Supplementary: Weakly Supervised 3D Hand Pose Estimation via Biomechanical Constraints

Adrian Spurr[1,2]*, Umar Iqbal[2], Pavlo Molchanov[2],
Otmar Hilliges[1], and Jan Kautz[2]

[1] Advanced Interactive Technologies, ETH Zurich, Switzerland
[2] NVIDIA, Santa Clara, USA
{adrian.spurr, otmar.hilliges}@inf.ethz.ch
{uiqbal, pmolchanov, jkautz}@nvidia.com

Here we provide additional implementation details and more experimental comparisons. In Section 1 we describe details on how the angle loss is computed and the joint angle interdependence is modeled. Section 2 repeats the ablation study on additional datasets (HO-3D) to highlight the generalizability of results. Section 3 demonstrates the effect of weak-supervision in two additional settings, one using a real dataset as the fully-supervised data and the other using MPII in-the-wild data as weak-supervision. Section 4 compares BMC to an adversarial loss. Sections 5 and 6 provide additional results of bootstrapping via weak-supervision with synthetic or real data. Section 7 shows further qualitative results of using BMC. Sections 8 and 9 provide additional implementation details and results on HANDS2019 challenge, respectively.

1 Joint angle loss

Joint angle ambiguity. The computation of the joint angles lead to ambiguities. More specifically, two different vectors on the unit sphere may map to the same joint angles.

For example, given two bones $\mathbf{b}_i^{\mathbf{F}_i,1} = [1,0,1]$ and $\mathbf{b}_i^{\mathbf{F}_i,2} = [-1,0,1]$ in a coordinate frame $\mathbf{F}_i$, we have using $\mathrm{P}_{xz}(\mathbf{b}_i^{\mathbf{F}_i,1}) = [1,0,1]$ and $\mathrm{P}_{xz}(\mathbf{b}_i^{\mathbf{F}_i,2}) = [-1,0,1]$:

$$\begin{aligned}
\theta_i^{\mathrm{f},1} &= \alpha(\mathrm{P}_{xz}(\mathbf{b}_i^{\mathbf{F}_i,1}), \mathbf{z}_i) = \alpha([1,0,1], \mathbf{z}_i) \\
&= \pi/4 \\
\theta_i^{\mathrm{a},1} &= \alpha(\mathrm{P}_{xz}(\mathbf{b}_i^{\mathbf{F}_i,1}), \mathbf{b}_i^{\mathbf{F}_i,1}) \\
&= \alpha([1,0,1],[1,0,1]) = 0 \\
\theta_i^{\mathrm{f},2} &= \alpha(\mathrm{P}_{xz}(\mathbf{b}_i^{\mathbf{F}_i,2}), \mathbf{z}_i) = \alpha([-1,0,1], \mathbf{z}_i) \\
&= \pi/4 \\
\theta_i^{\mathrm{a},2} &= \alpha(\mathrm{P}_{xz}(\mathbf{b}_i^{\mathbf{F}_i,2}), \mathbf{b}_i^{\mathbf{F}_i,2}) \\
&= \alpha([-1,0,1],[-1,0,1]) = 0
\end{aligned} \tag{1}$$

Therefore, both bones map to the same angle pair $(\pi/4, 0)$. To resolve this, we perform an octant look up. Given the flexion angle θ_i^{f} and abduction angle θ_i^{a} of bone i, we negate the respective angle if the bone lies within the negative x-octant or negative y-octant:

$$\begin{aligned} \theta_i^{\mathrm{f}} &= \begin{cases} -\theta_i^{\mathrm{f}}, & \text{if } b_{i,x}^{\mathbf{F}_i} < 0 \\ \theta_i^{\mathrm{f}}, & \text{else} \end{cases} \\ \theta_i^{\mathrm{a}} &= \begin{cases} -\theta_i^{\mathrm{a}}, & \text{if } b_{i,y}^{\mathbf{F}_i} < 0 \\ \theta_i^{\mathrm{a}}, & \text{else} \end{cases} \end{aligned} \tag{2}$$

Where $b_{i,x}^{\mathbf{F}_i}$,$b_{i,y}^{\mathbf{F}_i}$ is the x/y-component of the bone vector given in coordinates of its local coordinate frame $\mathbf{F}_i$. This leads to angles in the range $\theta_i^{\mathrm{f}} \in [-\pi, \pi]$ and $\theta_i^{\mathrm{a}} \in [-\pi/2, \pi/2]$ respectively.

Approximation of Convex Hull. Fig. 1 plots the distribution of the pinkys MCP flexion/extension angles of the FH dataset, visualized as red points. The red rectangle corresponds to the valid range of angles when considering both angle limits independently. Hence the corners correspond to $(\min_i^f, \min_i^a), (\min_i^f, \max_i^a), (\max_i^f, \max_i^f), (\max_i^f, \min_i^a)$ in counter-clockwise order, where $\min_i^k, \max_i^k$ corresponds to the minimum/maximum of angle θ_i^k, where $k \in \{a, f\}$.

In order to take the dependence of the angle limits in account, we first compute the convex hull of the angle points. However, depending on the shape of the point cloud, the number of points lying on the hull can vary and be numerous. In order to keep the number of hull points low and consistent for all joint angles, we approximate this hull in two steps. We first employ the Ramer-Douglas-Peucker algorithm, a polygon simplification algorithm. This significantly reduces the number of vertices in the hull, but still results in a variable number. To ensure consistency, we apply a greedy algorithm that iteratively removes points such that the hull encompasses as many points as possible until we reach the desired number of points, resulting in our approximation $\mathcal{H}_i$. For all our experiments, we set number of points to be 10. The green polygon in Fig. 1 displays this approximation to the convex hull.

Distance computation. To compute the distance $\mathcal{H}_i$, we compute two values. The first indicates if an angle point $\boldsymbol{\theta}_i$ is contained within the hull. The second corresponds to the distance to the hull. Here we detail how we compute both values. For ease of notation, we assume that the points in $\mathcal{H}_i$ are ordered counter-clockwise beginning from any point in $\mathcal{H}_i$. Let $\mathcal{H}_{i,k}$ be the k-th point in $\mathcal{H}_i$. An edge $\mathbf{v}_k$ of the hull is given as:

$$\begin{aligned} \mathbf{v}_k &= \mathcal{H}_{i,k+1} - \mathcal{H}_{i,k}, \text{ for } k \in [1, 10] \\ \mathbf{w}_k &= \boldsymbol{\theta}_i - \mathcal{H}_{i,k}, \text{ for } k \in [1, 10] \end{aligned} \tag{3}$$

Where we define $\mathcal{H}_{i,11} = \mathcal{H}_{i,1}$ to wrap around the hull.

To compute if a point $\boldsymbol{\theta}_i$ is contained within $\mathcal{H}_i$, we exploit the convexity of the hull and make use of the cross-product. Specifically, we compute the 2D

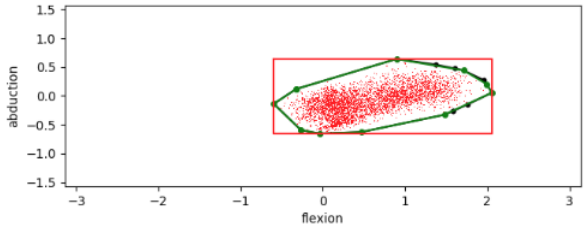

Fig. 1: $(\theta^{\mathrm{f}}, \theta^{\mathrm{a}})$-plane. Green: $\mathcal{H}_i$. Red: min/max-box

cross-product between $\mathbf{v}_k$ and $\mathbf{w}_k$. Intuitively, if the cross-product $\mathbf{w}_k \times \mathbf{v}_k$ is positive for any given edge k, then the angle point lies outside of the hull. If its negative for all, it is contained within. If it lies on the hull, we consider it to be contained within it. More formally:

$$c = \prod_{k=1}^{10} \mathbb{1}_{(\mathbf{w}_k \times \mathbf{v}_k) \leq 0} \tag{4}$$

To compute the distance of $\boldsymbol{\theta}_i$ to the hull, we compute its distance to each edge and take the minimum. Given edge $\mathbf{v}_k$ and point $\boldsymbol{\theta}_i$, their distance is the minimum distance between either endpoints of $\mathbf{v}_k$ or the projection of $\mathbf{w}_k$ onto $\mathbf{v}_k$. Formally:

$$\begin{aligned} t &= \max(0, \min(1, \mathbf{w}_k^T \mathbf{v}_k / ||\mathbf{v}_k||_2^2)) \\ \mathbf{p}_k &= \mathcal{H}_{i,k} + t\mathbf{v}_k \\ D(\mathbf{v}_k, \boldsymbol{\theta}_i) &= |\cos(\boldsymbol{\theta}_i) - \cos(\mathbf{p}_k)| + |\sin(\boldsymbol{\theta}_i) - \sin(\mathbf{p}_k)| \end{aligned} \tag{5}$$

Where the min/max ensures that we do not extend beyond the endpoints of $\mathbf{v}_k$. Given the distance to the edge, we can compute the distance to the hull $\mathcal{H}_i$:

$$D(\boldsymbol{\theta}_i, \mathcal{H}_i) = \min_k D(\mathbf{v}_k, \boldsymbol{\theta}_i) \tag{6}$$

This formulation computes the distance towards $\mathcal{H}_i$, whether the point is contained or not. We do not want to penalize points that lie within the hull, as that constitutes our range of valid angles. Therefore we make use of the quantity c computed in Eq. 4, which leads to the final angle loss function:

$$D_A(\boldsymbol{\theta}_i, \mathcal{H}_i) = (1 - \mathbb{1}_c) D(\boldsymbol{\theta}_i, \mathcal{H}_i) \tag{7}$$

This returns a loss of 0 if the angle point $\boldsymbol{\theta}_i$ is contained, otherwise it returns the distance to the approximation of the convex hull $\mathcal{H}_i$. This constitutes our angle loss for bone i.

2 Ablation study

We repeat the ablation study with the HO-3D dataset. All evaluations are done on a custom split, where we manually extract two sequences for the test and use

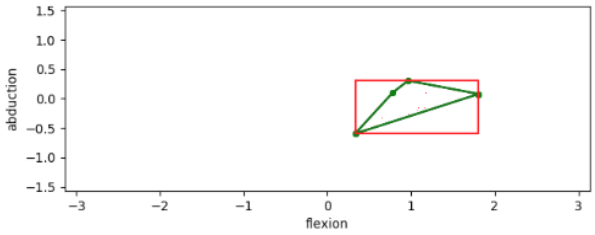

Fig. 2: $(\theta^{\mathrm{f}}, \theta^{\mathrm{a}})$-plane. Green: $\mathcal{H}_i$. Red: min/max-box

the remainder for the training set. Each error is computed for the root relative case.

Refinement network. We train two models using full supervision on HO-3D ($\mathbf{3D}_{\mathrm{HO3D}}$). The first model (w/o refinement) does not use the proposed refinement network, whereas the second does (w.refinement). We showcase the performance difference in the first row of Tab. 1. We note a reduction of 2.97mm mean error when using the refinement network.

BMC ablation. We study the individual contribution of the BMC losses. We bootstrap the 3D annotation from synthetic data and use only the 2D annotation of HO-3D. The first model constitutes our baseline, which is trained only on that data ($\mathbf{3D}_{\mathrm{RHD}} + \mathbf{2D}_{\mathrm{HO3D}}$). We incrementally add the bone length loss $\mathcal{L}_{\mathrm{BL}}$, the root bone loss $\mathcal{L}_{\mathrm{RB}}$ and lastly the angle loss $\mathcal{L}_{\mathrm{A}}$. We train a fully supervised model ($\mathbf{3D}_{\mathrm{RHD}} + \mathbf{3D}_{\mathrm{HO3D}}$) which is our upper bound. We refer to the second section of Tab. 1. Each loss contributes towards a reduction in mean error, culminating in a total decrease of 5.21mm as compared to our 2D only baseline.

Co-dependency between angles. We train two models. The first models the angle limits independently, whereas the second takes the dependency of the limits into account. The resulting performance is shown in Tab. 1. We note a minor performance degradation. We attribute this to the extremely limited angle range contained in the HO-3D dataset. As it contains subjects holding various object in a gripping pose while rotating it in front of the camera, the actual angles of the fingers do not change. Therefore the range of angles across the dataset is low, which leads to a very tight angle limit. This does not generalize well, which in turn hurts performance. Fig. 2 displays the angle-plane plot for HO-3D using the pinkys MCP flexion/extension angles. Comparing with Fig. 1, which plots the plane for the same finger for FH, we see that the resulting range of HO-3D is a lot more severely limited. This is to be expected, as HO-3D is a very constrained dataset due to the aforementioned reason.

BMC limits. We study the effect of approximating the BMC limits when using a different dataset to compute these values. We compute the hand parameters from RHD and perform the same weakly-supervised experiment as previously ($\mathbf{3D}_{\mathrm{RHD}} + \mathbf{2D}_{\mathrm{HO3D}}$). As can be seen in the last row of Tab. 1, we note a slight increase in loss, however it still clearly outperforms the 2D baseline in mean error (18.50 mm vs 23.71 mm).

Table 1: Ablation studies on **validation** split of HO-3D. The models of the first section was trained on our train split of HO-3D.

HO-3D	3D Pose Estimation (root-relative) EPE (mm) mean ↓	median ↓	AUC ↑
Effect of Z^{root} refinement			
w/o refinement	25.34	24.39	0.79
w. refinement	22.37	23.01	0.83
Effect of BMC components			
$\mathbf{3D}_{\text{RHD}}$ + $\mathbf{2D}_{\text{HO3D}}$	23.71	22.07	0.78
+ $\mathcal{L}_{\text{BL}}$	22.15	20.27	0.80
+ $\mathcal{L}_{\text{RB}}$	18.83	17.79	0.87
+ $\mathcal{L}_{\text{A}}$	18.50	17.41	0.87
$\mathbf{3D}_{\text{RHD}}$ + $\mathbf{3D}_{\text{HO3D}}$	16.74	16.94	0.89
Effect of angle co-dependency			
Independent	18.30	17.40	0.87
Dependent	18.50	17.41	0.87
Effect of BMC limits			
Approximated	19.21	17.88	0.86
Computed	18.50	17.41	0.87

3 Effect of Weak-Supervision

We repeat the experiments of Section 5.3 in the main paper using different datasets. We show that the effect of weak-supervision also holds when using fully labeled real data or weakly-labeled in-the-wild data.

STB. We reproduce the results of Section 5.3 in the main paper, but instead of using RHD we use STB [4] as the fully supervised dataset. The weakly-supervised dataset remains FH. The purpose of this experiment is to demonstrate that the effect of weak supervision also takes place when using a real dataset for full supervision. Table 2 (top) shows the result.

MPII - in-the-wild dataset. We reproduce the results of Section 5.3 in the main paper, but using MPII [3] as our weakly-supervised dataset. This is to demonstrate the effect of weak-supervision stemming from datasets collected in-the-wild, a potentially useful supervision source. We evaluate on the validation split of FH. Table 2 (bottom) shows the result. Note that as the MPII dataset only contains 2D labels and no 3D annotation is provided, the fully supervised upper bound cannot be performed and is therefor omitted from the table.

4 Comparison with Adversarial loss

It is intuitive to think of drawing parallels between BMC and an adversarial loss. BMC can be interpreted as a discriminator penalizing poses that do not adhere to the distribution of valid hand poses. However, BMC models the task at hand more closely and only requires the limits, whereas a discriminator requires access to a full dataset of 3D poses. In order to see how a discriminator performs against

Table 2: This table show-cases the same effect of weak-supervision as Table 2 in the main paper but evaluated in different settings. All models are evaluated on the **validation** split of FH. (top) We use STB as the fully labeled dataset and supplement is using weakly-labeled FH. (bottom) We use RHD as the fully labeled dataset and MPII as the weakly-supervised data. The same trend can be observed in both settings. Adding weakly-supervised data improves 3D prediction performance due to predicted 3D poses with the correct 2D projection. By incorporating our proposed biomechanically constraints we significantly improve 3D pose accuracy due to more accurate **Z**. Note that as the MPII dataset only contains 2D labels and no 3D annotation is provided, the fully supervised upper bound cannot be performed and is therefor omitted from the table.

Effect of weak-supervision	Description	mean ↓ 2D (pixel)	Z (mm)	3D (mm)
	3D labels: STB			
$\mathbf{3D}_{\text{STB}} + \mathbf{3D}_{\text{FH}}$	Fully supervised, real	3.85	5.68	9.05
$+ \mathcal{L}_{\mathbf{BMC}}$ **(ours)**	+ BMC	**3.83**	**5.50**	**8.89**
$\mathbf{3D}_{\text{STB}}$	Fully sup. lower bound	20.45	36.80	54.92
$+ \mathbf{2D}_{\text{FH}}$	+ Weakly supervised, real	**3.86**	35.41	42.02
$+ \mathcal{L}_{\mathbf{BMC}}$ **(ours)**	+ BMC	3.88	**11.17**	**18.58**
	2D labels: MPII			
$\mathbf{3D}_{\text{RHD}}$	Fully supervised, synthetic only	12.35	20.02	30.82
$+ \mathbf{2D}_{\text{MPII}}$	+ Weakly supervised, real	10.36	19.77	28.81
$+ \mathcal{L}_{\mathbf{BMC}}$ **(ours)**	+ BMC	**10.35**	**17.72**	**27.10**

BMC, we perform an experiment in the same setting as the ablation study. We train on fully supervised RHD and weakly-supervised FH, and evaluate on the validation split of FH. As it has not been shown if and how the adversarial loss works for the task of 3D hand pose estimation, we adapt a model from literature applied to 2D body pose [1]. In order to adjust to the new setting, we performed a search for the optimal hyperparameters to improve the performance of the discriminator. We show the results in Table 3. As can be seen, BMC outperforms the adversarial loss. We hypothesise this is due to BMC modeling the task at hand more closely.

5 Bootstrapping with Synthetic Data

We show the full results of the online evaluation on FH and HO-3D in Table 4.

6 Bootstrapping with Real Data

Tab. 5 shows the full result of Bootstrapping with real data, as evaluated by the online submission system [3]. Recall that we assume the remainder of the data to be weakly-supervised, i.e it contains the 2D annotation. We list the exact number

[3] `https://competitions.codalab.org/competitions/21238`

Table 3: We compare using BMC to an adversarial loss adapted from [1]. BMC outperforms the adversarial loss. We hypothesise this is due to BMC modeling the task at hand more closely.

Comparison to adversarial loss	Description	3D Pose Estimation (root-relative) EPE (mm) mean ↓	median ↓	AUC ↑
$\mathbf{3D}_{\mathrm{RHD}} + \mathbf{2D}_{\mathrm{FH}}$	Baseline	20.92	16.93	0.81
$\mathbf{3D}_{\mathrm{RHD}} + \mathbf{2D}_{\mathrm{FH}} + \mathcal{L}_{\mathbf{BMC}}$	BMC	**15.48**	**13.49**	**0.91**
$\mathbf{3D}_{\mathrm{RHD}} + \mathbf{2D}_{\mathrm{FH}} + \mathcal{L}_{\mathbf{adv}}$	Adversarial	17.60	14.38	0.87

Table 4: Bootstrapping results on the respective **test split**, as evaluated by the *online submission system*. Results are given in mm.

FH	Description	aligned mean ↓	aligned AUC ↑	unaligned mean ↓	unaligned AUC ↑
Zimmermann et al. [5]	fully supervised FH	1.10	0.78	**7.13**	0.19
$\mathbf{3D}_{\mathrm{RHD}} + \mathbf{3D}_{\mathrm{FH}}$	fully supervised RHD/FH	**0.90**	**0.82**	7.54	**0.20**
$\mathbf{3D}_{\mathrm{RHD}}$	fully supervised RHD	1.60	0.69	15.15	0.06
$+ \mathbf{2D}_{\mathrm{FH}}$	+ weakly-supervised FH	1.26	0.75	13.02	0.14
$+ \mathcal{L}_{\mathbf{BMC}}$	+ BMC	**1.13**	**0.78**	**10.39**	**0.15**
HO3D	Description	EXTRAP ↓	INTERP ↓	OBJECT ↓	SHAPE ↓
$\mathbf{3D}_{\mathrm{RHD}} + \mathbf{3D}_{\mathrm{HO3D}}$	fully supervised HO3D	18.22	5.02	16.56	10.79
$\mathbf{3D}_{\mathrm{RHD}}$	fully supervised RHD	20.84	33.57	35.08	23.94
$+ \mathbf{2D}_{\mathrm{HO3D}}$	+ weakly supervised HO3D	19.57	25.16	25.79	21.05
$+ \mathcal{L}_{\mathbf{BMC}}$	+ BMC	**18.42**	**10.31**	**19.91**	**12.51**

of 3D labeled samples used, in addition to the percentage wrt. to the entire dataset it corresponds to. Note that the percentage values have been rounded for readability, but the number of samples is exact. We divide the table according to three categories a) **Aligned / Unaligned** - Procrustes analysis is used to align before computing the score b) **Mean / AUC** - The AUC is given for PCK values that lie in an interval from 0 mm to 50 mm with 100 equally spaced thresholds. c) **With / Without BMC** - Using our proposed biomechanical constraints. We first focus on the aligned results. Using BMC, the required amount of 3D annotated data for a given AUC is approximately *halved*. This trend continues for labeling percentages up to $\sim 13\%$. For example, to achieve the same performance as a model that is trained without BMC on 3810 3D labeled data samples, BMC achieves the same performance with 1993 3D labeled samples, roughly half the amount.

A similar trend can be observed for the unaligned score. For labeling percentages up to 6.8% (1993), the required amount of data to reach the same performance is approximately halved (997).

7 Qualitative results

We show qualitative results of the Bootstrapping with Synthetic Data experiment in Fig. 3. We display the predicted $\mathbf{J}^{3D}$ of both $\mathbf{3D}_{\mathrm{RHD}} + \mathbf{2D}_{\mathrm{FH}}$ (w/o BMC) and

Table 5: Scores as evaluated on the online submission system. The first column denotes the percentage (in brackets) of 3D annotated samples used during training, where the remainder is annotated only with 2D labels. Note that the percentages are rounded, but the number of samples are exact. + **indicates the model trained with BMC,** − **indicates the model trained without it.**

FH	+: with BMC (**ours**)	Aligned				Unaligned			
	−: without BMC	mean ↓		AUC ↑		mean ↓		AUC ↑	
3D samples: Number	3D samples: Perc.	−	+	−	+	−	+	−	+
1	(3.4e-3%)	1.96	1.64	0.62	0.68	34.86	18.40	0.08	0.11
5	(0.017%)	1.85	1.41	0.64	0.72	26.40	15.26	0.11	0.13
14	(0.045%)	1.78	1.39	0.65	0.73	25.24	12.98	0.11	0.13
27	(0.094%)	1.75	1.34	0.66	0.73	23.90	11.93	0.12	0.14
127	(0.43%)	1.54	1.24	0.70	0.76	21.83	12.08	0.13	0.16
499	(1.7%)	1.23	1.18	0.76	0.77	11.68	10.88	0.17	0.18
997	(3.4%)	1.14	1.12	0.77	0.78	9.85	9.42	0.18	0.19
1993	(6.8%)	1.10	1.07	0.78	0.79	8.83	8.75	0.19	0.20
3810	(13%)	1.06	1.04	0.79	0.79	8.01	7.90	0.21	0.21
7327	(25%)	1.02	1.01	0.80	0.80	7.91	7.84	0.21	0.21
14653	(50%)	0.99	1.00	0.80	0.80	7.46	7.56	0.22	0.22
29305	(100%)	0.98	0.98	0.81	0.81	7.18	7.18	0.23	0.23

$\mathbf{3D}_{\text{RHD}} + \mathbf{2D}_{\text{FH}} + \mathcal{L}_{\text{BMC}}$ (w. BMC). Two views are shown. The first displays the view from the front or camera view (looking in direction of the z-axis), the second shows the view from the top of the world space, looking down (looking in the *opposite* direction of the x-axis). Additionally, we plot the 2D predictions of both models, where green corresponds to without BMC and red is the model using BMC.

We see that despite both models predicting accurately the 2D pose, its predicted 3D pose are different. Not using BMC, the model predicts bio-physically implausible poses. This is due to unseen 3D poses, views and occlusions. Additionally, the 3D component of the model has only been trained on synthetic data. For example, RHD does not contain object occlusions or ego-centric views. Using BMC, our model can better adapt its depth-component during training to these unseen 3D poses, resulting in more accurate predictions.

8 Architecture and training

We use a standard ResNet-50 network for our backbone. We replace the last linear layer to output a 21×3 dimensional vector. The first two dimensions correspond to the $2D$ keypoints, whereas the last layer corresponds to the root-relative depth Z^{r}.

Our Z^{root} refiner consists of a three layered MLP, using leaky ReLU non-linearity. We used BatchNorm in between all layers except the last. For the

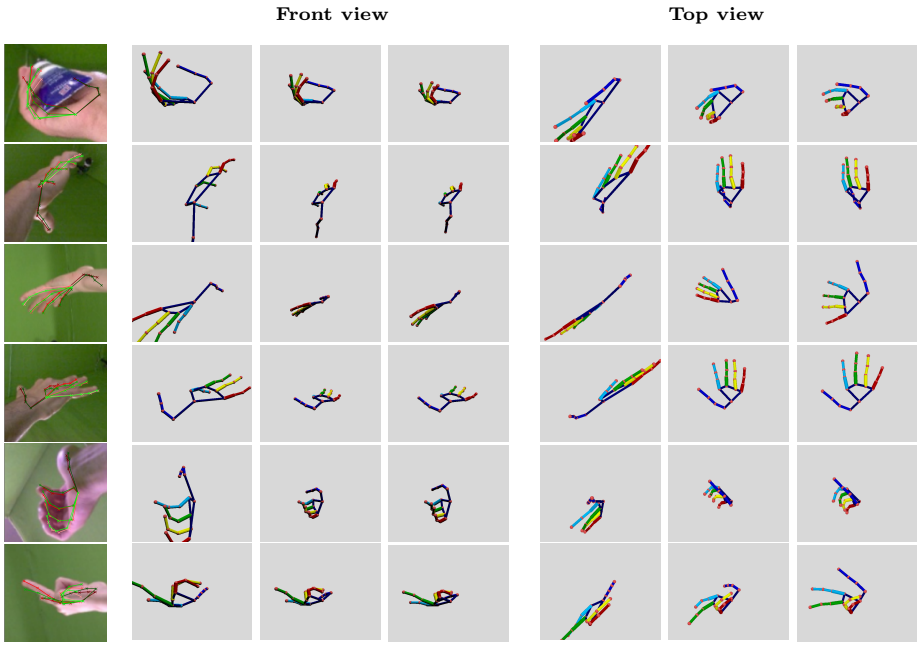

Fig. 3: Qualitative results of the Boostrapping with Synthetic Data experiment. Testing performed on custom split of FH. Fig. 3a: We see that the model trained without BMC (green), as well as the model trained with BMC (red), perform equally well on the 2D prediction task. Fig. 3d, Fig. 3g show the ground-truth joint skeleton from the camera view, as well as the "top" view looking down, respectively. Fig. 3b and Fig. 3e show the 3D predictions of the model trained fully supervised on RHD and weakly-supervised on FH. Despite the accurate 2D predictions, the 3D pose is incorrect, displaying implausible bio-physical poses. Fig. 3c and Fig. 3f show the result of incorporating BMC into the model. The predictions are kinematically and structurally sound, and as a result closer to the ground-truth predictions.

Table 6: Architecture of the refinement network. It takes the predicted and calculated values $\mathbf{z}^r \in \mathbb{R}^{21}$, $\mathbf{K}^{-1}\mathbf{J}^{2D} \in \mathbb{R}^{21\times 3}$, $Z^{root} \in \mathbb{R}$ and outputs a residual term r such that $\hat{Z}^{root}_{\text{ref}} = \hat{Z}^{root} + r$

Refinement Network
Linear(85, 128)
LeakyReLU(0.01)
BatchNorm
Linear(128, 128)
LeakyReLU(0.01)
BatchNorm
Linear(128, 1)

Table 7: HANDS2019 challenge results on the **test split** of HO-3D, as evaluated by the *online submission system*. All methods were trained only on HO-3D. We show the top four submission. The winner was selected based on the extrapolation score. Results are given in mm.

HO-3D	EXTRAP ↓	INTERP ↓	OBJECT ↓	SHAPE ↓
Ours	**24.74**	6.70	27.36	**13.21**
Nplwe	29.19	**4.06**	**18.39**	15.79
lin84	31.51	19.15	30.59	23.47
Hasson et al. [2]	38.42	7.38	31.82	15.61

cross-dataset evaluation, we empirically found that not using BatchNorm resulted in better accuracy. The exact architecture using BatchNorm is listed in Tab. 6.

The network was trained for 70 epochs using SGD with a learning rate of $5e$–3 and a step-wise learning rate decay of 0.1 after every 30 epochs.

We set the weight values as follows: $\lambda_{2D} = 1$, $\lambda_{Z^{\text{r}}} = 5$, $\lambda_{Z^{\text{root}}} = 1$. For all experiments using BMC, we set the individual weights of the losses as follows: $\lambda_{\text{BL}} = 0.1$, $\lambda_{\text{RB}} = 0.1$, $\lambda_{\text{A}} = 0.01$

9 HANDS2019 challenge

The HANDS2019 challenge[4] was organized to evaluate cutting edge methods for 3D hand pose estimation. The rules of challenge task #3 required us to train solely on the HO-3D dataset. We trained the proposed model without auxiliary losses. The refinement step was vital for achieving the first place of the competition, demonstrating the performance of the underlying backbone model.

[4] `https://competitions.codalab.org/competitions/21116`

References

1. Drover, D., Chen, C.H., Agrawal, A., Tyagi, A., Phuoc Huynh, C.: Can 3d pose be learned from 2d projections alone? In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 0–0 (2018)
2. Hasson, Y., Varol, G., Tzionas, D., Kalevatykh, I., Black, M.J., Laptev, I., Schmid, C.: Learning joint reconstruction of hands and manipulated objects. In: CVPR (2019)
3. Simon, T., Joo, H., Matthews, I., Sheikh, Y.: Hand keypoint detection in single images using multiview bootstrapping. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 1145–1153 (2017)
4. Zhang, J., Jiao, J., Chen, M., Qu, L., Xu, X., Yang, Q.: 3d hand pose tracking and estimation using stereo matching. arXiv:1610.07214 (2016)
5. Zimmermann, C., Ceylan, D., Yang, J., Russell, B., Argus, M., Brox, T.: FreiHAND: A dataset for markerless capture of hand pose and shape from single RGB images. In: ICCV (2019)