Hybrid Analytical and Neural IK for Human Pose and Shape Estimation

Sihui Wang
Simon Fraser University
swa279@sfu.ca

1. Introduction

Recovering 3D human poses and shapes from one monocular RGB image has a wide range of applications, however, it is challenging because it is a fundamentally ill-posed problem. A number of methods have been proposed to tackle this problem, which can be roughly categorized into 2 classes: the *model-based* methods and the *3D keypoint estimation-based* methods.

The *model-based* methods assume that human body meshes can be generated from a few parameters, such as the shape parameters and relative rotations of body parts. Researchers have proposed optimization-based approaches and learning-based approaches to estimate the body parameters. While optimization-based approach can reduce the misalignment between the 2D projection of the model and the image, it is a non-convex, iterative fitting process which is time-consuming and sensitive to the initialization. The learning-based method seems to be able to circumvent the iterative process during inference, however the parameter space is abstract and it is difficult to learn the regression function from the images.

Due to the limitations of the model-based methods, researchers resort to the *3D keypoint estimation-based* methods, which typically adopts volumetric heatmap as the target representation to learn 3D joint locations. While 3D keypoint estimation-based methods have achieved impressive performance, one drawback of such method is that it might predict unrealistic body structures (for example, left-right asymmetry and abnormal proportions of limbs) due to the lack of explicit modelling of human bodies.

The collaboration between 3D joints and body mesh estimation is a promising direction, as accurate 3D joints help improve body mesh estimation, and parametric body models in turn eliminate the unrealistic body structures generated by the 3D keypoint estimation-based methods.

In this project, we propose to replicate the paper, "*HybrIK: A Hybrid Analytical-Neural Inverse Kinematics Solution for 3D Human Pose and Shape Estimation*", which aims at bridging the gap between 3D keypoint estimation and body mesh estimation. As inverse kinematics (IK) is an ill-posed problem which doesn't have a unique solution, HybrIK proposes to perform hybrid estimations for the relative rotation of skeleton parts. In HybrIK, relative rotations are decomposed into *twist* and *swing*. While *swing* rotation is predicted by neural network via visual clues, *twist* rotation is calculated analytically.

The paper of *HybrIK* proposed 2 methods for the computation of *twist* rotation: *naïve HybrIK* and *adaptive HybrIK*. It is shown that HybrIK can generate relative rotations which are naturally aligned with the 3D skeleton, and the differentiable nature of HybrIK allows us to jointly train accurate 3D joint locations and convert them to human body mesh estimations in an end-to-end manner. However, *adaptive HybrIK* is heuristic, and it is unclear if it is the best option for minimizing the reconstruction errors. In this project, we propose "λ-HybrIK", which is an *interpolation* between *naïve HybrIK* and *adaptive HybrIK*. We will evaluate "λ-HybrIK" to see what λ produces pose estimation with the minimized reconstruction error.

2. Related Work

2.1. 3D Keypoint Estimation

3D human pose estimation can be formulated as the problem of locating the 3D joints of the human body, which can be solved by two-stage approaches: 2D pose is estimated first and then lifted to 3D joint locations. While such methods achieved impressive performance, it lacks an explicit model for human structural information and can't guarantee that the output skeleton is realistic. HybrIK, on the other hands, combines the advantages of 3D skeleton and parametric models to predict human poses and shapes that are both accurate and realistic.

2.2. Model-based 3D Pose and Shape Estimation

Model-based works capture the statistics prior of body shapes and directly predict body meshes. In recent years, learning-based frameworks are widely adopted for model-based methods. However, the challenge is that shape parameters and relative rotations are hard to predict from RGB images, and intermediate representations are required to alleviate such problems. HybrIK addresses this problem by a transformation from the pixel-aligned 3D joints to the relative rotations.

2.3. Body-part Rotation in Pose Estimation

Previous works proposed to use neural networks to predict the rotation angle, quaternions, rotation matrices, or rotation Euler angles of each body joint. However, these methods either encounter hard-to-learn problems, or require additional fitting process. In HybrIK, the body-part rotation is recovered from 3D joint locations in a direct and accurate manner.

2.4. Inverse Kinematics Process

There are different solutions for the inverse kinematics. The numerical solutions are simple to implement, but they are time-consuming because of the iterative optimization processes; In special cases, there are analytical solutions, and researchers have developed methods to combine analytical and numerical methods. In recently years, researchers are also interested in using neural networks to solve the IK problem. HybrIK combines the interpretability of analytical solutions and the flexibility of neural networks by proposing a feed-forward hybrid IK algorithm with the decomposition of *twist* and *swing*. Compared with previous analytical solutions which are only applicable to specific joint linkage, HybrIK is flexible and can generalize well to the entire skeleton in a differentiable manner.

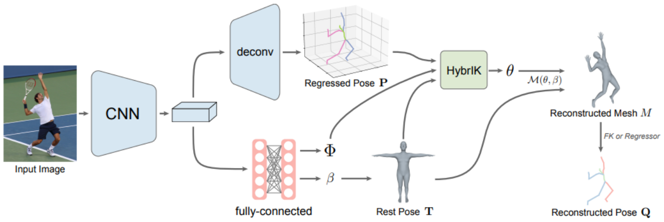


Figure 1. **Overview of the Learning Framework**

3. Algorithm

3.1. Learning Framework

The input image is first going through a CNN-based backbone network. Then, the deconvolution layer generates a 3D heatmap, which is used to regress 3D joints P. Meanwhile, the shape parameters β and the twist angle Φ are learned from the fully-connected layers. Using the shape parameters β, we can obtain the rest pose T of the human body mesh by the SMPL model. Then, taking P, T, Φ as inputs, HybrIK solves the relative rotations and obtains the pose parameters θ. Finally, with the function $\mathbf{M}(\theta, \beta)$ provided by the SMPL model, we can obtain the body mesh M and the reconstructed pose Q. A diagram of the learning framework is shown in figure 1.

3.2. Inverse Kinematic and HybrIK

3.2.1 Recursive IK

Our goal is to find the relative rotation R so that the reconstructed body mesh is aligned with the predicted 3D keypoints given the rest pose template $T = \{t_k\}_{k=1}^{K}$ and the input body joints $P = \{p_k\}_{k=1}^{K}$:

$$R = \text{IK}(P, T) \tag{1}$$

Ideally, the rotations R_k should satisfy the following condition:

$$p_k - p_{pa(k)} = R_k(t_k - t_{pa(k)}) \tag{2}$$

Here, p_k denotes the k-th joint of the input pose, and t_k denotes the k-th joint of the rest pose template. $pa(k)$ is the parent's index of the k-th joint, and $R_k \in SO(3)$ is the global rotation of the k-th joint with respect to the canonical rest pose space.

The global rotations can be calculated recursively from $R_{pa(k)}$ to R_k:

$$R_k = R_{pa(k)} R_{pa(k),k} \tag{3}$$

where $R_{pa(k),k}$ is the relative rotation of joint k with respect to joint $pa(k)$.

3.2.2 Twist-and-Swing Decomposition

In HybrIK, we decompose the relative pose $R_{pa(k),k}$ into a *twist* rotation $R^{tw}_{pa(k),k}$ and a *swing* rotation $R^{sw}_{pa(k),k}$. Given the template body-part vector $\vec{t}$, the target vector $\vec{p}$, and the *twist* angle estimated by a neural network, the solution process of $R_{pa(k),k}$ can be formulated as:

$$\begin{aligned} R_{pa(k),k} &= \mathcal{D}(\vec{p}, \vec{t}, \phi) = \mathcal{D}^{sw}(\vec{p}, \vec{t}) \cdot \mathcal{D}^{tw}(\vec{t}, \phi) \\ &= R^{sw}_{pa(k),k} \cdot R^{tw}_{pa(k),k} \end{aligned} \tag{4}$$

The *swing* rotation is in the plane of $\vec{t}$ and $\vec{p}$. The rotation is along the axis $\vec{n} = \frac{\vec{t} \times \vec{p}}{||\vec{t} \times \vec{p}||}$, and the *swing* angle α satisfies $cos\alpha = \frac{\vec{t} \cdot \vec{p}}{||\vec{t}|| \cdot ||\vec{p}||}$ and $sin\alpha = \frac{||\vec{t} \times \vec{p}||}{||\vec{t}|| \cdot ||\vec{p}||}$.

Therefore, the closed-form solution of *swing* rotation $R^{sw}_{pa(k),k}$ can be obtained from *Rodrigues formula*:

$$\begin{aligned} R^{sw}_{pa(k),k} &= \mathcal{D}^{sw}(\vec{p}, \vec{t}) \\ &= \mathcal{I} + sin\alpha [\vec{n}]_{\times} + (1 - cos\alpha)[\vec{n}]^2_{\times} \end{aligned} \tag{5}$$

The *twist* rotation is the rotation along $\vec{t}$. Given that $\vec{t}$ is the axis and ϕ is the rotation angle, we can obtain the *twist* rotation by *Rodrigues formula*:

$$\begin{aligned} R^{tw}_{pa(k),k} &= \mathcal{D}^{tw}(\vec{t}, \phi) \\ &= \mathcal{I} + \frac{sin\phi}{||\vec{t}||}[\vec{t}]_{\times} + \frac{(1 - cos\phi)}{||\vec{t}||^2}[\vec{t}]^2_{\times} \end{aligned} \tag{6}$$

3.2.3 Naive HybrIK

First, we need to obtain the global root rotation R_0, which is acquired from the locations of *spine*, *left hip*, *right hip* by Singular Value Decomposition (SVD). Then, we recursively solve each joint k's relative rotation with respect to its parent joint, $R_{pa(k),k}$, which is done by the following calculation:

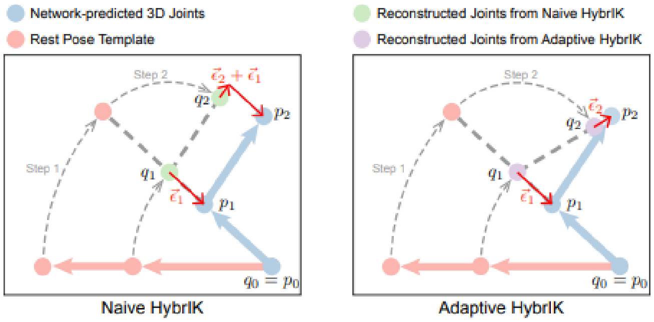


Figure 2. **Illustration of Reconstruction Error** In the second step, *naïve hybrIK* takes $p_2 - p_1$ as the target direction, whereas *adaptive hybrIK* takes $p_2 - q_1$ as the target direction. *Naïve hybrIK* leads to the accumulation of error $\vec{\epsilon}_1 + \vec{\epsilon}_2$ in the second step, whereas *adaptive hybrIK* might reduce the error to only $\vec{\epsilon}_2$.

$$\hat{p}_k = R^{-1}_{pa(k)}(p_k - p_{pa(k)}) \tag{7}$$

$$\hat{t}_k = t_k - t_{pa(k)} \tag{8}$$

$$R_{pa(k),k} = \mathcal{D}(\hat{p}_k, \hat{t}_k, \phi_k) \tag{9}$$

One drawback of *naïve HybrIK* is that it always assumes that:

$$||p_k - p_{pa(k)}|| = ||t_k - t_{pa(k)}|| \tag{10}$$

However, in reality, the keypoints predicted by the neural networks can be inconsistent with the template. In such condition, we have:

$$p_k - p_{pa(k)} = R_k(t_k - t_{pa(k)}) + \vec{\epsilon_k} \tag{11}$$

where $\vec{\epsilon_k}$ denotes the error for the k-th joint. The authors of *HybrIK* argues that *naïve HybrIK* leads to accumulation of reconstruction errors along the steps (figure 2), which is why they proposed *adaptive HybrIK* as a mitigation.

3.2.4 Adaptive HybrIK

In *adaptive HybrIK*, the authors propose to adaptively update the target vector $\vec{p}_k$ by the newly reconstructed parent joints $q_{pa(k)}$. That is, to compute $\vec{p}_k$, we first compute the position of the reconstructed parent joint:

$$q_{pa(k)} = R_{pa(k)}(t_{pa(k)} - t_{pa^2(k)}) + q_{pa^2(k)} \tag{12}$$

Then, we replace $p_{pa(k)}$ in equation (7) by the newly reconstructed parent joint $q_{pa(k)}$ and obtain:

$$\hat{p}_k = R^{-1}_{pa(k)}(p_k - q_{pa(k)}) \tag{13}$$

Then, we follow equation (8) and (9) to recursively solve $R_{pa(k),k}$. The motivation of *adaptive HybrIK* is to reduce the reconstruction errors of the joints.

3.2.5 Our Method: λ-HybrIK

Adaptive HybrIK is a heuristic technique for reduction of reconstruction error. However, we are not sure if 1) *adaptive HybrIK* is effective in error reduction; and 2) if there are better options. In this project, we propose "λ-HybrIK", which is an *interpolation* between *naïve HybrIK* and *adaptive HybrIK*.

```
Algorithm: $\lambda$-HybrIK
Input: $P, T, \Phi$
Output: R
Determine $R_0$;
for $k$ along the kinematic tree do
    $q_{pa(k)} \leftarrow R_{pa(k)}(t_{pa(k)} - t_{pa^2(k)}) + q_{pa^2(k)}$;
    $r_{pa(k)} \leftarrow \lambda \cdot q_{pa(k)} + (1 - \lambda) \cdot p_{pa(k)}$;
    $\vec{p}_k \leftarrow R^{-1}_{pa(k)}(p_k - r_{pa(k)})$;
    $\vec{t}_k \leftarrow t_k - t_{pa(k)}$;
    $R^{sw}_{pa(k),k} \leftarrow \mathcal{D}^{sw}(\vec{p}_k, \vec{t}_k)$;
    $R^{tw}_{pa(k),k} \leftarrow \mathcal{D}^{tw}(\vec{t}_k, \phi_k)$;
    $R_{pa(k),k} \leftarrow R^{sw}_{pa(k),k} R^{tw}_{pa(k),k}$;
```

When $\lambda = 0$, λ-HybrIK degenerates to *naïve HybrIK*; when $\lambda = 1$, λ-HybrIK becomes *adaptive HybrIK*. In the next section, we will evaluate reconstruction error of λ-HybrIK with different λs.

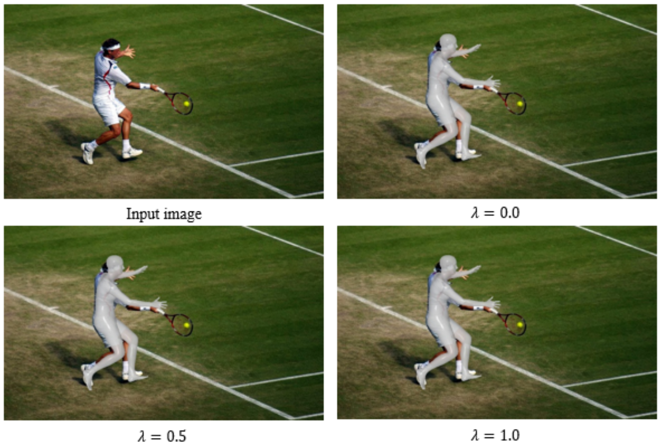
4. Experiments and Results

4.1. Dataset

We use **MPI-INF-3DHP**'s test set for evaluation.

4.2. Model

We use **HybrIK (HRNet-W48)** pre-trained model for evaluation.



Input image

$\lambda = 0.0$

$\lambda = 0.5$

$\lambda = 1.0$

Figure 3. **Qualitative Results: Reconstruction of Body Meshes** for $\lambda = 0.0$, $\lambda = 0.5$, and $\lambda = 1.0$

4.3. Qualitative Results

We generated body meshes from one monocular RGB image for $\lambda = 0.0, \lambda = 0.5$, and $\lambda = 1.0$. The qualitative results are shown in figure 3.

From figure 3 we can see that the visual difference of the body meshes is negligible, which seems to suggest that different choices of λ doesn't affect the reconstruction outcome much.

We did experiments on a number of images and videos. The full outcome can be found in our google drive[1].

4.4. Quantitative Results

We evaluated HybrIK on **MPI-INF-3DHP**'s test set with $\lambda = 0.0, 0.2, 0.4, 0.6, 0.8, 1.0$. To measure HybrIK's pose estimation accuracy, we used the following evaludation metrics: *procrustes aligned mean per joint position error* (**PA-MPJPE**), *mean per joint position error* (**MPJPE**), and *percentage of correct keypoints* (**PCK**).

λ	PA-MPJPE(↓)	MPJPE(↓)	PCK(↑)
0.0	62.7749	92.4795	87.0496
0.2	**62.7587**	92.4571	**87.0599**
0.4	62.7600	92.4596	87.0517
0.6	62.7625	92.4568	87.0297
0.8	62.7683	92.4486	87.0299
1.0	62.7616	**92.4402**	87.0270

[1] https://drive.google.com/drive/folders/1K6U3LFzHFFvpT908A8deQIGXEGuqQJVV?usp=sharing

While the original paper reported small difference between *HybrIK-Naïve* and *HybrIK-Adaptive* (*HybrIK-Adaptive* improves PCK by 0.3 and reduces MPJPE by 0.5), we found even smaller difference among all variants of *HybrIK*. *HybrIK* with $\lambda = 1.0$ seems to produce the most accurate results in terms of MPJPE. However, with respect to PA-MPJPE and PCK, *HybrIK* with $lambda = 0.2$ seems to produce more accurate outcomes.

4.5. Conclusion

In this project, we obtained the following preliminary results:

1. The visual difference of body mesh reconstruction produced by different variants of *HybrIK* is very small. This is consistent with our quantitative results and the quantitative results from the original paper (the use of *naïve HybrIK* or *adaptive HybrIK* only causes a very small variation in reconstruction error).

2. We find that different λs only lead to very tiny variation in reconstruction errors. $\lambda = 0.0$ (*naïve HybrIK*) and $\lambda = 1.0$ (*adaptive HybrIK*) are not necessarily the optimal choices, as we might obtain minimal reconstruction error when λ is between 0.0 and 1.0.

5. Acknowledgements

Our implementation of λ-HybrIK is based on the authors' implementation of *naïve hybrIK* and *adaptive hybrIK*. We acknowledge the use of the authors' code from https://github.com/Jeff-sjtu/HybrIK.

6. Key References

[1] Li, Jiefeng, Chao Xu, Zhicun Chen, Siyuan Bian, Lixin Yang, and Cewu Lu. "Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3383-3393. 2021.

[2] Bogo, Federica, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. "Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image." In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14, pp. 561-578. Springer International Publishing, 2016.