Model	uv-loss	1 cm	2 cm	3 cm	5 cm	10 cm	20 cm
DensePose-RCNN (R50) [5]	`MSE`	5.21	18.17	31.01	51.16	68.21	78.37
	`full` (ours)	**5.67**	**18.67**	**32.70**	**53.14**	**71.25**	**80.47**
HRNetV2-W48 [*]	`MSE`	4.31	15.19	27.14	47.07	69.76	78.66
	`full` (ours)	**5.70**	**18.81**	**31.88**	**52.20**	**74.21**	**82.12**
HG, 1 stack (Slim DensePose [12])	`MSE`	4.31	15.62	28.30	49.92	74.15	**83.01**
	`full` (ours)	**5.34**	**18.23**	**31.51**	**52.40**	**74.69**	82.94
HG, 8 stacks (Slim DensePose [12])	`MSE`	6.04	20.25	35.10	56.04	79.63	87.55
	`full` (ours)	**6.41**	**20.98**	**35.17**	**56.48**	**80.02**	**87.96**

Table 1: **Performance of uncertainty-based models on the DensePose-COCO dataset [5].** [*] Sun et al. High-Resolution Representations for Labeling Pixels and Regions. arXiv:1904.04514v1, 2019.

1: R1: The label-conditioned branch . . . seems [to be] only in Tab. 4. R2: The model whose uncertainty heads are conditioned on the ground truth during training performs better at test time. There are two reasons for modelling uncertainty: (i) to better understand systematic annotation errors at training time, which leads to more robust training and better point-wise prediction accuracy at test time and (ii) to be able to predict uncertainty at test time, regardless of whether this also results in better point-wise prediction.

Effect (i) was observed in several papers (e.g. [14]) and is mostly due to the ability of the model to detect and discount annotation errors and very hard examples.

Conditioning on the ground-truth part labels is useful for (i) but not for (ii) (because part labels are not available at test time). Since our goal is to *also* achieve (i), we focus on the conditioned models for (ii) in Tab. 4 and use the non-conditioned models in the other experiments. We have now conducted additional experiments for Tab. 4 using conditioned variants of the `simple` and `iid` models (in addition to the `full` as already in the table) and observed consistent gains (0.4-0.6pp @5cm, UV only).

2: R1: Difference between simple-2D and full. `simple-2D`: assumes per-pixel error vectors to be independent (but not isotropicaly nor identically distributed); `full`: captures the correlation between per-pixel errors.

3: R1: I found the evaluation choices are random. As requested, we have filled some gaps in the tables: For Tab. 1 in the paper, the HG-8stack performance of the `full` model (see Tab. 1 above). For Tab. 4: the performance of all models with uncertainty (see answer 1). For Tab. 5: the performance with tight thresholds with ensembling (similar gains 0.2-0.4pp@2cm, UV only, observed everywhere).

4: R1: Simple-2D... best... in Table 3 with tight thresholds? R2: Simple-2D perform slightly better than the full error model, which however in turn receives a better neg. log-likelihood. Why? In practice, all our models that use uncertainty improve the *average* per-pixel prediction errors (PPE) by a similar amount. However, the `full` model *also* captures the error distribution better (because the errors between different pixels are highly correlated), which is reflected in the higher likelihood but not necessarily reflected in a lower average PPE. This is because average PPE is merely a marginal statistic which ignores the correlations predicted by our models.

5: R1: Is the log-likelihood directly comparable? Yes, all models define a distribution on the same variables.

6: R1: Is the uncertainty not fully correlated to the dense pose performance? See answer 4.

7: R2: do not present the results of related work. R3: The only baseline is based on [13]. We report & outperform the Thrifty DensePose baseline of [12], which is near state-of-the-art for the problem of dense pose recognition (see also table at the top) (Parsing R-CNN is slightly better, but their models are unavailable). In Tab. 1 above, we also compare to the original DensePose-RCNN [5] and additionally report performance using the HRNet architecture (state-of-the-art in pose estimation and semantic segmentation) applied to the dense pose estimation task. In all cases, our models show consistent gains over the whole range of thresholds.

8: R2: Significance of ensembling. Considering that predictions of the ensemble do not significantly differ (as noted in capt. of Tab. 5), which is a necessary condition for better performance, we find the improvement satisfactory.

9: R2: Related... Probabilistic U-Net. Will add & discuss.

10: R2: [does not model] the error between the part label predictions... nor... correlation of errors specific to regions. Model (3) *does* capture the correlations of error vectors within each region via the error term ϵ. Note, in particular, that this term is part-specific, not global. Part-labelling errors are also important, but accounting for them would require a dramatically more complex model due to the resulting switching behaviour.

11: R2: Why learning with an uncertainty model helps training and final performance? See answer 1.

12: R3: "Dense Human Body" by Wei et al.? The "Dense Human Body" is concerned with learning descriptors for matching *pairs* of 3D bodies; DensePose learns instead a map from *any single image* to a 3D model, so they solve different problems and their training setup is also quite different (as it is based on a set of classification problems).

13: R3: Why a Gaussian distribution is a good model? Because errors usually have unimodal distributions and strong linear correlation, so a Gaussian is a reasonable model.