

DexAvatar: 3D Sign Language Reconstruction with Hand and Body Pose Priors

Supplementary Material

Kaustubh Kundu¹, Hrishav Bakul Barua^{1,2}, Lucy Robertson-Bell¹, Zhixi Cai¹, Kalin Stefanov¹

¹Monash University

²TCS Research

{kaustubh.kundu, hrishav.barua, lucy.robertson-bell, zhixi.cai, kalin.stefanov}@monash.edu

1. Background Knowledge

SMPL-X [8] is an advanced parametric human body model, an extension of the original SMPL [7], integrating hand articulations through MANO [9] and facial expressions through FLAME [6]. This enables comprehensive full-body representations that include hand and face dynamics. SMPL-X is defined by a mapping function $M(\theta, \beta, \psi) : \mathbb{R}^{|\theta| \times |\beta| \times |\psi|} \rightarrow \mathbb{R}^{3N}$, parameterized by the pose $\theta \in \mathbb{R}^{3(K+1)}$, where K is the number of body joints in addition to a joint for global rotation. β represents shape coefficients, and ψ are the facial expression coefficients.

The model uses vertex-based linear blend skinning with $N = 10,475$ vertices and $K = 54$ joints, including joints for hands, neck, jaw, and eyeballs. The formulation of SMPL-X is defined as follows:

$$M(\beta, \theta, \psi) = W(T_P(\beta, \theta, \psi), J(\beta), \theta, W), \quad (1)$$

where,

$$T_P(\beta, \theta, \psi) = \bar{T} + B_S(\beta; \mathcal{S}) + B_E(\psi; \mathcal{E}) + B_P(\theta; \mathcal{P}). \quad (2)$$

$B_P(\cdot)$, $B_S(\cdot)$, and $B_E(\cdot)$ in Eq. (2) represent the pose, shape, and expression-dependent corrective blend functions. \mathcal{S} , \mathcal{E} , and \mathcal{P} represent the orthonormal principal components of vertex displacements of shape, pose, and expression blend shape variations. These functions apply vertex displacements to the canonical template mesh \bar{T} based on the pose parameters θ , shape parameters β , and expression parameters ψ . In particular, $B_P(\theta)$ and $B_S(\beta)$ capture non-linear deformations specific to pose and shape variations. After these corrections, the deformed mesh is processed using linear blend skinning, denoted as \mathcal{W} , which rotates the vertices around the joints $J(\beta)$ according to the skeletal kinematics. The final mesh is smoothed using a pre-defined set of blend weights W , resulting in the articulated 3D human body mesh.

2. Range of Motion and Signer Space

This section complements section 3.2.1 of the main paper. We constrain the upper limb using the physiological degrees of freedom (DOFs) [4, 5] of the joints most active in sign language. The shoulder has three DOFs, *i.e.*, flexion/extension, abduction/adduction, and internal/external rotation. The elbow-forearm complex has two DOFs, *i.e.*, humeroulnar flexion/extension and radioulnar pronation/supination. For the wrist, we adopt a three DOF formulation covering forearm pronation/supination, wrist flexion/extension, and radial/ulnar deviation.

Clinical range of motion (ROM) [4, 5] for these DOFs is typically reported as *unsigned* magnitudes in anatomical planes (*e.g.*, wrist flexion 90° and extension 90°). To use these values, we convert them to *signed* bounds in an anatomical Euler [11] convention. For each DOF, we align a local axis with the corresponding motion, adopt the right-hand rule for signs, and verify the orientation on the rig.

We express ROMs as a single signed interval in the aligned Euler convention. For bilateral joints, we mirror the sign across the sagittal plane (including wrist flexion/extension), matching the left/right labeling in Fig. 1. This deterministic normalization yields SMPL-X [8] compatible signed bounds from clinical ROM with a brief visual sanity check.

Having normalized clinical ROM to SMPL-X compatible signed Euler bounds, we further constrain the shoulder using the notion of signer space [2, 12], a torso-centric 3D region where signs are typically produced (see Fig. 3). We model this as a compact volume anchored to the torso, bounded laterally near the shoulders, vertically from the lower chest to the forehead, and in depth slightly in front of the chest.

We restrict shoulder motion to be consistent with a torso-anchored, ground-parallel signer space envelope (see Fig. 3). Accordingly, we cap horizontal abduction by disallowing motion behind the torso (see Fig. 2a) while permitting substantial horizontal adduction to support cross-body

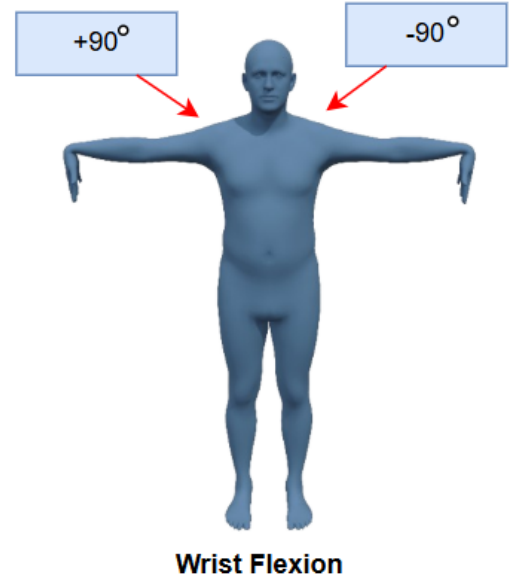
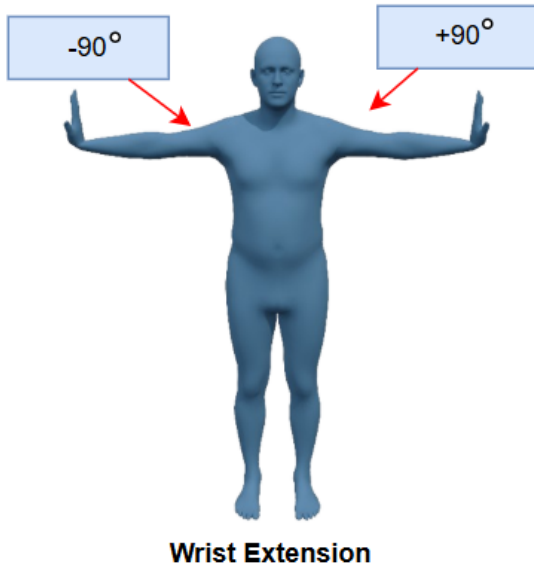
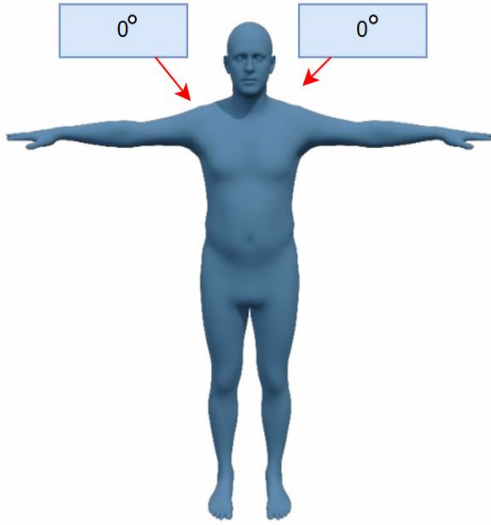
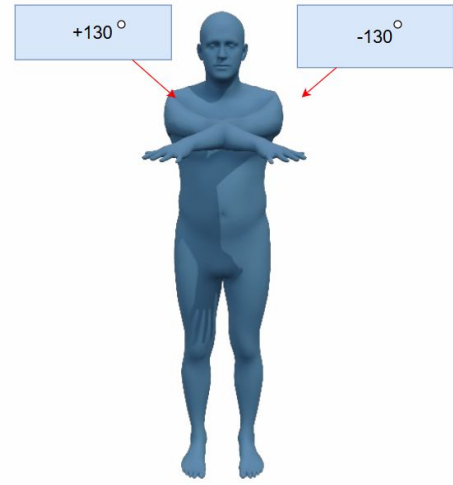


Figure 1. We show extreme poses at $\pm 90^\circ$ for wrist extension/flexion sign convention with left and right sign mirroring consistent with SMPL-X euler angle setup.



(a) Maximum for horizontal abduction.



(b) Maximum horizontal adduction ($\pm 130^\circ$).

Figure 2. Shoulder horizontal ad/abduction within signer space in SMPL-X compatible euler angle setup. Shoulder motion is constrained within a torso-anchored, ground-parallel signer space, disallowing horizontal abduction behind the torso while permitting substantial horizontal adduction for cross-body movements.

movements (see Fig. 2b). This yields a simple deterministic rule that filters out poses with shoulder angles outside these bounds.

3. Motion Capture Data Acquisition

This section complements section 3.2.2 of the main paper. We export MANUS [1] motion from Unity as FBX and

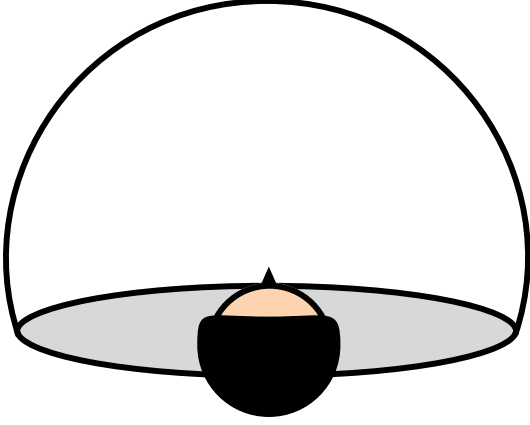


Figure 3. Bird's eye view of signer space envelope showing a torso-anchored 3D workspace. The figure has been adapted from [2].



Figure 4. FBX hierarchy from the MANUS export. Hands controls global placement. ManusHand_L and ManusHand_R parent the SK_Hand meshes and a per-hand root armature with finger and thumb mocap. Animation and Interaction store non-finger transforms that we remove for retargeting.

process it in Blender. The FBX hierarchy in Fig. 4 comprises a top-level Hands node that controls global translation, child nodes ManusHand_L and ManusHand_R, per-hand SK_Hand meshes, and a terminal root that contains the armature.

Rotations in mocap reside on the finger and thumb bones of the armature, while the parent nodes position the gloves about the wrist pivot. After testing several approaches, the most reliable process was to set the armature to a flat “rest” pose, then delete keyframes on the parent containers so that only the finger animations remained. The armature was then aligned as closely as possible with the SMPL-X finger bones as shown in Fig. 5. Differences in finger length did not affect retargeting substantially, but misaligned knuckles caused distortion and stretching in the SMPL-X rig, so

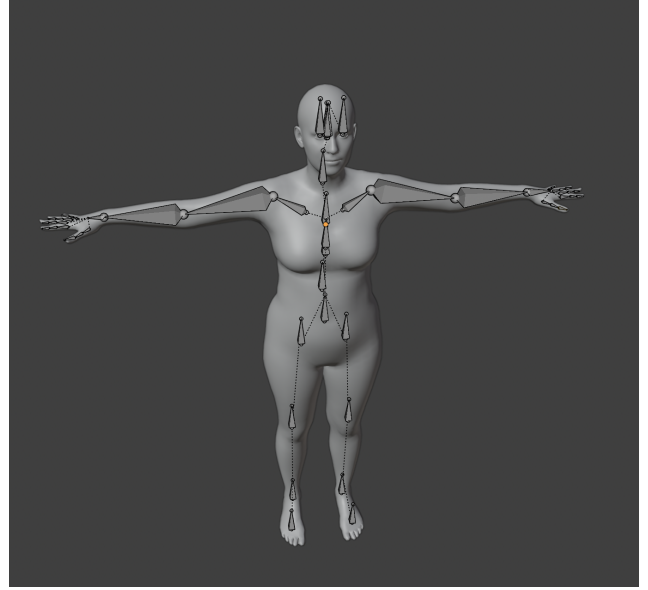


Figure 5. SMPL-X in the rest pose with visible armature.

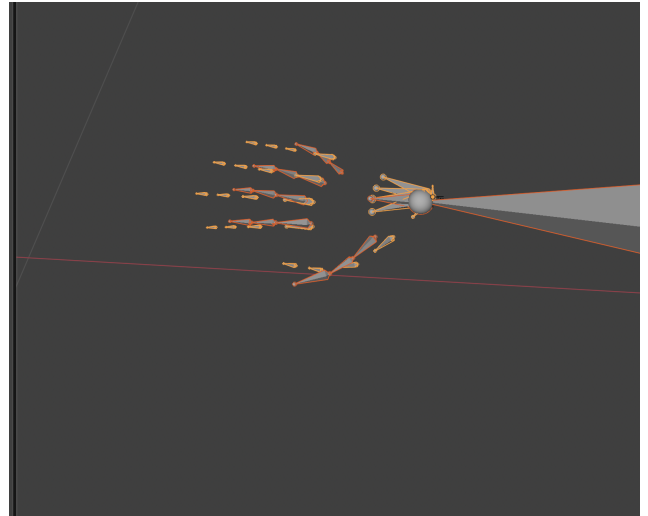


Figure 6. Separate hand armatures during retargeting. We retarget each hand one at a time. We duplicate the SMPL-X rig, retarget the right hand first, then retarget the left hand on the duplicate. We copy the left-hand keyframes from the duplicate back to the original SMPL-X rig, consolidating both hands’ motion. Finger animations are then transferred to SMPL-X.

careful attention was paid to joint spacing. Some discrepancies remained due to restrictions on altering the SMPL-X T-pose.

Since the hands were on separate armatures, they were retargeted one at a time as shown in Fig. 6. We duplicated the SMPL-X rig, retargeted the right hand first, then retargeted the left hand on the duplicate. The left-hand keyframes from the duplicate were copied back to the origi-

nal SMPL-X rig, consolidating both hands’ motion. At this stage, the finger animations were successfully transferred to SMPL-X. A fresh copy of the .fbx file was imported, and only the parent “Hands” keyframes were removed. This preserved the movement of the hands in space, allowing them to be positioned naturally in front of the SMPL-X body.

To enable the arms to move with the MANUS gloves, an inverse kinematics (IK) setup was added. Each arm was given an IK handle constrained to a duplicate wrist bone, which was then linked to `ManusHand.L` and `ManusHand.R`. This enabled the arms to follow the hand movements using a single bone, rather than requiring manual adjustment of the forearm and upper arm. Wrist rotations could not be transferred due to incompatibilities between the SMPL-X T-pose and the bone roll of the MANUS rig, which meant that constraints could not replicate these rotations accurately. Finally, we baked the animations into the SMPL-X armature, replacing the constraints with explicit per-frame rotation keyframes. The temporary constraints and auxiliary IK bones were removed, leaving a clean animation that could be extracted and used.

4. Analysis of Prior Parameters

This section complements section 5.1 of the main paper. We study how data filtering and lightweight bio-mechanical constraints affect body and hand pose estimation. Tables 1 and 2 report hyperparameter sweeps for SignBPoser and SignHPoser under matched architectures. Each setting varies only the training data correction and the presence of a bio-mechanical loss, while we select the best hyperparameter on Evaluation (DEV) and TEST data. We evaluate with MPJPE and MPVPE on both splits, and we summarize the main trends below.

Table 1 reports hyperparameter tuning for three settings of SignBPoser. BP_u uses the unfiltered data, BP_f uses the bio-mechanically filtered data, and BP_{f+bio} adds a body bio-mechanical loss on top of the filtered data. We select the best hyperparameter for each setting on the DEV and TEST sets.

In the unfiltered setting BP_u latent 31 performs worst. Increasing to 32 reduces error on DEV by 21% on MPJPE and 17% on MPVPE, and on TEST by about 18% and 16%. A further increase to 33 brings additional reductions of about 2% and 11% on DEV, and about 5% and 14% on TEST. Latent 33 is therefore the most reliable choice in this setting.

On the other hand, in the bio-mechanical filtered setting BP_f , latent 32 has the highest error. Switching to 31 reduces error on DEV by about 1% MPJPE and 6% MPVPE, and on TEST by about 3% and 2%. Increasing to 33 brings smaller additional gains of roughly 2% on both metrics on DEV and about 2% on TEST. The effect of latent size is therefore mild in this setting.

Finally for filtered-plus-constraint setting BP_{f+bio} , the extremes 0.5 and 2.5 give slightly higher errors. Setting the weight to 1.5 reduces MPJPE by about 1–3% and MPVPE by about 1% on both DEV and TEST. The configurations are close to each other, which indicates stable behavior with the constraint.

In Table 2 we compare three settings for SignHPoser. HP_u is trained on the uncorrected data, HP_f is trained on a bio-mechanically corrected data, and HP_{f+bio} keeps the corrected poses and adds a lightweight hand bio-mechanical loss during training. The architecture matches SignBPoser, and the only differences are data correction and the presence of the hand constraint.

In the HP_u setting, moving from latent 22 to 23 lowers error by about 5% on MPJPE and about 5% on MPVPE on TEST, with similar gains on DEV. Increasing to 24 gives an additional reduction of roughly 2% on MPJPE across splits, and about 2% on MPVPE on TEST and about 5% on DEV. Latent 24 is therefore the most reliable within this setting.

For HP_f , both latent 24 and 22 are worse than 23. From 24 to 23 the error drops by about 8% on MPJPE and about 8% on MPVPE on both DEV and TEST. From 22 to 23, the drop is smaller on MPJPE at about 3%, but larger on MPVPE at about 12–14% across splits. Latent 23 is the preferred choice.

Finally for HP_{f+bio} , weight 2.5 performs worst. Reducing it to 1.5 lowers MPJPE by about 9–10% and MPVPE by about 16–18% on both DEV and TEST. Compared with 0.5, the 1.5 setting also improves by about 2–3% on MPJPE and about 8% on MPVPE. The hand constraint is most effective at 1.5.

From the above results, we can conclude that training for SignBPoser and SignHPoser remains stable under a similar architecture. Simple choices like latent size and lightweight bio-mechanical constraints guide accuracy without instability across the DEV and TEST sets.

5. Ablation of SignHPoser with VPoser

This section complements section 5.1 of the main paper. We evaluate the hand prior SignHPoser using VPoser. Table 3 summarizes ablation results. The first two rows of Table 3 show the results for HP_u and HP_f variants. It can be observed that HP_f outperforms HP_u on all metrics, with relative error reductions of **1.2%** on Upper Body, **1.3%** on Left Hand, and **3.2%** on Right Hand. This demonstrates the importance of the correction process using bio-mechanical constraints on the hand data. Adding a bio-mechanical regularizer to the filtered prior in HP_{f+bio} increases accuracy compared to HP_f , yielding slight improvements on Upper Body (0.05%), Left Hand (0.15%), and Right Hand (1.7%). This indicates that introducing bio-mechanical constraints provides useful physical regularization in the fitting process.

Table 1. **Hyper-parameter tuning of SignBPoser**. We denote SignBPoser trained on BP_u: unfiltered body data, BP_f: filtered body data, BP_{f+bio}: filtered body data with body bio-mechanical loss. We evaluate using Mean Per Joint Position Error (MPJPE) and Mean Per Vertex Position Error (MPVPE), on the recovered joints and meshes.

| Variant | KL | Parameters | | | DEV | | TEST | |
|---------------------|--------------|------------|-----------|------------------|-------------|-------------|-------------|-------------|
| | | Neuron | Latent | Biomech constant | MPJPE↓ | MPVPE↓ | MPJPE↓ | MPVPE↓ |
| BP _u | 0.001 | 512 | 33 | × | 5.87 | 3.73 | 5.69 | 3.62 |
| | 0.001 | 512 | 32 | × | 5.99 | 4.17 | 5.98 | 4.21 |
| | 0.001 | 512 | 31 | × | 7.56 | 5.05 | 7.28 | 5.00 |
| BP _f | 0.001 | 512 | 33 | × | 7.21 | 4.33 | 7.04 | 4.14 |
| | 0.001 | 512 | 32 | × | 7.45 | 4.68 | 7.43 | 4.32 |
| | 0.001 | 512 | 31 | × | 7.37 | 4.41 | 7.17 | 4.24 |
| BP _{f+bio} | 0.001 | 512 | 33 | 0.5 | 7.42 | 4.43 | 7.21 | 4.32 |
| | 0.001 | 512 | 33 | 1.5 | 7.30 | 4.39 | 7.10 | 4.25 |
| | 0.001 | 512 | 33 | 2.5 | 7.37 | 4.42 | 7.29 | 4.29 |

Table 2. **Hyper-parameter tuning of SignHPoser**. We denote SignHPoser trained on HP_u: unfiltered hand data, HP_f: filtered hand data, HP_{f+bio}: filtered hand data with hand bio-mechanical loss.

| Variant | KL | Parameters | | | DEV | | TEST | |
|---------------------|---------------|------------|-----------|------------------|-------------|-------------|-------------|-------------|
| | | Neuron | Latent | Biomech constant | MPJPE↓ | MPVPE↓ | MPJPE↓ | MPVPE↓ |
| HP _u | 0.0001 | 512 | 24 | × | 0.56 | 0.55 | 0.56 | 0.54 |
| | 0.0001 | 512 | 23 | × | 0.57 | 0.58 | 0.57 | 0.55 |
| | 0.0001 | 512 | 22 | × | 0.59 | 0.58 | 0.60 | 0.58 |
| HP _f | 0.0001 | 512 | 24 | × | 0.40 | 0.38 | 0.40 | 0.38 |
| | 0.0001 | 512 | 23 | × | 0.37 | 0.35 | 0.37 | 0.35 |
| | 0.0001 | 512 | 22 | × | 0.38 | 0.40 | 0.38 | 0.40 |
| HP _{f+bio} | 0.0001 | 512 | 23 | 0.5 | 0.40 | 0.41 | 0.40 | 0.41 |
| | 0.0001 | 512 | 23 | 1.5 | 0.39 | 0.38 | 0.39 | 0.38 |
| | 0.0001 | 512 | 23 | 2.5 | 0.43 | 0.45 | 0.43 | 0.45 |

Table 3. Ablation study of SignHPoser with Vposer within DexAvatar.

| Method | UBody (-F)↓ | LHand↓ | RHand↓ |
|---------------------|--------------|--------------|--------------|
| HP _u | 37.25 | 13.56 | 14.53 |
| HP _f | 36.79 | 13.39 | 14.06 |
| HP _{f+bio} | 36.77 | 13.37 | 13.82 |

ing, which explains why plausibility corrections may not yield large vertex reductions. Qualitative comparisons in Fig. 7 show cleaner finger alignment. For certain signs like BESUCHENEINMISCHEN and FRECH, DexAvatar generates more plausible hand poses compared to the ground truth. These effects arise as SignHPoser optimizes toward anatomically plausible hand poses learned from our mocap data.

6. Limitations of the SGNify Ground Truth

This section complements section 5.2 of the main paper. We evaluate using TR-V2V against the SGNify ground truth, which contains occasional implausible hand configurations. DexAvatar shows consistent improvements for both hands and the upper body. The margin remains modest because TR-V2V penalizes distance to the ground truth mesh. When the reference encodes anatomically inconsistent finger postures or knuckle spacing, moving toward a plausible pose does not always reduce vertex distance. The ground truth often contains collapsed fingers and irregular knuckle spac-

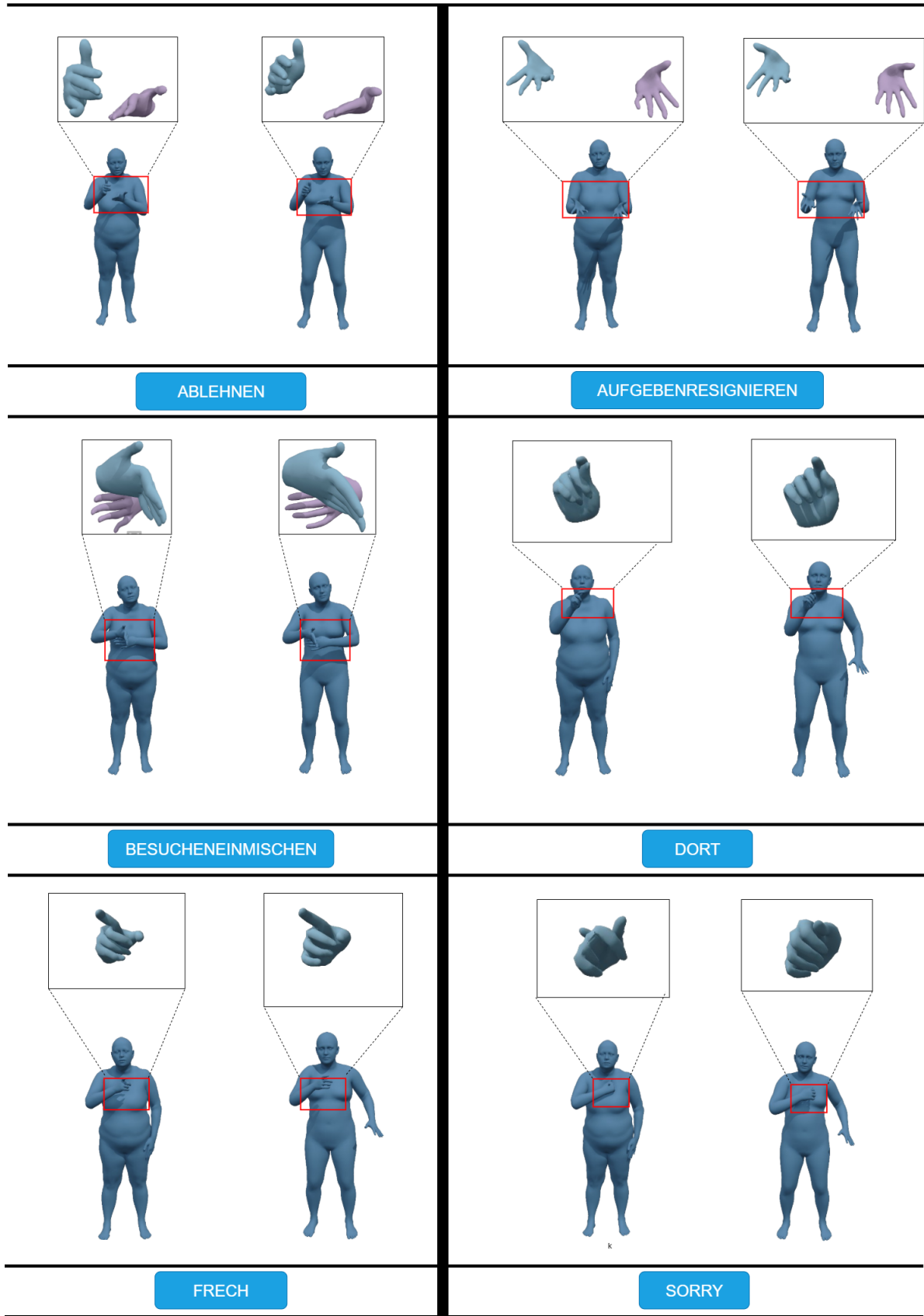


Figure 7. **Examples of independent signs.** Each panel shows two signers, the left signer is the ground truth from the SGNiFy evaluation set and the right signer is DexAvatar generated by our fitting optimization. The SGNiFy [3] ground truth often contains low-quality hand shapes and placements, while DexAvatar produces more plausible poses. The improvement comes from a SignHPrior trained on our mocap dataset.

7. Additional qualitative evaluations with SOTA method

This section complements section 5.2 of the main paper. In Fig. 8, 9, and 10 we present qualitative results of DexAvatar on the MM-WLAuslan [10] SL dataset, compared against SGNify [3] and EVA* under challenging scenarios such as motion blur, self-occlusion, and noise.

7.1. Qualitative evaluation on motion blur images

Fig. 8 presents three examples under motion blur. In Example 1, EVA* shows overspread fingers with a unnatural bending and uneven gaps, while SGNify is an incorrect wrist configuration although bio-mechanically correct, whereas DexAvatar maintains a compact rounded configuration with evenly spaced fingertips that matches the blurred target contact. Building on this, in Example 2, EVA* produces distorted fingers with incorrect spacing and no bio-mechanical stability, SGNify is again an incorrect detection although bio-mechanically correct, while DexAvatar preserves clear fingers with realistic curl and fingertip positions that support the intended overlap with a clean stable contact. Continuing this pattern, in Example 3, EVA* compresses the fingers into a tight bundle so separation is lost and local interpenetration appears along the contact between the index finger and thumb, and also shows a body misalignment relative to the image, SGNify exhibits penetration of fingers with palm that are inconsistent with the image, whereas DexAvatar forms a coherent rounded cluster with visible fingertip order and contact that follows the blurred evidence without fusion and is correct. Overall, DexAvatar maintains accurate and plausible hand contacts under blur.

7.2. Qualitative evaluation on self-occluded images

Fig. 9 presents three examples under self-occlusion. In Example 1, EVA* overspreads the fingers with interpenetration of right hand with the left and inconsistent fingertip spacing, SGNify is bio-mechanically reasonable but misplaces the hand contact, whereas DexAvatar maintains a compact closed configuration with fingertips aligned to the intended contact. Similarly, in Example 2, EVA* keeps both index fingers unnaturally extended while the others curl, SGNify reconstruction consists of overlapping of hands with a wrong wrist orientation for the left hand, while DexAvatar preserves clear fingers with realistic curl and a clean stable contact. Finally, in Example 3, EVA* shows right hand finger postures beyond plausibility, SGNify is plausible but infers the overlap order incorrectly since the right hand should occlude the left rather than the reverse, whereas DexAvatar provides the correct estimation with plausible hands throughout. Overall, DexAvatar maintains accurate and plausible hand contacts under occlusion.

7.3. Qualitative evaluation on images with gaussian noise

We add Gaussian noise to the input frames and compare EVA*, SGNify [3], and DexAvatar in Fig. 10. In Example 1, EVA* reconstructs excessive spacing between the pinky and ring fingers which is bio-mechanically implausible, while SGNify yields a bio-mechanically reasonable hand that nevertheless does not match the target configuration in the image. In contrast, DexAvatar recovers the intended finger arrangement and maintains bio-mechanical constraints across both hands. Moving to Example 2, EVA* degrades to an implausible finger arrangement and SGNify fails to produce any mesh due to missing keypoints under noise, whereas DexAvatar still reconstructs a complete and accurate pose with plausible finger angles and stable contact. Finally, in Example 3, EVA* shows a left-thumb posture beyond plausibility together with distorted right-hand fingers and SGNify again produces no mesh because no keypoints are detected, while DexAvatar returns an anatomically plausible reconstruction with unbroken fingers and consistent bilateral alignment. Overall, DexAvatar remains stable under noisy frames and preserves both accuracy and bio-mechanical plausibility.

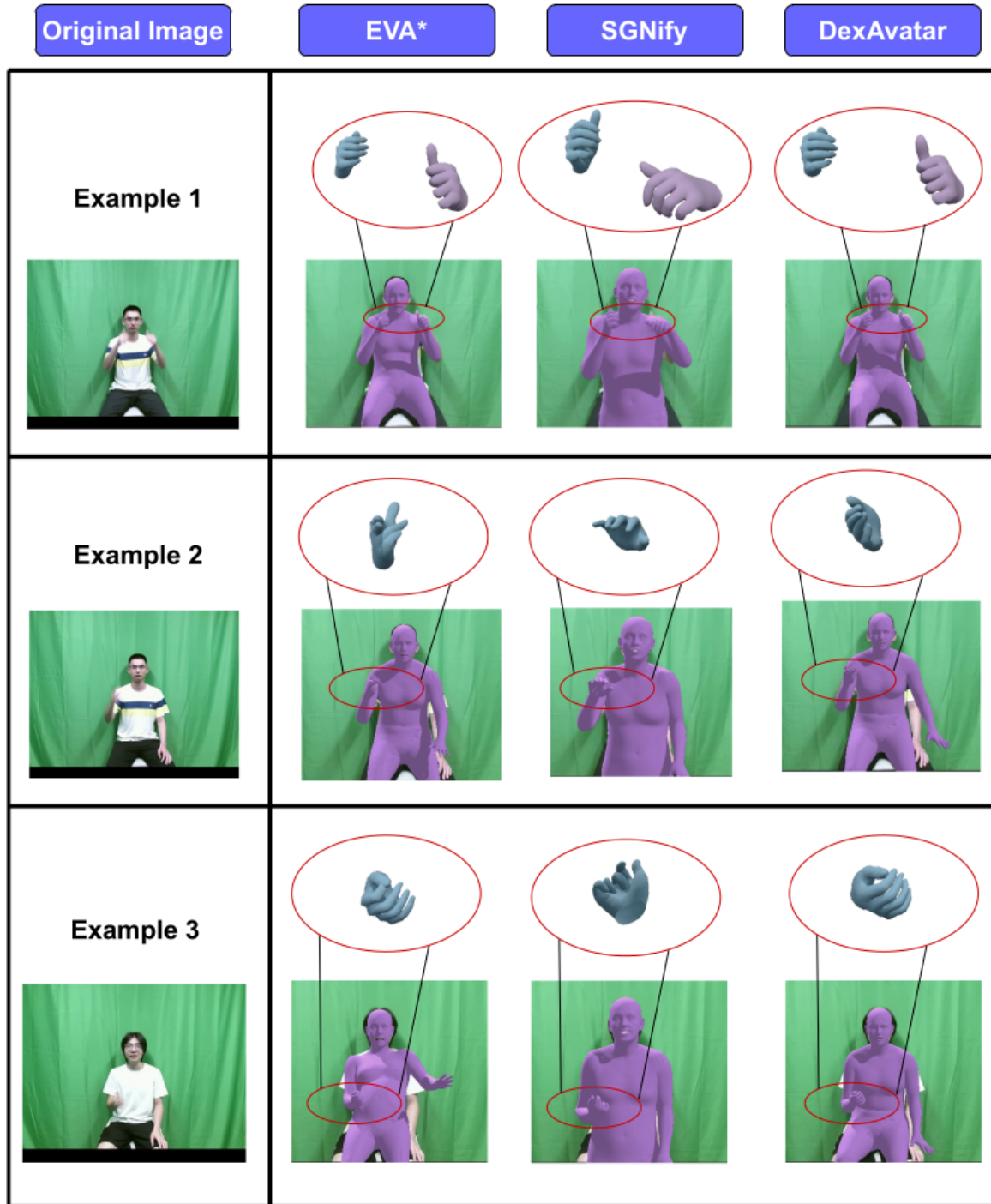


Figure 8. **Qualitative evaluation under motion blur.** We compare EVA*, SGNify [3], and DexAvatar. DexAvatar preserves compact rounded finger configurations and clean contact, while EVA* overspreads or distorts the fingers and SGNify yields incorrect detections or misrepresented contact, with additional body misalignment appearing in harder cases.

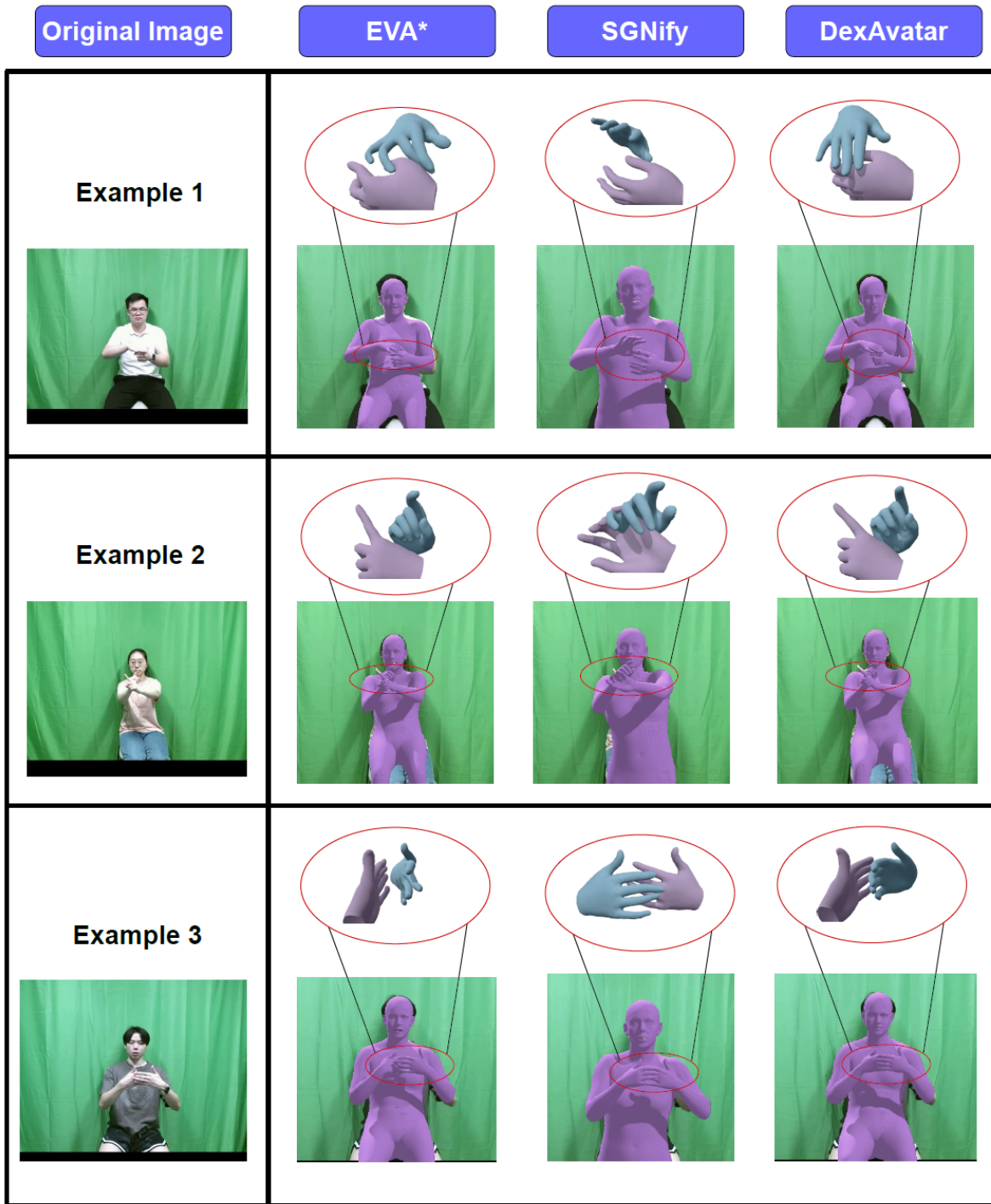


Figure 9. **Qualitative evaluation under self-occlusion.** We compare EVA*, SGNify [3], and DexAvatar. DexAvatar maintains compact finger configurations and correct overlap, while EVA* overspreads or distorts the fingers and SGNify misplaces contact or infers the overlap order incorrectly.

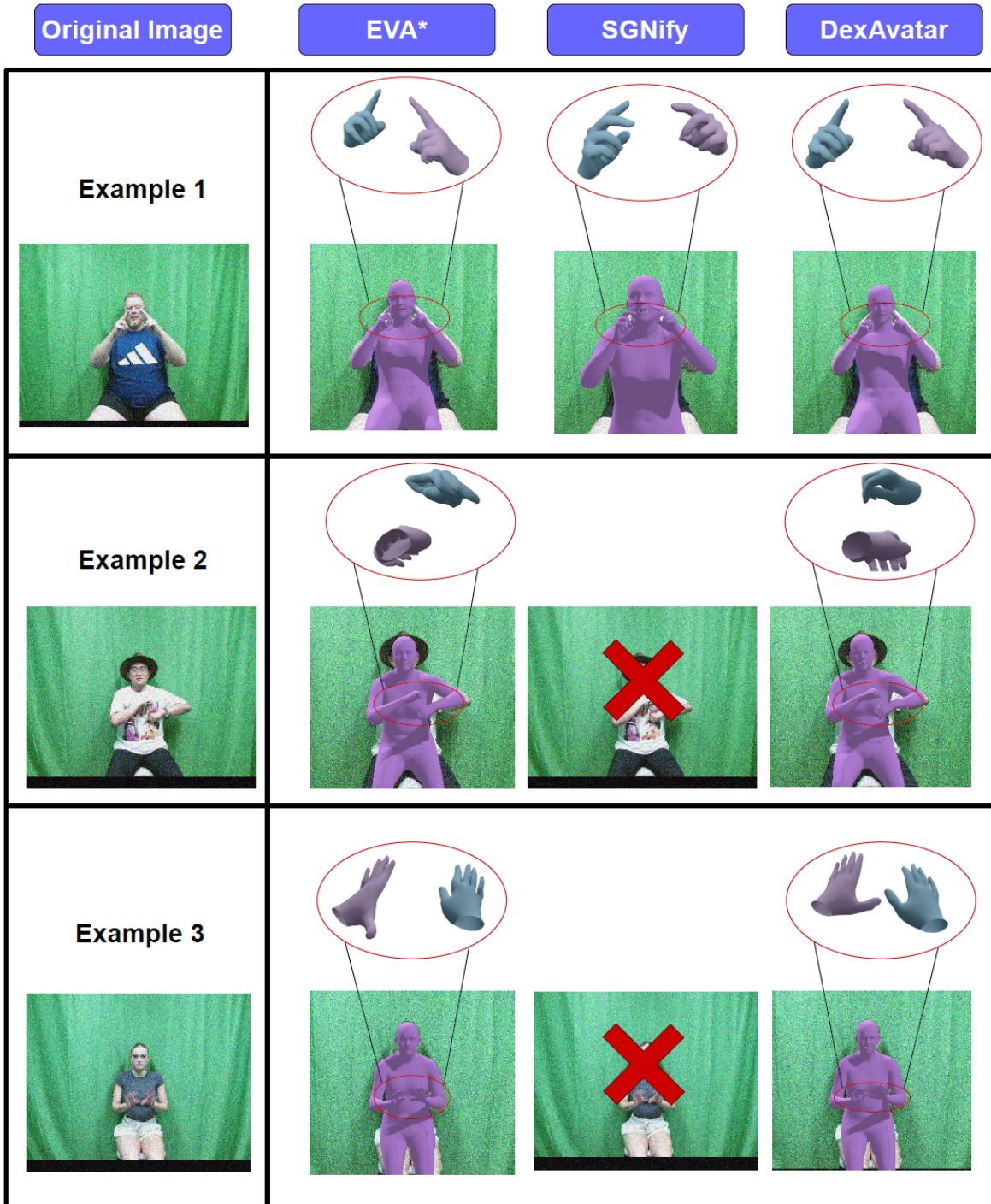


Figure 10. **Qualitative evaluation under gaussian noise.** We add Gaussian noise to input frames and compare EVA*, SGNify [3], and DexAvatar. DexAvatar preserves plausible finger shape and clean contact, while EVA* exhibits implausible spacing and distortions, and SGNify fails to produce a mesh in harder cases due to missing keypoints.

References

- [1] Metagloves pro, 2025. [2](#)
- [2] Chiara Branchini, Lara Mantovan, et al. *A Grammar of Italian Sign Language (LIS)*. Edizioni Ca'Foscari, 2020. [1](#), [3](#)
- [3] Maria-Paola Forte, Peter Kulits, Chun-Hao P Huang, Vasileios Choutas, Dimitrios Tzionas, Katherine J Kuchenbecker, and Michael J Black. Reconstructing signing avatars from video using linguistic priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12791–12801, 2023. [6](#), [7](#), [8](#), [9](#), [10](#)
- [4] Joseph Hamill and Kathleen M Knutzen. *Biomechanical basis of human movement*. Lippincott Williams & Wilkins, 2006. [1](#)
- [5] Duane Knudson. *Fundamentals of biomechanics*. Springer, 2007. [1](#)
- [6] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM Trans. Graph.*, 36(6):194–1, 2017. [1](#)
- [7] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. pages 851–866, 2023. [1](#)
- [8] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019. [1](#)
- [9] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*, 2022. [1](#)
- [10] Xin Shen, Heming Du, Hongwei Sheng, Shuyun Wang, Hui Chen, Huiqiang Chen, Zhuojie Wu, Xiaobiao Du, Jiaying Ying, Ruihan Lu, et al. Mm-wlauslan: Multi-view multi-modal word-level australian sign language recognition dataset. *Advances in Neural Information Processing Systems*, 37:69700–69715, 2024. [7](#)
- [11] Wikipedia contributors. Euler angles. *Wikipedia, The Free Encyclopedia*. Accessed 2025-09-20. [1](#)
- [12] Sherman Wilcox and Rocío Martínez. The conceptualization of space: Places in signed language discourse. *Frontiers in Psychology*, 11:1406, 2020. [1](#)