Towards Agile Human Pose Estimation: A Case Study in Ice Hockey Analytics

Bavesh Balaji[1] **David A Clausi**[1]
[1]Vision and Image Processing Group, System Design Engineering, University of Waterloo
{bbalaji, dclausi}@uwaterloo.ca

Abstract

Human Pose Estimation is a fundamental task in vision-driven hockey analytics which is highly useful for a variety of downstream tasks such as action recognition and player assessment. Estimating human pose keypoints from a monocular video is a challenging task, especially in agile environments. Fast-paced games such as Ice-hockey and Lacrosse often have large amounts of motion blur and occlusions in their video feed. However, most of the previous research works use high-resolution inputs curated in isolated environments. As a result, the existing benchmarks do not capture the model's performance in real-world agile settings. Hence, in this work, we evaluate several state-of-the-art (SOTA) 2d pose estimators on our custom Ice-hockey dataset created from broadcast hockey videos. We conduct extensive comparison studies on 4 SOTA pose estimators, both quantitatively and qualitatively, and empirically demonstrate that Multi Stage Pose Networks (MSPN) produces the best results on our dataset.

1 Introduction

Human action recognition has been one of the most researched problems in computer vision. The most successful approaches model this problem as a function of pose estimation, based on the keypoint representation of human joints. Pose Estimation, in general, refers to predicting the pose of an entity (humans, in most cases), and helps to determine their physical orientation with respect to an environment. This plays a vital role in sports analytics, in estimating the 'correct pose' of sportsmen to decode their style of play, analyze their actions, avoid injuries, plot their activities, and for a gamut of other use cases. Vision-based predictions of poses are especially relevant, due to their minimally invasive approach and methodology.

But, there exist certain constraints in vision settings, which include occlusion, foreshortening, shadows, depth ambiguity, and misdirection which results in inconsistent pose estimates, especially in uncontrolled agile environments. This is easily observed in fast-paced team sports such as Ice-hockey, basketball, soccer, rugby, etc., where there exist several constraints to estimate the relevant pose of a player. This has proven to be an ill-poised problem in vision.

In our work, we study the different approaches to mitigate this, by an end-to-end implementation and comparative study of novel state-of-the-art architectures for 2D Pose Estimation. To this end, we utilize a novel Ice-Hockey dataset, which contains manually annotated 2D ground truth keypoints and bounding boxes, sampled at 30 frames per second. We opt for the top-down approach (human bounding-box detection followed by pose estimation) as it has proven to be the most efficient and handpick four models [1–4] based on our use-case.Through this exploration, we aim to contribute to the ongoing advancement of pose estimation techniques, ultimately enhancing their efficacy in demanding real-world scenarios.

2 Related Works

Human Pose estimation is a fundamental problem in computer vision that has been researched for a long time. Classical approaches to pose estimation include pictorial structures models [5] and Flexible mixture-of-parts [6]. These frameworks broadly use tree-based probabilistic graphical approaches to model the spatial relationships between different joints. Other classical approaches involve extracting important features through various feature extraction techniques such as contour detection, color histograms, and histogram-of-gradients (HOG) [7]. However, these approaches were not able to handle occlusion and model the spatial information effectively.

The advent of deep learning and convolutional neural networks helped in efficient feature encoding and better generalization. Hence, a multitude of works has been conducted on the use of deep learning approaches for pose estimation. Toshev et al. [8] initially formulated pose estimation as a regression problem of finding body joints, and used CNNs to estimate the poses. Pischulin et al. [9] followed a bottom-up approach by detecting all the keypoints first using CNNs, and then using ILP to cluster the keypoints. Newell et al. [10] was the first to use a multi-stage architecture where each stage consisted of repeated down and upsampling layers with skip connections [11] to extract as much information as possible. Subsequently, a lot of architectures [1, 2, 12, 13] use the multi-stage technique and follow a top-down approach.

Recently, transformer architectures have gained a lot of traction in various computer vision tasks. Most models [4, 14, 15] incorporate CNN backbones to extract features and then employ transformer encoder and/or decoder layers to refine the features. On the other hand, HRFormer [16] and ViTPose [17] directly use transformers to extract features and predict keypoints.

3 2D Pose Estimation

3.1 MultiStage Pose Network

Multi-Stage Pose Networks [1] adopt the top-down approach in two steps. In the first step, manual annotations of all the players on the rink are used to crop the input frames and create multiple images consisting of a single person. The pose estimation network then uses repeated down and up-sampling to continuously refine the estimation of poses. This network mainly proposes three major design improvements on other multi-stage networks.

The first one is the equal channel width design followed in all the networks. All the existing networks use the same number of channels in each level of a downsampling module. However, this reduces the size of the feature map as we go down a single stage, making it more difficult to capture relevant information. To solve this problem, this network doubles the number of channels(convolutional kernels) at every level of a downsampling module, maintaining the size of the feature map throughout the stage. This helps the model capture more information in the downsampling module, resulting in better localization of keypoints.

The second improvement made by this architecture is the cross-stage feature aggregation. This network enables us to propagate the features extracted during the initial stages by aggregating them with the features in successive stages. This helps in retaining a lot of information without adding a lot of layers, making the model more robust and foolproof.

The third and most important improvement made by the model is the coarse-to-fine supervision. At the end of every stage, the outputs are converted into gaussian heatmaps and compared with ground truth heatmaps to refine the localization accuracy. In this network, they perform this intermediate supervision at every stage using decreasing gaussian kernel sizes (instead of using same size kernels at every stage) as this gives a more accurate estimate of the features extracted.

3.2 High Resolution Network

High Resolution Network [2] is a multi-stage pose estimation network that focuses on producing and maintaining accurate high-resolution relationships. This network starts by extracting features at a higher resolution and then goes on reducing the resolution as we go deeper into the architecture. The one unique aspect of this network is the parallel connections across different resolutions, in comparison to the serial connections that are used in other pose estimation architectures.

These parallel connections help in effectively maintaining the high-resolution features extracted at the start of the network without losing important information. The other differentiating factor of this model from other existing models is the repeated multi-scale fusion across different resolutions of a single stage. This novel technique aggregates features from different resolutions by either upsampling(using nearest neighbor interpolation followed by 1x1 convolutions) or downsampling(using 3x3 strided convolutions). This concatenation of features within a single stage helps in producing refined and robust representations.

3.3 Distribution-Aware coordinate Representation of Keypoint

The state-of-the-art pose estimation models do not directly take the 2D coordinates of each joint as their input and predict 2D coordinates for every keypoint in a given image. This is mainly because the 2D coordinates do not contain any spatial and contextual information, making pose estimation extremely challenging. Hence, all the existing networks convert these 2D coordinates to heatmaps using a gaussian kernel to gain some much-needed spatial information. This work [3] focuses on improving this encoding and subsequent coordinate decoding part(after the predictions made by the network) and provides a more principled distribution-aware method.

The standard coordinate encoding methods downsample bounding boxes to a smaller dimension, transforming the ground-truth coordinates accordingly. This downsampling is defined as shown in equation 1.

$$g' = (u', v') = g/\lambda = (u/\lambda, v/\lambda) \tag{1}$$

In equation 1, $g = (u, v)$ as the ground-truth coordinate and λ is the downsampling ratio. After performing this downsampling, standard methods generally quantize these coordinates using the floor or the ceil function to facilitate kernel generation. However, this causes an inaccurate and biased representation because of quantization error. This work solves that problem by eliminating the quantization step and placing the center of the heatmap at the downsampled coordinate g'.

The standard coordinate decoding methods find the coordinates of the maximal and second maximal activation. The joint location is then predicted by shifting the location of the maximal activation 0.25 pixels towards the second maximal activation. This shifting is done to compensate for the quantization error. However, this is an empirical method that is found to have success but does not have any intuition behind it. Also, the predicted heatmap is not exactly gaussian in nature and hence, the point with the maximal activation may not be estimated accurately. Hence, this work proposes a theoretically sound method to explore the entire heatmap and find the underlying maximal activation by modulating the heatmap to make it gaussian and then using that fact along with the Taylor series to find the actual coordinates of the keypoint.

3.4 TransPose

This network [4] leverages the recent success of transformers in computer vision tasks and replicates it in pose estimation. More specifically, this network uses a common CNN backbone such as ResNet or HRNet to extract the features from the input image. The extracted features are then passed through N transformer encoder layers where each layer consists of a multi-self-attention head, layer normalization, and feed-forward neural networks. The attention layers help in capturing long-range relationships and reveal the dependencies that determine the location of the maximal activation. The N different attention layers capture different positions of maximal activation corresponding to different joints. The final attention layer acts as an aggregator, which collects contributions from image clues and forms the maximum positions of keypoints. The output from the final encoder layer is then passed through a regression head to find the heatmap for every keypoint.

4 Comparison Study

To estimate the 2D pose of the players for our custom dataset, we optimized the four state-of-the-art (SOTA) 2D pose estimation networks explained in Section 3. We fine-tuned these SOTA models for our ice-hockey dataset and compared the performance and robustness of these models.

4.1 Dataset

The dataset consists of a total of 10 broadcast video feeds where each sequence is recorded at 30 fps and encompasses a total of 9000 frames. Each frame in the dataset is manually annotated with 17 keypoints per player, out of which 5 keypoints correspond to the head. Since the head of hockey players are not visible due to the heavy helmets that they wear, we disregard those 5 keypoints and only consider the other 12 keypoints for our task. Some of the frames obtained from the broadcast video of the hockey dataset can be visualized in figure 1.

Fig. 1: Ice-hockey dataset

4.2 Implementation Details

All of our experiments were conducted using an NVIDIA Geforce RTX 2070 GPU with 8 GB RAM. All the 2D pose estimation models were fine-tuned using the pretrained COCO [18] models for 10 epochs. Because of memory and hardware constraints, we were unable to perform extensive experiments and could only use a batch size of 8. The input size of all the models was (192, 256).

In order to perform all our experiments, we used a 2-stage MSPN, HRNet-W48, DARK with HRnet-W32 as the backbone and TransPose with HRNet-W48 as the feature extractor. Furthermore, for MSPN, we used SGD [19] as our optimizer with a learning rate of 1e-2, momentum of 0.9, and a weight decay of 1e-5. As for all the other models, we used Adam [20] as our optimizer with a learning rate of 1e-3.

4.2.1 Metrics

We used the PCK metric to find out the accuracy of our models. We essentially find out the Manhattan distance between the predicted and ground-truth keypoints, and check if it is lesser than a threshold (20 in our case). The threshold value was found by doing a grid search over all values from 1 to 100. Additionally, in order to prevent occlusion from skewing our model's performance, we filter out the occluded points by using the confidence score of each prediction and consider prediction only if confidence of prediction is > 0.6.

4.3 Results

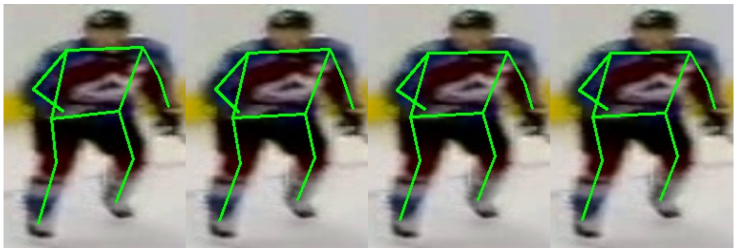
Table 1 provides a comprehensive overview of the training and validation per joint accuracy achieved by a network trained from the ground up. Notably, the results demonstrate that MSPN, as discussed in [1], outperforms other models on our dataset. A significant contributing factor to its superior performance is the effectiveness of the downsampling encoder, which enhances feature extraction, resulting in high-fidelity representations. This is corroborated by our qualitative findings presented in the accompanying figures, where MSPN exhibits a superior ability to model the pose of a player even in the presence of occlusions, distinguishing itself from other networks.

Moreover, our analysis, as depicted in Table 1, reveals variations in accuracy across different joints. Specifically, the shoulder, knee, and ankle joints exhibit higher accuracy compared to the wrist and elbow joints. This discrepancy underscores the challenges posed by

Table 1: Total and per joint accuracy of the fine-tuned networks

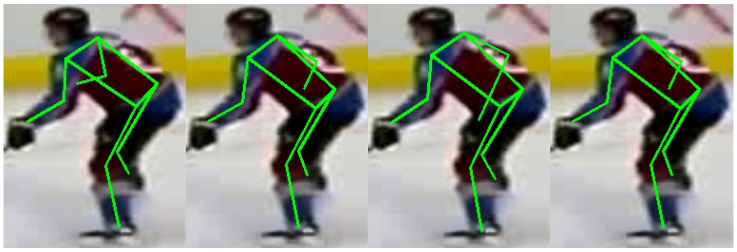
Joint	Training Accuracy(%)				Validation Accuracy(%)			
	MSPN	HRNet-W48	HRNet-W32 + DARK	TransPose	MSPN	HRNet-W48	HRNet-W32 + DARK	TransPose
Left Shoulder	**97.1**	95.27	95.08	91.44	**90.13**	86.13	86.92	85.83
Right Shoulder	**96.63**	95.23	95.07	91.53	**89.96**	86.23	86.98	84.95
Left Elbow	**94.86**	89.97	90.80	86.92	**87.12**	81.07	82.51	81.95
Right Elbow	**94.91**	88.33	89.52	85.53	**89.03**	81.69	85.96	84.00
Left Wrist	**93.34**	87.69	87.13	83.70	**86.13**	80.27	80.38	80.77
Right Wrist	**92.98**	86.11	84.58	81.16	**86.61**	81.18	82.26	82.04
Left Hip	**93.76**	90.52	89.07	84.78	**79.38**	75.69	75.04	74.80
Right Hip	**94.27**	90.72	88.88	84.4	**79.75**	75.63	78.45	79.50
Left Knee	**97.19**	94.38	94.70	92.59	**92.39**	89.54	89.25	88.27
Right Knee	**97.28**	94.67	94.78	92.15	**92.76**	89.35	90.66	80.81
Left Ankle	**97.21**	93.89	93.02	91.23	**92.01**	87.00	86.51	86.65
Right Ankle	**97.40**	94.18	93.92	91.4	**93.70**	87.48	86.19	86.93
Total accuracy	**95.71**	91.86	91.40	83.75	**88.35**	83.51	84.32	88.16

high motion-induced blur and occlusions in the vicinity of hand joints, which are in constant, rapid motion during gameplay.



(a) HRNet *(b)* DARK *(c)* MSPN *(d)* TransPose-H

Fig. 2: Inference of the implemented 2D pose estimators without Occlusion



(a) HRNet *(b)* DARK *(c)* MSPN *(d)* TransPose-H

Fig. 3: Inference of the implemented 2D pose estimators with Occlusion

5 Conclusion

This study has effectively showcased the performance of current state-of-the-art networks in challenging, fast-paced environments characterized by motion blur and occlusions. Through a comprehensive comparative analysis, we have delineated the strengths and weaknesses of these existing models, revealing that hand keypoints, particularly wrists and elbows, are susceptible to errors when subjected to rapid non-linear motion. The insights gained from this research hold substantial promise for the advancement of future networks, fostering their resilience in the face of blur, occlusions, and distortions. These findings are pivotal in the ongoing pursuit of developing robust computational systems capable of thriving in dynamic, real-world scenarios.

Our future work will focus on leveraging the pose of the stick as a prior for refining the pose of the wrist and elbow keypoints.

Acknowledgments

This work was supported by Stathletes through the Mitacs Accelerate Program and Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] W. Li, Z. Wang, B. Yin, Q. Peng, Y. Du, T. Xiao, G. Yu, H. Lu, Y. Wei, and J. Sun, “Rethinking on multi-stage networks for human pose estimation,” *arXiv preprint arXiv:1901.00148*, 2019.

[2] K. Sun, B. Xiao, D. Liu, and J. Wang, “Deep high-resolution representation learning for human pose estimation,” in *CVPR*, 2019.

[3] F. Zhang, X. Zhu, H. Dai, M. Ye, and C. Zhu, “Distribution-aware coordinate representation for human pose estimation,” *CoRR*, vol. abs/1910.06278, 2019. [Online]. Available: http://arxiv.org/abs/1910.06278

[4] S. Yang, Z. Quan, M. Nie, and W. Yang, “Transpose: Towards explainable human pose estimation by transformer,” *CoRR*, vol. abs/2012.14214, 2020. [Online]. Available: https://arxiv.org/abs/2012.14214

[5] V. Belagiannis, S. Amin, M. Andriluka, B. Schiele, N. Navab, and S. Ilic, “3d pictorial structures for multiple human pose estimation,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1669–1676.

[6] Y. Yang and D. Ramanan, “Articulated human detection with flexible mixtures of parts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2878–2890, 2013.

[7] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 1, 2005, pp. 886–893 vol. 1.

[8] A. Toshev and C. Szegedy, “Deeppose: Human pose estimation via deep neural networks,” *CoRR*, vol. abs/1312.4659, 2013. [Online]. Available: http://arxiv.org/abs/1312.4659

[9] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele, “Deepcut: Joint subset partition and labeling for multi person pose estimation,” *CoRR*, vol. abs/1511.06645, 2015. [Online]. Available: http://arxiv.org/abs/1511.06645

[10] A. Newell, K. Yang, and J. Deng, “Stacked hourglass networks for human pose estimation,” *CoRR*, vol. abs/1603.06937, 2016. [Online]. Available: http://arxiv.org/abs/1603.06937

[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[12] J. Huang, Z. Zhu, F. Guo, and G. Huang, “The devil is in the details: Delving into unbiased data processing for human pose estimation,” *CoRR*, vol. abs/1911.07524, 2019. [Online]. Available: http://arxiv.org/abs/1911.07524

[13] Y. Cai, Z. Wang, Z. Luo, B. Yin, A. Du, H. Wang, X. Zhou, E. Zhou, X. Zhang, and J. Sun, “Learning delicate local representations for multi-person pose estimation,” *CoRR*, vol. abs/2003.04030, 2020. [Online]. Available: https://arxiv.org/abs/2003.04030

[14] Y. Li, S. Zhang, Z. Wang, S. Yang, W. Yang, S. Xia, and E. Zhou, “Tokenpose: Learning keypoint tokens for human pose estimation,” *CoRR*, vol. abs/2104.03516, 2021. [Online]. Available: https://arxiv.org/abs/2104.03516

[15] K. Li, S. Wang, X. Zhang, Y. Xu, W. Xu, and Z. Tu, “Pose recognition with cascade transformers,” *CoRR*, vol. abs/2104.06976, 2021. [Online]. Available: https://arxiv.org/abs/2104.06976

[16] Y. Yuan, R. Fu, L. Huang, W. Lin, C. Zhang, X. Chen, and J. Wang, “Hrformer: High-resolution transformer for dense prediction,” *CoRR*, vol. abs/2110.09408, 2021. [Online]. Available: https://arxiv.org/abs/2110.09408

[17] Y. Xu, J. Zhang, Q. Zhang, and D. Tao, “Vitpose: Simple vision transformer baselines for human pose estimation,” 2022. [Online]. Available: https://arxiv.org/abs/2204.12484

[18] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Doll’a r, and C. L. Zitnick, “Microsoft COCO: common objects in context,” *CoRR*, vol. abs/1405.0312, 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

[19] S. Ruder, “An overview of gradient descent optimization algorithms,” *CoRR*, vol. abs/1609.04747, 2016. [Online]. Available: http://arxiv.org/abs/1609.04747

[20] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014. [Online]. Available: https://arxiv.org/abs/1412.6980