ∈ {1,…,T} as

$$\boldsymbol{Cs^t = \alpha^t(F, Cs^{t-1}, Af^{t-1}), \quad \forall t \geq 2} \qquad (4)$$

$$\boldsymbol{Af^t = \beta^t(F, Cs^{t-1}, Af^{t-1}), \quad \forall t \geq 2} \qquad (5)$$

ISSN: 2233-7857 IJFGCN

where α and β are the CNNs for inference at each stage. At each stage, the predictions from both branches are convolved with the image features F for the next stage. Two loss functions are implemented at the end of each branch, one at each branch, in order to direct the network to iteratively predict body parts maps in the first branch and AVFs in the second branch. Each branch is an iterative predictive architecture, optimizing predictions over consecutive stages, with approximate supervision for each stage. This improves the prediction after each step, i.e. the confidence map is more accurate after passing through all 4 stages.

3. Result and Discussion

Our framework is implemented with help of Pycharm Community Editon 2019 software in Conda Interpreter, Windows 8.1 and above, OpenCV 3.4, Python 3. The hardware specifications are 4 GB RAM minimum (8 GB RAM recommended), 1.5 GB hard disk space + at least 1 GB for caches, 1024x768 minimum screen resolution.

Trained on the MPII human pose dataset[5] with Caffe deep learning platform, the network is a state-of-the-art standard for expressive human pose estimation. Caffe-based network uses two files: the .prototxt file that defines the architecture of the neural network as the way the different layers are arranged and the .caffemodel file that stores the weights of the trained model. The MPII model produces 15 points as in

Table 1.

Head	0
Neck	1
Right Shoulder	2
Right Elbow	3
Right Wrist	4
Left Shoulder	5
Left Elbow	6
Left Wrist	7
Right Hip	8
Right Knee	9
Right Ankle	10
Left Hip	11
Left Knee	12

ISSN: 2233-7857 IJFGCN

Left Anklet	13
Chest	14
Background	15

Table 1: Generated KEY-POINTS and Associated Values by MPII Dataset

The network for estimation is loaded into the memory as cv2.dnn.readNetFromCaffe(protoFile, weightsFile). The input frame is converted to input blob, so it can be transmitted to the network. This is done using the blobFromImage function which converts the image to Caffe blob format from OpenCV format. Once the image is passed to the model, the forward method for the DNN class in OpenCV makes a forward pass through the network, which can make the predictions. The prediction produces a 4D matrix which consist of :

- The first dimension is that of the image ID.
- The second dimension displays a keypoint index. The model produces maps of Confidence and Part Affinity that are all concatenated.
- The third dimension is output map height.
- The fourth dimension of the output map is its width.

The location of the keypoint is defined by finding the maximum of that keypoint's confidence map. To reduce false detections it also uses a threshold value (equal to 1). The final result is collected for a given sample video in the form of .avi file as shown in Figure 3 with human 2D keypoint estimates.

Figure 3: Input Frame and Output Frame with generated keypoint

4. Conclusion and Future Work

In this paper we find an essential element of interpretation about how the machine is equipped with human realistic behavior in real time as real-time algorithms to detect the human 2D pose in images. We are addressing an implied non-parametric description of keypoint association, which codes the position and orientation of the human limbs. Secondly, architecture has been developed for the identification of parts and the integration of parts to know together. Because of the noise experienced in video frame from the recorded video the pose estimate doesn't often work. Finally, we demonstrate significant failure cases in our work in Figure 4.

ISSN: 2233-7857 IJFGCN

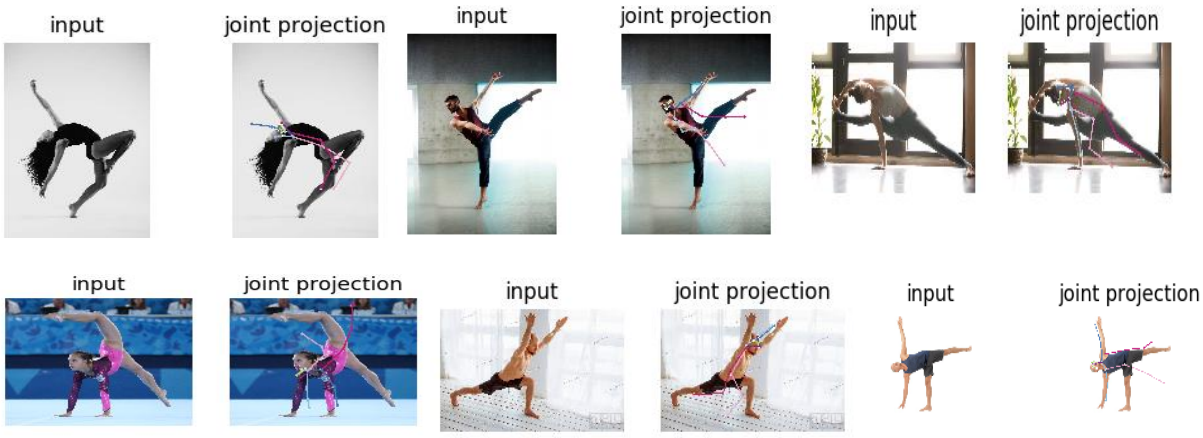


Figure 4: Common failure cases with rare poses and false part detections

Future work should be dedicated to promoting greater ties of computer animation and human pose estimation to imitate user gestures in the 3D virtual character model. Besides that, create a algorithm that detect 3D key-points (x, y, z coordinates) from the 2D key-points thus making the character's behavior over the 3D domain. Future work is suggested to improve depiction of motion and motion capturing. Finally, efforts will be made to enhance the device accuracy through the implementation of new machine learning frameworks TensorFlow, Keras etc.

5. Acknowledgments
We are grateful to all those who provided insights or contributed in one way or the other towards the success of the study.

6. Declaration of Conflicting Interests
We declare that there is no conflict of interest.

7. Data Availability Statement
All relevant data are within the paper.

Reference

[1] Y. Chen, Y. Tian, and M. He, "Monocular human pose estimation: A survey of deep learning-based methods," Comput. Vis. Image Underst., 2020, doi: 10.1016/j.cviu.2019.102897.
[2] Fabrizio Lamberti,Valentina Gatteschi, Alberto Cannavo, and Paolo Montuschi ;"Virtual Character Animation Based on Affordable Motion Capture and Reconfigurable Tangible Interfaces ";IEEE transactions on visualization and computer graphics,vol. 24, no. 5; 2018.
[3] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," 2017, doi: 10.1109/CVPR.2017.143.
[4] Dr.Shadab Adam Pattekari, Dr.Shamima Akter Somi, Piyal Saha, Anit N Roy, Aswin S, Chinnu Rajesh, "Detection Of Pandemic Virus Covid-19 Using CNN", IJAST, vol. 29, no. 8s, pp. 3954 - 3958, Apr. 2020
[5] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New

ISSN: 2233-7857 IJFGCN

benchmark and state of the art analysis," 2014, doi: 10.1109/CVPR.2014.471.

ISSN: 2233-7857 IJFGCN