Protocol 24: Pose Estimation Refinement

05.07.2019

1. Introduction

As mentioned in the previous protocols, data quality plays an important role in accurate lameness detection. In order to improve the result, the data from pose estimation should be refined. The pose is predicted using DeepLabCut [1], which is a frame-based pose estimator that predicts the pose without considering temporal information. The prediction errors can be divided into four types [2]:

- **Joint transposition**: Similar joints are swapped, usually in asymmetrical motion.

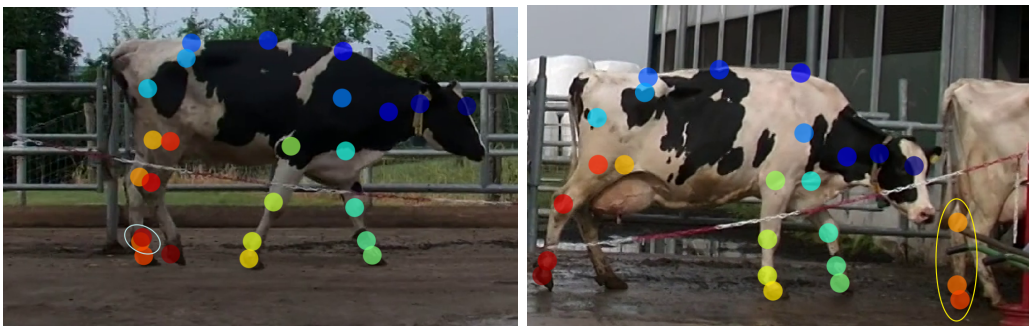

Figure 1: Examples of joint transposition: Joint swap of fetlock joint on the same cow (left); joint swap of rear-right limb joints on two different cows (right).

- **Outlier**: False prediction when the residual magnitude is large. This can be corrected by robust interpolation.

Figure 2: Outlier errors of the fetlock and hoof of rear-right limb.

- **False negative**: Ground truth location not estimated by the estimator.
- **Location jitter**: Joint-specific variance

2. Literature Review

(1) Learning to Refine Human Pose Estimation [3]
PoseRefiner has an architecture of ResNet101 network, accepting an RGB image I and an initial estimate of human body pose P_{input} as input. P_{input} has n binary channel, where n is the number of joints. The network learns to predict the likelihood heatmaps of each joint and offset vectors to recover

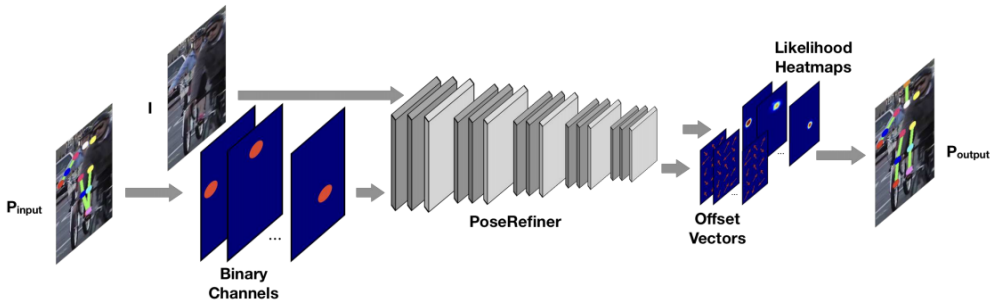

Figure 3: Workflow of PoseRefiner [3].

from downscaled heatmaps to ground truth joint locations. The output is the refined pose prediction P_{output}. PoseRefiner can be applied on top of existing human pose estimator as a post-processing step.

(2) Global Pose Refinement using Bidirectional Long-Short Term Memory [4]
The authors proposed a bidirectional LSTM framework for pose refinement of multiple humans. The key concept is to utilize temporal consistencies of human body shapes between subsequent frames. The LSTM was added on top of a CNN-based pose estimator called OpenPose [5]. The estimated shapes of humans from OpenPose are normalized and passed to the LSTM that encodes the temporal information of shapes between frames. The output is then de-normalized, such that the pose is refined in original resolution.

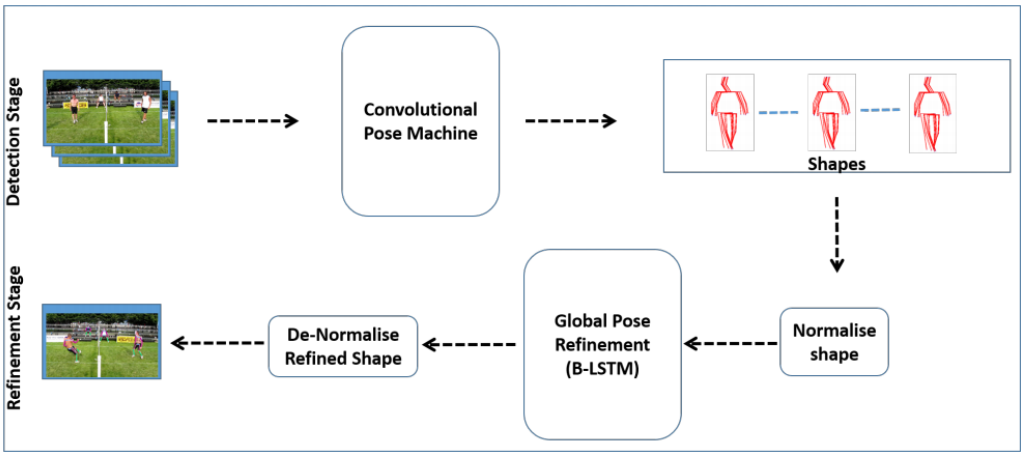

Figure 4: Framework of pose refinement using bidirectional LSTM for [4].

(3) Kinematic Pose Rectification for Performance Analysis and Retrieval in Sports [6]
A three-stage pipeline for pose recitification was proposed in [6]. The first stage is to corrects joint swaps by minimizing the joint velocities and joint accelerations (joint-kinematic based optimization), the second stage applies windowed least squares robust regression to identify outliers and replace them with motion-consistent values; the final stage is adaptive filtering for reducing the variance of joint predictions.

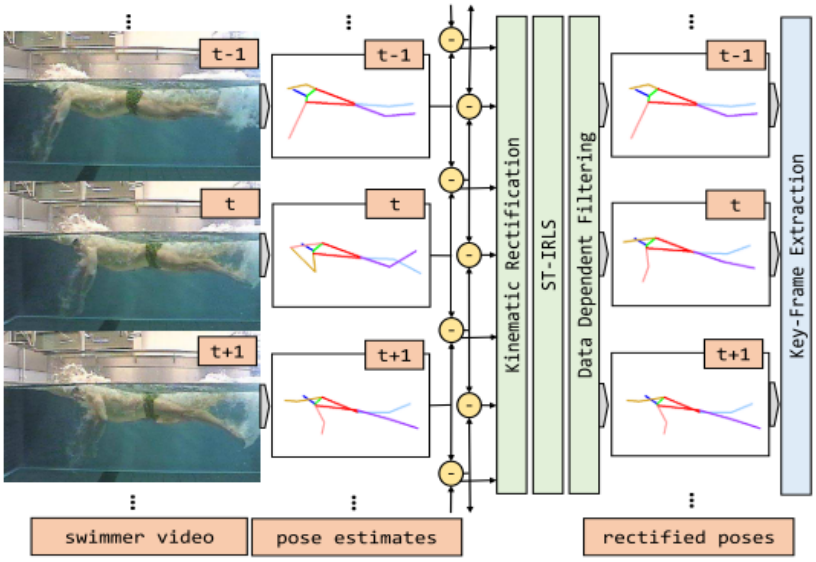

Figure 5: Kinematic rectification pipeline (green blocks) improves joint localization by enforcing temporal consistency between consecutive pose estimates [6].

(4) PoseFix: Model-agnostic General Human Pose Refinement Network [7]
As the authors mentioned, conventional approach usually estimates and refines pose in an end-to-end manner with multiple stages, and it is highly dependent on the estimation model. To circumvent the model-dependent issue, they used error statistics as prior information to generate synthesized poses for model training. PoseFix contains a ResNet architecture with upsampling layers at the end.

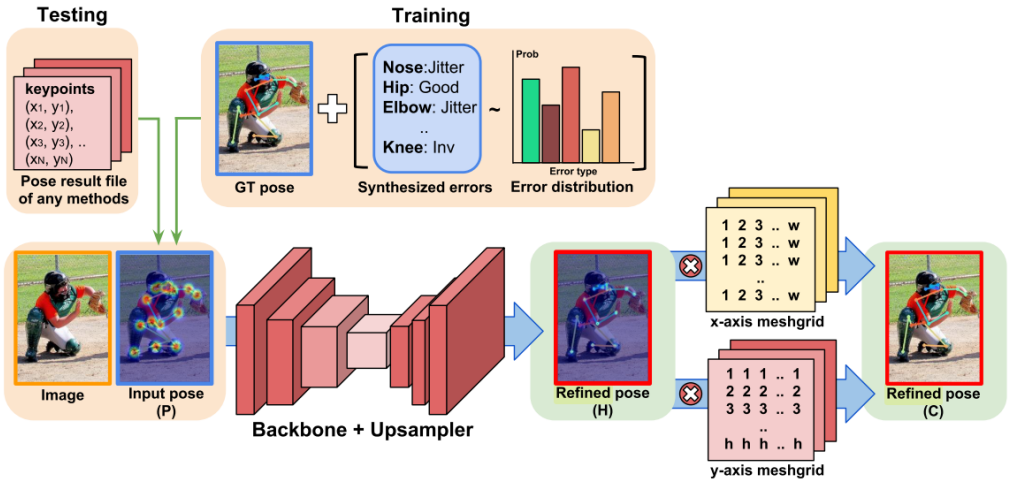

Figure 6: Pipeline of the PoseFix [7].

(5) Refining by Filtering [8, 9]

In addition to the aforementioned approaches using networks, there are some work that applied filtering as a pre-processing step of action recognition or activity prediction. [8] utilized a Savitzky-Golay smoothing filter to improve the signal to noise ratio of raw joint data, while [9] used Extended Kalman Filters (EKFs) to track the joint positions.

3. Pose Refinement

Currently, the poses of cows are refined by re-training the DeepLabCut by taking the frames with obvious estimation errors and refining the keypoints manually. The re-annotated frames are used as additional training samples of DeepLabCut.

Figure 7: Pose estimation errors usually occurs in poor light conditions, when there are occlusions or similar objects.

Bibliography

[1] A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, and M. Bethge, "Deeplabcut: markerless pose estimation of user-defined body parts with deep learning," tech. rep., Nature Publishing Group, 2018.

[2] D. Zecha, M. Einfalt, and R. Lienhart, "Refining joint locations for human pose tracking in sports videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0–0, 2019.

[3] M. Fieraru, A. Khoreva, L. Pishchulin, and B. Schiele, "Learning to refine human pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 205–214, 2018.

[4] I. Radwan, A. Asthana, and R. Goecke, "Global pose refinement using bidirectional long-short term memory," in *ICCV*, 2017.

[5] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: realtime multi-person 2d pose estimation using part affinity fields," *arXiv preprint arXiv:1812.08008*, 2018.

[6] D. Zecha, M. Einfalt, C. Eggert, and R. Lienhart, "Kinematic pose rectification for performance analysis and retrieval in sports," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1791–1799, 2018.

[7] G. Moon, J. Yong Chang, and K. Mu Lee, "Posefix: Model-agnostic general human pose refinement network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7773–7781, 2019.

[8] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1110–1118, 2015.

[9] B. Reily, F. Han, L. E. Parker, and H. Zhang, "Skeleton-based bio-inspired human activity prediction for real-time human–robot interaction," *Autonomous Robots*, vol. 42, no. 6, pp. 1281–1298, 2018.