1 Supplementary Materials

Table 1. Details of different baseline approaches considered. The output layers of MCD and DE to predict $\mathcal{N} \sim (\mu, \sigma^2)$ for each of the reference locations in pose $p \in R^{D\times 9}$.This enables the use of Gaussian Negative Log-Likelihood as our loss function and serves to establish a fair comparison with *QAERTS*. For EDL, we keep the default settings as in [1] as this model already accounts for variances without any changes.

Approaches	Description
Monte-Carlo Dropout (MCD) [2]	Bayesian Neural Networks (BNNs) have conventionally been used to formulate uncertainty by defining probability distributions over the model parameters, reducing overfitting. Other approaches aim to overcome the intractability of the posterior distribution, such as MCD. MCD applies Bernoulli dropout before each weighted layer to approximate the aposteriori distribution via variational inference [2].
Deep Ensembles (DE) [4]	This measures uncertainty by training multiple DNNs independently and averaging their outputs at inference time, with considerable computational and time expense. Since Deep Ensembles combine the predictions of M DNNs, the final predictive distribution is assumed as a uniformly weighted mixture of Gaussian distributions. Thus, the ensemble mean is calculated by averaging the output means of M models.
Deep Evidential Regression (EDL) [1]	This is a deterministic method requiring only a single-forward pass through a single model. This is done by placing evidential priors over a Gaussian likelihood function.

References

1. Amini, A., Schwarting, W., Soleimany, A., Rus, D.: Deep evidential regression. Advances in Neural Information Processing Systems **33**, 14927–14937 (2020)
2. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: international conference on machine learning. pp. 1050–1059. PMLR (2016)
3. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7482–7491 (2018)
4. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. Advances in neural information processing systems **30** (2017)

Table 2. Mean results ($\pm$ standard deviation) for quantitative metrics during inference on the testing set across the implemented model during loss ablations study. *MO* indicates a modification where the MVE model has five output heads, each predicting the reference points of the landmark locations. MSE indicates training the model with $\mathcal{L}2$ loss instead of Gaussian Negative Log-Likelihood. Learned Weights (LW) implies the use of multi-task learning with learned homoscedastic uncertainty to weight the MSE loss respective to each transformation instead of a joint loss [3]. K is the dimensionality of the weights, ranging from a scalar value for each task, versus a vector containing scalar values for each reference coordinate. DE+*QAERTS* offers a minor boost to DE.

QAERTS	ED↓	PA↓	MSE↓	NCC↑	SSIM↑	Parameters
MSE	0.33 ± 0.27	0.31 ± 0.13	321.00 ± 300.41	0.30 ± 0.18	0.28 ± 0.17	∼35.89M
MVE-MO	0.36 ± 0.28	0.44 ± 0.28	258.24 ± 283.88	0.56 ± 0.27	0.54 ± 0.30	∼35.90M
$LW_{K=1}$	0.38 ± 0.28	0.35 ± 0.15	395.10 ± 293.67	0.27 ± 0.16	0.17 ± 0.15	∼35.90M
$LW_{K=9}$	0.37 ± 0.27	0.29 ± 0.13	345.74 ± 289.90	0.41 ± 0.17	0.29 ± 0.15	∼35.92M
DE+*QAERTS*	0.30 ± 0.23	0.39 ± 0.25	185.86 ± 230.10	0.75 ± 0.227	0.63 ± 0.25	∼142M

Table 3. Rotational parameterizations and their descriptions.

Parameters	Description
Quaternions	The complete parameterization is described as translation in Euclidean space along with a rotation $\phi_Q := q_0 + iq_1 + jq_2 + kq_3, \phi_Q \in R^4$, where q_0, q_1, q_2 and q_3 are real numbers, and i, j and k are mutually orthogonal imaginary unit vectors respectively. Quaternions are a continuous and smooth representation of rotation laying on a unit manifold. To overcome the challenge of there being two unique values for a single rotation, all quaternion derivations are constrained to a single hemisphere.
Axis-angles	A translation in Euclidean space along with a rotation $\phi_A := \alpha\omega, \phi_A \in R^3$, where ω and α denote a normalized rotation axis and a rotation angle respectively. Axis-angle representations have repetition at 2π radians.
Euler angles	Described as translation in Euclidean space along with a rotation $\phi_E := \alpha\beta\gamma, \phi_E \in R^3$, where α, β, and γ denote *yaw* (around Z axis), *pitch* (around modified Y axis) and *roll* (around modified X axis) respectively. Euler angles wrap around at 2π radians, leading to giving multiple values representing the same angles, indicating they are not injective, and suffer from gimbal lock.
Rotation matrices	($SO(3)$) described as translation in Euclidean space along with a rotation about its three axes $R := R_z R_y R_x, R \in R^{3\times3}$, where each row in R corresponds to coordinates of the rotated axes respectively. The 3×3 rotation matrices are square matrices subject to orthogonality condition $R^T R = I$ and have $det(R) = 1$, making it a member of the special orthogonal Lie group $SO(3)$, but an explicit loss is needed to enforce orthognality during backpropagation.
Translation	Translation displacement parameters are simply the x, y, and z coordinates.
Scaling	The scaling factor is a single scalar multiplied by the plane per dimension ($R^{H\times W\times D}$), where $D = 3$.