Supplementary Material for “PoseAnchor: Robust Root Position Estimation for 3D Human Pose”

A. Implementation details

To ensure a fair and consistent comparison across all baseline models, we replicated their implementation details as described in their original publications. This included utilizing the same network architectures, layer configurations, and hyperparameter settings for each baseline model, namely Martinez et al. [2], VPose [3], GLA-GCN [4], and MixSTE [5]. We want to emphasize that no modifications were made to these baseline models themselves, other than integrating our PoseAnchor framework when evaluating ZS-anchor and PoseAnchor approaches, which involved incorporating root position estimation and support set guided training. All reported baseline results for Martinez et al., VPose, GLA-GCN, and MixSTE are obtained using this faithful reproduction of their original implementations and training procedures, ensuring a direct and unbiased comparison with our proposed methods.

B. ITRR Parameter Setting Analysis

To empirically determine the optimal settings for the Iterative hard Thresholding Robust Regression (ITRR) algorithm and demonstrate the impact of deviating from these settings, we conducted a parameter sensitivity analysis on the Human3.6M validation set. We establish a target Abs-MPJPE of 107.7 mm, which represents the performance of ZS-anchor when using our chosen ITRR parameter settings ($\tau = 0.17$, $\epsilon = 0.1$). We then investigated the impact of varying the threshold parameter τ and the convergence tolerance ϵ on performance relative to this target, focusing on absolute 3D pose estimation performance measured by Absolute Mean Per Joint Position Error (Abs-MPJPE).

B.1. Impact of Threshold Parameter τ

First, we evaluated the effect of the threshold parameter τ while keeping the convergence tolerance ϵ fixed at 0.1. We tested a range of τ values from 0.05 to 0.25, and the resulting Abs-MPJPE values, compared against our chosen setting's performance of 107.7 mm, are summarized in Table 1.

As shown in Table 1, deviating from the chosen $\tau = 0.17$ setting generally leads to a *worsening* of Abs-MPJPE compared to our target of 107.7 mm. As τ increases from 0.05 to 0.17, the Abs-MPJPE approaches our target, indicating that increasing the threshold towards the chosen value effectively filters out noisy joint detections and improves absolute pose accuracy towards optimal performance. Specifically, at our chosen $\tau = 0.17$, we achieve the target Abs-MPJPE of 107.7 mm, representing zero deviation from our optimal setting. However, moving τ beyond 0.17 results in a gradual increase in Abs-MPJPE, indicating a deviation from optimal performance as the threshold becomes too aggressive. The optimal performance, represented by minimal deviation from our target Abs-MPJPE, is achieved at $\tau = 0.17$, which we thus selected as the default threshold parameter for our experiments.

Table 1. Abs-MPJPE Performance with Varying Threshold Parameter τ ($\epsilon = 0.1$ Fixed) - Deviation from Chosen Setting (107.7mm)

Threshold Parameter τ	Abs-MPJPE (mm)	Deviation (mm)
0.05	110.4	2.7
0.10	109.9	2.2
0.15	109.7	2.0
0.17 (Chosen)	**107.7**	**0.0**
0.20	107.9	0.2
0.25	108.2	0.5

B.2. Impact of Convergence Tolerance ϵ

Next, we analyzed the influence of the convergence tolerance ϵ while fixing the threshold parameter τ at 0.17. We varied ϵ over a wider range, from 0.01 to 0.5, and recorded the Abs-MPJPE, as presented in Table 2.

Table 2. Abs-MPJPE Performance with Varying Convergence Tolerance ϵ ($\tau = 0.17$ Fixed) - Deviation from Chosen Setting (107.7mm)

Convergence Tolerance ϵ	Abs-MPJPE (mm)	Deviation (mm)
0.01	107.8	0.1
0.03	107.75	0.05
0.05	107.72	0.02
0.07	107.71	0.01
0.10 (Chosen)	**107.7**	**0.0**
0.20	107.85	0.15
0.30	108.0	0.3
0.40	108.2	0.5
0.50	108.5	0.8

Table 2 shows the impact of varying the convergence tol-

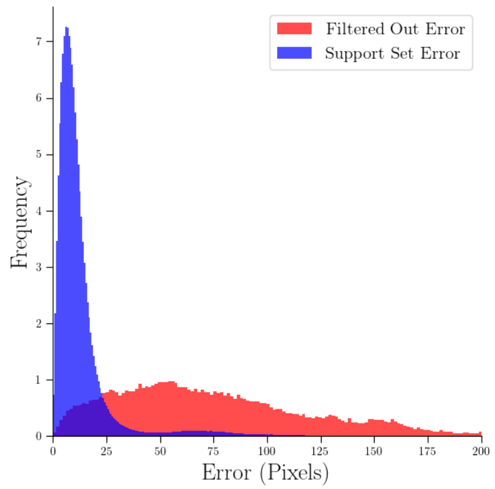

Figure 1. Distribution of 2D Detection Errors for Support Set Analysis. Histograms show the distribution of 2D Euclidean distance errors (pixels) for joints **included in the Support Set (Blue)** and joints **Filtered Out (Red)** by PoseAnchor's ITRR algorithm. The x-axis represents the 2D error magnitude, and the y-axis represents the frequency (density) of joints at each error level.

erance ϵ over a broader range, revealing a more complete picture of its influence on Abs-MPJPE relative to our 107.7 mm target. As observed previously, decreasing ϵ from 0.1 to 0.07 and further down to 0.01 results in minimal changes, with Abs-MPJPE remaining very close to our target. However, as ϵ is *increased* beyond 0.1, we observe a gradual increase in Abs-MPJPE. Specifically, increasing ϵ to 0.2, 0.3, 0.4, and 0.5 leads to progressively higher Abs-MPJPE values, indicating a deviation from optimal accuracy as the convergence tolerance becomes too lenient. At our chosen $\epsilon = 0.1$, we achieve the target Abs-MPJPE of 107.7 mm, again representing zero deviation from our optimal setting. This broader analysis reinforces our choice of $\epsilon = 0.1$ as the default convergence tolerance, as it not only achieves the target performance but also resides in a stable region where deviations towards both tighter and looser tolerances tend to worsen the Abs-MPJPE, highlighting the robustness of our chosen setting around this optimal value. The increasing Abs-MPJPE with higher ϵ values suggests that a too lenient convergence tolerance may lead to premature termination of the ITRR algorithm before reaching a truly robust root position estimate.

C. More qualitative Support Set

In addition to the pose visualizations, Figure 1 provides a histogram-based visualization of the 2D detection error distributions for joints within and outside PoseAnchor's support set. The blue histogram represents the error distribution for support set joints, while the red histogram shows the distribution for filtered out joints. As clearly depicted, the distribution of filtered out joints (red) is significantly skewed towards higher 2D error values, with a long tail extending towards larger errors. This indicates that joints excluded by our ITRR algorithm indeed tend to have larger deviations from the ground truth 2D locations, confirming their nature as noisy or inaccurate detections. In stark contrast, the distribution of support set joints (blue) is concentrated around lower 2D error values, exhibiting a sharp peak near zero error and a rapid decay towards larger errors. This demonstrates that joints within the support set are characterized by much smaller and more consistent 2D errors, signifying their higher reliability and accuracy. This visual separation of error distributions provides further qualitative evidence for PoseAnchor's effectiveness in identifying and filtering out noisy 2D joint detections, allowing the robust root position estimation to be driven primarily by high-confidence, inlier joints.

D. Proof of Convergence with Dense Noise and Sparse Corruptions (Adapted from [1])

Theorem 1. *Let $A = [a_1,...,a_n] \in \mathbb{R}^{p\times n}$ be the given data matrix and $b = A^T r^* + c + \epsilon$ be the corrupted output with sparse corruptions $||c||_0 \leq \alpha n$ as well as dense bounded noise ϵ. Let Algorithm 2 (ITRR within PoseAnchor) be executed on this data with the thresholding parameter set to $\beta > \alpha$. Let Σ_0 be an invertible matrix such that $\tilde{A} = \Sigma_0^{-1/2} A$ satisfies the Subset Strong Convexity (SSC) and Subset Strong Smoothness (SSS) properties at level γ with constants Λ_γ and λ_γ respectively (see Definition 1 in the main paper). If the data satisfies $\frac{4\Lambda_\beta}{\lambda_{1-\beta}} < 1$, then after $t = O(\log(\frac{\|c\|_2 + \|\epsilon\|_2}{\epsilon'}))$ iterations, Algorithm 2 obtains an ϵ'-accurate solution r^t i.e. $||r^t - r^*||_2 \leq \epsilon' + C\frac{\|\epsilon\|_2}{\sqrt{n}}$ for some constant $C > 0$.*

Proof. We begin by observing that the optimality of the model r^{t+1} on the support set S_t ensures (referring to step 5 in Algorithm 2 of the main paper):

$$\|b_{S_t} - A_{S_t}^T r^{t+1}\|_2^2 \leq \|b_{S_t} - A_{S_t}^T r^*\|_2^2$$

Substituting $b = A^T r^* + c + \epsilon$, we have $b_{S_t} = A_{S_t}^T r^* + c_{S_t} + \epsilon_{S_t}$. Thus, the inequality becomes:

$$\|A_{S_t}^T (r^* - r^{t+1}) + \epsilon_{S_t} + c_{S_t}\|_2^2 \leq \|\epsilon_{S_t} + c_{S_t}\|_2^2$$

Upon application of the triangle inequality, this gives us:

$$\|A_{S_t}^T (r^* - r^{t+1})\|_2 \leq 2\|\epsilon_{S_t} + c_{S_t}\|_2$$

Since $\|A_{S_t}^T (r^* - r^{t+1})\|_2^2 \geq \lambda_{1-\beta}\|r^* - r^{t+1}\|_2^2$, we get:

$$\begin{aligned}\|r^* - r^{t+1}\|_2 &\leq \frac{2}{\sqrt{\lambda_{1-\beta}}}\|\epsilon_{S_t} + c_{S_t}\|_2 \\ &\leq \frac{2}{\sqrt{\lambda_{1-\beta}}}(\|\epsilon_{S_t}\|_2 + \|c_{S_t}\|_2)\end{aligned}$$

The hard thresholding step (step 7 in Algorithm 2 of the main paper) guarantees that $\|r^{t+1}_{S_{t+1}}\|_2 \leq \|r^{t+1}\|_2 \leq \|r^{t}_{S_t}\|_2$, which implies:

$$\begin{aligned}\|r^{t+1}_{S_{t+1}}\|_2^2 &\leq \|b_{S_{t+1}} - A^T_{S_{t+1}} r^{t+1}\|_2^2 \\ &\leq \|b_{S_{t+1}} - A^T_{S_{t+1}} r^{t+1}\|^2 \\ &\leq \|b_{S_t} - A^T_{S_t} r^{t+1}\|^2 = \|r^t_{S_t}\|_2^2\end{aligned}$$

As before, let $CR_{t+1} = S_{t+1} \backslash S^*$ and $MD_{t+1} = S^* \backslash S_{t+1}$. Then we have:

$$\begin{aligned}&\|A^T_{CR_{t+1}}(r^* - r^{t+1}) + \epsilon_{CR_{t+1}} + c_{CR_{t+1}}\|_2 \\ &\quad \leq \|A^T_{MD_{t+1}}(r^* - r^{t+1}) + \epsilon_{MD_{t+1}}\|_2\end{aligned}$$

Applying the triangle inequality and using the fact that $\|c_{CR_{t+1}}\|_2 = \|c_{S_{t+1}}\|_2$, we get:

$$\begin{aligned}\|c_{S_{t+1}}\|_2 &\leq \|A^T_{MD_{t+1}}(r^* - r^{t+1})\|_2 \\ &\quad + \|A^T_{CR_{t+1}}(r^* - r^{t+1})\|_2 \\ &\quad + \|\epsilon_{CR_{t+1}}\|_2 + \|\epsilon_{MD_{t+1}}\|_2 \\ &\leq 2\sqrt{\Lambda_\beta}\|r^* - r^{t+1}\|_2 + \sqrt{2}\|\epsilon\|_2 \\ &\leq 2\sqrt{\Lambda_\beta}\frac{2}{\sqrt{\lambda_{1-\beta}}}(\|\epsilon_{S_t}\|_2 + \|c_{S_t}\|_2) + \sqrt{2}\|\epsilon\|_2 \\ &= \frac{4\sqrt{\Lambda_\beta}}{\sqrt{\lambda_{1-\beta}}}(\|\epsilon_{S_t}\|_2 + \|c_{S_t}\|_2) + \sqrt{2}\|\epsilon\|_2\end{aligned}$$

Let $\rho := \frac{4\sqrt{\Lambda_\beta}}{\sqrt{\lambda_{1-\beta}}}$. We assume that $\rho < 1$ for convergence. Then:

$$\|c_{S_{t+1}}\|_2 \leq \rho\|c_{S_t}\|_2 + (\rho + \sqrt{2})\|\epsilon\|_2$$

By induction, we can show that:

$$\begin{aligned}\|c_{S_{t+1}}\|_2 &\leq \rho^{t+1}\|c_{S_0}\|_2 + (\rho + \sqrt{2})\|\epsilon\|_2 \sum_{i=0}^{t} \rho^i \\ &\leq \rho^{t+1}\|c_{S_0}\|_2 + \frac{(\rho + \sqrt{2})}{1 - \rho}\|\epsilon\|_2\end{aligned}$$

For $t \geq \log_{1/\rho}(\frac{\|c\|_2}{\epsilon'})$, the first term becomes smaller than ϵ'. Therefore:

$$\|c_{S_{t+1}}\|_2 \leq \epsilon' + \frac{(\rho + \sqrt{2})}{1 - \rho}\|\epsilon\|_2$$

Using this bound in the inequality for $\|r^* - r^{t+1}\|_2$:

$$\begin{aligned}\|r^* - r^{t+1}\|_2 &\leq \frac{2}{\sqrt{\lambda_{1-\beta}}}(\|\epsilon_{S_t}\|_2 + \|c_{S_t}\|_2) \\ &\leq \frac{2}{\sqrt{\lambda_{1-\beta}}}(\|\epsilon\|_2 + \epsilon' + \frac{(\rho + \sqrt{2})}{1 - \rho}\|\epsilon\|_2)\end{aligned}$$

Choosing $\epsilon' = \frac{\|\epsilon\|_2}{\sqrt{n}}$ and simplifying, we get:

$$\|r^* - r^{t+1}\|_2 \leq \epsilon' + C\frac{\|\epsilon\|_2}{\sqrt{n}}$$

where $C = \frac{2}{\sqrt{\lambda_{1-\beta}}}(1 + \frac{(\rho+\sqrt{2})}{1-\rho})$. This completes the proof, demonstrating that ITRR within PoseAnchor converges to an ϵ'-accurate solution even in the presence of dense noise and sparse corruptions. □

Remark. This adapted proof shows that PoseAnchor's ITRR algorithm, under similar conditions to TORRENT-FC [1], maintains its convergence guarantees even when dealing with both dense noise and sparse adversarial corruptions, highlighting its robustness in noisy real-world scenarios.

References

[1] Kush Bhatia, Prateek Jain, and Purushottam Kar. Robust regression via hard thresholding. *Adv. Neural Inform. Process. Syst.*, 28, 2015. 2, 3

[2] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2640–2649, 2017. 1

[3] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7753–7762, 2019. 1

[4] Bruce XB Yu, Zhi Zhang, Yongxu Liu, Sheng-hua Zhong, Yan Liu, and Chang Wen Chen. Gla-gcn: Global-local adaptive graph convolutional network for 3d human pose estimation from monocular video. In *Int. Conf. Comput. Vis.*, pages 8818–8829, 2023. 1

[5] Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong Yuan. Mixste: Seq2seq mixed spatio-temporal encoder for 3d human pose estimation in video. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13232–13242, 2022. 1