From Keypoints to Predictive Distributions: Post-Hoc Uncertainty for YOLO-Pose Models

Anonymous authors
Paper under double-blind review

Abstract

YOLO-Pose models provide efficient keypoint localization, but do not quantify the associated spatial uncertainty. We introduce a lightweight post-hoc probabilistic extension that augments a trained YOLO-Pose model with calibrated bivariate predictive distributions over keypoint locations, centered at the model's original predictions. Concretely, we train additional probabilistic heads with an importance-weighted negative log-likelihood to predict an input-dependent 2×2 dispersion matrix for each keypoint, followed by Gaussian calibration for broad downstream compatibility or Student-t calibration for distributional fidelity. Complementing this, we propose an evaluation protocol that combines a suite of distributional calibration diagnostics with average keypoint precision (AKP), a keypoint-level extension of the COCO AP protocol for assessing reliability rankings. Experiments on COCO show that the learned uncertainty estimates enable effective keypoint-level reliability ranking, Student-t calibration best captures the empirical residual distribution, and uncertainty-based pruning removes unreliable keypoints. A central application-level demonstration is vision-based aircraft landing, where calibrated covariances for runway keypoints support uncertainty-aware aircraft position estimation and downstream sensor fusion.

1 Introduction

YOLO-Pose models (Maji et al., 2022) are widely used in real-time applications because they predict object instances alongside their keypoints in a single forward pass, while benefiting from highly optimized implementations and deployment pipelines (Aharon et al., 2023; Jocher et al., 2025). This makes them attractive whenever low latency and reliable keypoint localization must be achieved under practical deployment constraints. However, each keypoint is represented by a single predicted location, and the instance-level confidence score does not quantify keypoint-specific spatial uncertainty. To support localization reliability assessment and informed keypoint selection or weighting in downstream applications, localization uncertainty must be quantified at the level of individual keypoints.

Within the YOLO family, uncertainty quantification has largely focused on object detection, typically by jointly learning bounding-box predictions and their associated uncertainty (e.g. Kraus & Dietmayer, 2019; Choi et al., 2019). Consequently, these approaches do not provide uncertainty estimates for individual keypoints. Moreover, the joint-learning paradigm is not always desirable. First, many models are already trained, tuned, validated, or deployed, so changing their predictions may require renewed evaluation or certification. Second, replacing a task-specific loss with a likelihood-based objective is not guaranteed to preserve prediction performance. If the assumed residual distribution is misspecified, the induced error weighting may degrade prediction accuracy. In addition, joint mean–variance optimization, for example, can compromise the mean fit unless carefully regularized (Stirn et al., 2023; Seitzer et al., 2022).

Post-Hoc Probabilistic Extension for YOLO-Pose Models. Our formulation separates uncertainty estimation from keypoint localization. Starting from an already trained YOLO-Pose model, we keep its parameters fixed and learn bivariate predictive distributions over keypoint locations, centered at the model's original predictions. To this end, we propose a lightweight post-hoc probabilistic extension that introduces additional heads into the YOLO architecture to estimate a heteroscedastic positive-definite 2×2 dispersion

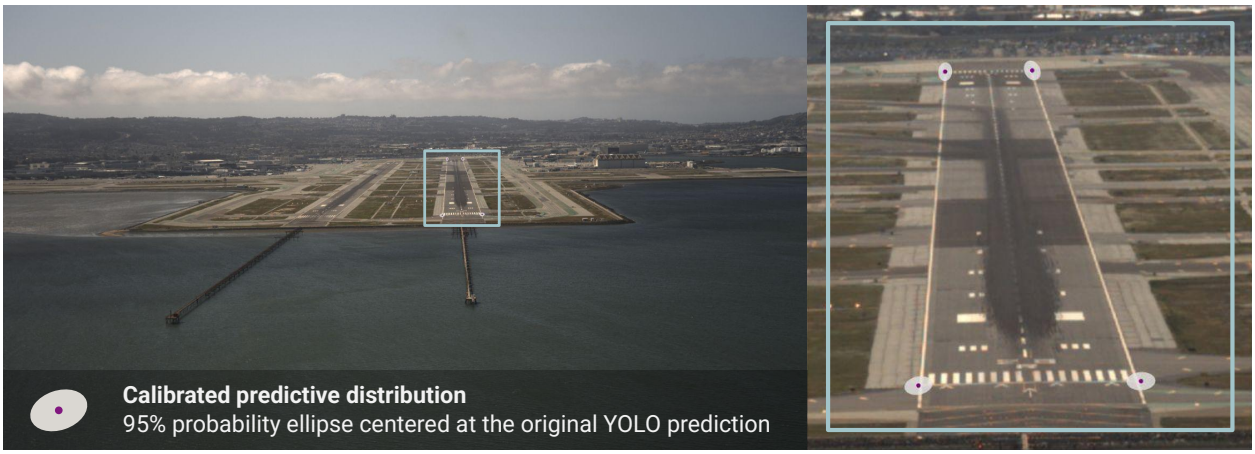


Figure 1: Motivating example: vision-based aircraft landing. Given a trained YOLO-Pose model, our post-hoc probabilistic extension provides calibrated predictive distributions over runway keypoint locations, centered at the model's original predictions. Ellipses denote the corresponding 95% probability regions.

matrix for each keypoint. This compact distributional representation provides input-dependent uncertainty estimates that can be learned from finite data and used directly in downstream processing. An importance-weighted negative log-likelihood shifts dispersion learning toward the prediction distribution expected after inference-time post-processing and targets balanced calibration across object scales.

The input data encountered at inference may differ in domain and image conditions from the data used for dispersion learning; we therefore calibrate the learned dispersion matrices on held-out data representative of the intended operating conditions. Moreover, keypoint residuals can be heavy-tailed due to occlusion, truncation, ambiguity, annotation noise, and the robust, saturating form of the object keypoint similarity loss (Maji et al., 2022) commonly used in YOLO-Pose models. We therefore consider two complementary calibration schemes: Gaussian calibration for broad downstream compatibility with covariance-based methods, and Student-t calibration for improved distributional fidelity to the empirical residual distribution. Both calibration schemes are applied to the same frozen YOLO-Pose model and trained dispersion model. Using calibrated uncertainty estimates as a keypoint-level reliability signal, we further propose uncertainty-based keypoint pruning to remove high-uncertainty keypoints before downstream processing.

Evaluation Protocol for Keypoint Uncertainty Estimates. Complementing the probabilistic extension, we introduce an evaluation protocol for heteroscedastic bivariate Gaussian and Student-t distributions. To this end, we separate uncertainty evaluation into two distinct but related aspects of uncertainty quality: reliability ranking and distributional calibration. First, we introduce average keypoint precision (AKP), a keypoint-level extension of COCO average precision (AP) (Lin et al., 2014), to assess whether uncertainty estimates provide reliability rankings for selecting or pruning keypoints. Second, we evaluate distributional calibration through complementary diagnostics: coverage evaluates nominal probability regions, variance calibration compares predicted covariance magnitudes with empirical squared residuals across uncertainty levels, and quantile–quantile diagnostics reveal distributional shape and tail mismatches.

Experiments. We first benchmark our framework on COCO. The results show that the learned uncertainty estimates provide effective keypoint-level reliability rankings under AKP and achieve strong distributional calibration across the proposed diagnostics, with Student-t calibration most faithfully capturing the heavy-tailed structure of keypoint localization residuals. Beyond calibration, uncertainty-based pruning removes unreliable keypoint predictions before downstream processing. Finally, we demonstrate the relevance of calibrated keypoint uncertainty estimates in vision-based aircraft landing, a real-world safety-critical application, using a large-scale internal aircraft-landing dataset. Fig. 1 visualizes the predicted image-plane covariances for runway keypoints; these covariances support uncertainty-aware Perspective-n-Point and provide the measurement covariances required for downstream sensor fusion.

2 Related Work

Uncertainty Quantification Within the YOLO Family. Uncertainty quantification for YOLO-based models has mainly focused on object detection by augmenting bounding-box predictions with localization uncertainty. Kraus & Dietmayer (2019) develop a probabilistic extension of the YOLOv3 object detector (Redmon & Farhadi, 2018) that predicts variance for bounding-box position and size and captures epistemic uncertainty with Monte Carlo dropout (Gal & Ghahramani, 2016). They show that lower uncertainty correlates with higher overlap between predicted and ground-truth bounding boxes. However, their uncertainty estimates are not calibrated across heads and prior boxes, which limits the use of global calibration metrics. Moreover, their approach modifies and retrains the detector, whereas our method is post hoc and leaves the base detector's mean predictions unchanged. Choi et al. (2019) focus on localization uncertainty by predicting variance for all bounding-box parameters and training the detector with a scale-weighted Gaussian negative log-likelihood loss. They demonstrate that incorporating predicted localization uncertainty into detection confidence can improve the model's AP. He et al. (2019) model bounding-box regression probabilistically and use a Kullback–Leibler loss that allows ambiguous boxes to be assigned larger variance and therefore lower effective regression penalty. Overall, these approaches operate at the bounding-box level and modify the detector by jointly training localization and uncertainty estimates. Such modifications are undesirable when detectors are already trained, tuned, or deployment-constrained, since changing their predictions or training objective can require renewed validation and may compromise the established operating point.

Beyond object detection, likelihood-based objectives have also been explored for keypoint localization. Li et al. (2021) use normalizing flows to model flexible residual densities for improving localization accuracy. By contrast, we target calibrated predictive distributions within prescribed parametric families, yielding explicit covariance or scale matrices that are directly compatible with downstream weighting, uncertainty propagation, and sensor fusion. To the best of our knowledge, our work is the first extension of YOLO-Pose models that provides calibrated predictive distributions over keypoint localization errors without retraining the underlying model or modifying its original predictions, as detailed in Secs. 3.1 and 3.2.

Calibration Evaluation for Probabilistic Regression Models. While a large body of work focuses on probabilistic classification models and their calibration (Guo et al., 2017), the literature on probabilistic regression models is comparatively sparse. For classification models, the expected calibration error (ECE) measures the discrepancy between predicted confidence and the observed frequency of correctness (Guo et al., 2017). For object detection, Küppers et al. (2020) propose a detection-specific extension of ECE that conditions confidence calibration on detection attributes such as location and scale. However, their metric calibrates scalar confidence with respect to detection correctness rather than a continuous predictive distribution over localization parameters. Extending calibration to probabilistic regression, Kuleshov et al. (2018) propose *confidence calibration*, which assesses whether nominal predictive quantiles agree with empirical coverage levels. Song et al. (2019) show that quantile-based calibration alone does not necessarily ensure that the full predictive distribution is calibrated and propose distribution calibration as a stronger criterion. Levi et al. (2022) propose the expected normalized calibration error (ENCE) as a regression analogue of the ECE, comparing predicted and empirical error magnitudes within uncertainty bins. These works, however, primarily focus on scalar confidence calibration, univariate regression, or generic regression settings, whereas keypoint detection additionally requires evaluating the joint calibration of two-dimensional localization distributions across multiple keypoints and detections.

The lack of widely adopted calibration protocols has also shaped prior evaluations of localization uncertainty. Kraus & Dietmayer (2019) evaluate their probabilistic YOLOv3 model's bounding-box uncertainty estimates primarily qualitatively, since the uncertainties are not calibrated across different detection heads. Choi et al. (2019) evaluate the uncertainty estimates of their object detector by plotting the average uncertainty in the bounding-box parameters against the intersection over union (IoU). This indicates whether uncertainty estimates reflect localization error, but not whether they are calibrated as predictive distributions. The authors also modify the confidence of each prediction by a heuristic quantity derived from the model's uncertainty and evaluate the resulting AP. Our keypoint pruning method (Sec. 3.3) provides a principled alternative that does not change the post-processing pipeline and can be applied at the keypoint level instead of the instance level. Küppers et al. (2020) evaluate confidence calibration for object detection with

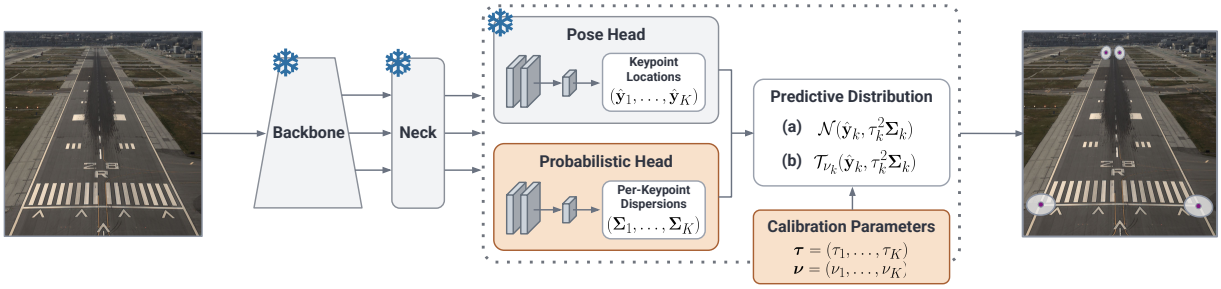


Figure 2: Post-hoc probabilistic extension for YOLO-Pose models. For clarity, only one set of keypoint predictions is shown, and other task heads are omitted. We freeze (❄) a trained YOLO-Pose model and augment each predicted keypoint $\hat{\mathbf{y}}$, used as mean location, with an input-dependent 2×2 dispersion matrix $\mathbf{\Sigma}$ calibrated by a temperature τ and, optionally, Student-t degrees of freedom ν. This yields either **(a)** a Gaussian predictive distribution $\mathcal{N}(\hat{\mathbf{y}}, \tau^2\mathbf{\Sigma})$ for broad downstream compatibility or **(b)** a heavy-tailed Student-t distribution $\mathcal{T}_\nu(\hat{\mathbf{y}}, \tau^2\mathbf{\Sigma})$ for distributional fidelity to the empirical residuals. In the full model, such candidate pose predictions are produced at every spatial location on the neck's multi-resolution feature maps.

a detection-specific extension of ECE, but their metric targets scalar detection confidence rather than a continuous predictive distribution over localization errors.

By contrast, in this work, we evaluate full bivariate predictive distributions over keypoint localization errors, rather than only localization accuracy. Our evaluation protocol combines complementary diagnostics for marginal and joint calibration (Sec. 4.2) with average keypoint precision (AKP) (Sec. 4.1), a keypoint-level metric that evaluates whether the estimated uncertainty reflects the reliability of keypoint predictions.

3 Post-Hoc Probabilistic Extension for YOLO-Pose Models

We assume access to a trained YOLO-Pose model that follows the standard single-stage YOLO design with a backbone–neck–head architecture and predicts keypoint locations, object bounding boxes, and instance-level confidence scores. For keypoint regression, the model is trained using the object keypoint similarity (OKS) loss (Maji et al., 2022). Representative examples include SuperGradients' *YOLO-NAS-Pose* (Aharon et al., 2023) and Ultralytics' *YOLO-Pose* (Jocher et al., 2025). Starting from this baseline, we freeze the original YOLO and attach a lightweight post-hoc probabilistic extension, visualized in Fig. 2. It augments each predicted keypoint $\hat{\mathbf{y}}$, used as mean location, with a heteroscedastic uncertainty estimate in the form of an input-dependent dispersion matrix $\mathbf{\Sigma}$.

Architecture Extension. We propose additional probabilistic heads (see Fig. 2) that mirror the architecture of the original pose heads and reuse the shared multi-resolution neck features to predict a heteroscedastic 2×2 dispersion matrix for each predicted keypoint, constrained to be positive definite via a Cholesky decomposition. The predicted dispersion matrices are index-aligned with the keypoint predictions, ensuring a one-to-one correspondence at each resolution and spatial location. We ablate this design against alternative architectures and simpler isotropic and diagonal dispersion parameterizations in Sec. 5.

Training & Calibration. In Sec. 3.1, we introduce an importance-weighted negative log-likelihood (NLL) for training the probabilistic heads; specifically, we justify using a Gaussian likelihood to learn the conditional dispersion structure. In Sec. 3.2, we propose Gaussian calibration for broad downstream compatibility with covariance-based methods and Student-t calibration for distributional fidelity to the empirical localization errors. After calibration, the predicted dispersion matrix $\mathbf{\Sigma}$ therefore represents a covariance matrix under the Gaussian model and as a scale matrix under the Student-t model.

Inference. At inference time, the post-processing pipeline closely follows the original YOLO design. Our key refinement at this stage is an additional pruning mechanism for uncertain keypoints, described in Sec. 3.3.

3.1 Importance-Weighted NLL for Learning Keypoint Uncertainty

We aim to post-hoc extend a trained YOLO-Pose model with heteroscedastic uncertainty estimates while keeping the YOLO-Pose model frozen and preserving its original predictions. In deep learning, predictive uncertainty is commonly decomposed into an aleatoric and an epistemic component (Kendall & Gal, 2017). In keypoint detection, aleatoric uncertainty primarily arises from irreducible ambiguity in the target annotations, for example due to occlusions, truncation, or subjective labeling. Epistemic uncertainty, by contrast, reflects uncertainty about the predictor itself, e.g., due to incomplete coverage of the deployment domain or suboptimal training. In this work, we do not attempt to disentangle these sources explicitly. Instead, we learn a model of the conditional residual distribution around the keypoint predictions of the frozen YOLO-Pose model, with parameters estimated from training-data residuals and therefore specific to the in-distribution regime defined by the training data. This modeled distribution captures annotation ambiguity, while also reflecting systematic, input-dependent residual structure arising from the imperfect approximation and optimization of the frozen YOLO-Pose model (cf. Jiménez et al., 2025).

3.1.1 A Probabilistic Surrogate for the OKS Loss

We learn a parametric approximation to the conditional residual distribution around the frozen YOLO-Pose model's keypoint predictions. This compact distributional representation can be learned from finite data and yields input-dependent uncertainty estimates that can be used directly in downstream processing. Importantly, this residual distribution is shaped by the keypoint regression objective, which determines how residuals are weighted across keypoints, object scales, and error magnitudes during training (cf. Hooker & Ramsay, 2012; Gneiting, 2011). In YOLO-Pose models, the OKS loss (Maji et al., 2022) has become a common default (cf. Aharon et al., 2023; Jocher et al., 2025). For a single prediction–ground-truth instance pair indexed by i, the OKS is given by

$$\mathrm{OKS} \;=\; \frac{1}{|\mathcal{K}_i|} \sum_{k \in \mathcal{K}_i} \exp\!\left(-\frac{\mathbf{r}_{i,k}^{\top} \mathbf{r}_{i,k}}{2\, s_i^2\, \kappa_k^2} \right), \tag{1}$$

$$\mathcal{L}_{\mathrm{OKS}} \;=\; 1 - \mathrm{OKS}, \tag{2}$$

where $\mathbf{r}_{i,k} \triangleq \mathbf{y}_{i,k} - \hat{\mathbf{y}}_{i,k} \in \mathbb{R}^2$ is the residual between the ground-truth and predicted keypoint locations, s_i^2 corresponds to the object area, and κ_k is a keypoint-specific hyperparameter that accounts for the label noise. Moreover, $\mathcal{K}_i = \big\{ k \in \{1, \ldots, K\} \mid v_{i,k} > 0 \big\}$ denotes the index set of labeled keypoints, where K is the number of keypoints per object and $v_{i,k}$ is a visibility flag to exclude unlabeled keypoints.

The OKS in Eq. (1) can be interpreted as a Gaussian kernel in the keypoint residual, i.e.,

$$\exp\!\left(-\frac{\mathbf{r}_{i,k}^{\top} \mathbf{r}_{i,k}}{2\, s_i^2\, \kappa_k^2} \right) \;\propto\; \mathcal{N}\big(\mathbf{y}_{i,k}; \hat{\mathbf{y}}_{i,k},\; s_i^2 \kappa_k^2 \mathbf{I}_2 \big)\,, \tag{3}$$

and thus induces a scale-normalized squared-distance geometry. This, in turn, suggests a Gaussian likelihood for learning the conditional dispersion structure of the residuals. In Sec. 3.2, we revisit this modeling choice and derive how to calibrate the resulting predictive distribution in order to address distributional misspecification beyond second moments, particularly tail heaviness.

Although the kernel above is isotropic, the induced keypoint residuals can be anisotropic and exhibit cross-axis correlation (e.g., due to structured annotation noise), as we verify experimentally in Sec. 5.3, Tab. 3. We therefore model the uncertainty for each keypoint using an input-dependent full covariance $\mathbf{\Sigma}_{i,k} \in \mathbb{S}^2_{++}$, where $\mathbb{S}^2_{++}$ denotes the set of symmetric positive definite 2×2 matrices, with isotropy as a special case. Accordingly, we adopt a bivariate Gaussian negative log-likelihood (GNLL) per keypoint as a principled probabilistic surrogate for the OKS loss:

$$\mathcal{L}_{\mathrm{GNLL}}(\mathbf{\Sigma}_{i,k}; \mathbf{r}_{i,k}) \;=\; \frac{1}{2}\, \mathbf{r}_{i,k}^{\top}\, \mathbf{\Sigma}_{i,k}^{-1}\, \mathbf{r}_{i,k} + \frac{1}{2} \ln|\mathbf{\Sigma}_{i,k}| + \ln Z_{\mathcal{N}}, \tag{4}$$

where we fix the mean to the frozen YOLO keypoint predictions, i.e., $\boldsymbol{\mu}_{i,k} \triangleq \hat{\mathbf{y}}_{i,k}$, and $\ln Z_{\mathcal{N}} = \ln(2\pi)$ denotes the corresponding normalization constant.

3.1.2 Importance Weighting for Alignment With Deployment Conditions

The residuals $\mathbf{r}_{i,k}$ required for minimizing the GNLL in Eq. (4) are obtained by matching predictions of the frozen YOLO-Pose model to ground-truth instances. To this end, we reuse the training-time label assignment of the underlying YOLO framework to align the probabilistic heads with the frozen YOLO pose heads, while preserving model-specific assignment semantics and providing a rich training signal. This applies to both the standard one-to-many label assignment (cf. Feng et al., 2021) used in most frameworks, including SuperGradients, and the more recent one-to-one label assignment (Wang et al., 2024), as used in Ultralytics' implementation of *YOLO26* (Sapkota et al., 2025). Each matched prediction–ground-truth instance pair defines a set $\mathcal{M}_i$ of associated quantities, comprising the YOLO-Pose predictions, the matched ground-truth annotations, and any derived quantities required by the loss. The training-time label assignment therefore induces an empirical distribution $p_{\text{train}}(\mathcal{M})$ over such matched-pair sets.

At inference time, label assignment is no longer used; instead, the post-processing pipeline filters predictions by confidence-score thresholding and potentially non-maximum suppression (NMS). Post-processing therefore determines which predictions are retained and shapes the residual distribution observed after evaluation-time matching. We denote the corresponding empirical distribution over evaluation-time matched prediction–ground-truth instance pairs by $p_{\text{inference}}(\mathcal{M})$. To obtain reliable uncertainty estimates under the intended deployment conditions, we define the target distribution

$$p_{\text{target}}(\mathcal{M}) \;\propto\; p_{\text{inference}}(\mathcal{M}) \cdot \pi(\mathcal{M}), \qquad \text{with} \qquad \pi(\mathcal{M}) \;\geq\; 0, \tag{5}$$

where $\pi(\mathcal{M})$ encodes optional deployment-relevant importance of a matched-pair set $\mathcal{M}$ in the target distribution. For example, this importance may depend on object scale, occlusion level, or image quality. In this work, we choose $\pi(\mathcal{M})$ to target balanced calibration across object scales, a reported issue in modern detection models (Küppers et al., 2020; Gilg et al., 2023) and a critical requirement for vision-based aircraft landing, where the image-plane runway scale varies substantially during approach, as discussed in Sec. 6.

Our experiments suggest a substantial mismatch, $p_{\text{train}}(\mathcal{M}) \neq p_{\text{target}}(\mathcal{M})$. To compensate for this mismatch and minimize the GNLL under the target distribution, we employ importance weighting (IW):

$$\mathbb{E}_{p_{\text{target}}(\mathcal{M})}\big[\mathcal{L}_{\text{GNLL}}(\mathbf{\Sigma};\mathcal{M})\big] \;\propto\; \mathbb{E}_{p_{\text{train}}(\mathcal{M})}\big[w(\mathcal{M}) \cdot \mathcal{L}_{\text{GNLL}}(\mathbf{\Sigma};\mathcal{M})\big]. \tag{6}$$

We define the importance weights up to normalization as

$$w(\mathcal{M}) \;\propto\; \frac{p_{\text{target}}(\mathcal{M})}{p_{\text{train}}(\mathcal{M})} \;\propto\; \frac{p_{\text{inference}}(\mathcal{M})}{p_{\text{train}}(\mathcal{M})} \cdot \pi(\mathcal{M}), \qquad \text{with} \qquad w(\mathcal{M}) \;\triangleq\; w_{\text{ret}}(\mathcal{M}) \cdot w_{\text{scale}}(\mathcal{M}), \tag{7}$$

where the retention weight $w_{\text{ret}}(\mathcal{M})$ approximates the density-ratio $\frac{p_{\text{inference}}(\mathcal{M})}{p_{\text{train}}(\mathcal{M})}$, and $w_{\text{scale}}(\mathcal{M})$ implements $\pi(\mathcal{M})$ through reweighting by object scale. In this sense, our approach combines covariate-shift importance weighting and group reweighting (Shimodaira, 2000; Sugiyama et al., 2007; Sagawa et al., 2020).

Retention and Object-Scale Importance Weights. An assigned prediction should contribute more strongly when it is likely to be retained after inference-time post-processing. To approximate this retention likelihood, we propose a retention weight for each prediction–ground-truth instance pair matched by the training-time label assignment:

$$w_{\text{ret}}(\mathcal{M}_i) \;\triangleq\; c_{\text{det},i}^{\alpha}, \tag{8}$$

where $c_{\text{det},i}$ is YOLO's instance-level confidence score used to rank and select detections at inference time. Analogous to the well-established anchor alignment metric of Feng et al. (2021), the hyperparameter α controls the strength of confidence-based reweighting in the training objective.

Consistent with prior work by Küppers et al. (2020); Gilg et al. (2023), we observe that smaller objects exhibit noticeably worse uncertainty calibration than larger ones, see Sec. 5.3, Fig. 8. We hypothesize that larger instances have a greater impact on optimization, as detailed in Apps. A.1 and A.2. To counteract this, we introduce a second multiplicative weight based on the object area of the assigned ground truth instance, which yields an approximately log-uniform weighting over object area, as we derive in App. A.3:

$$w_{\text{scale}}(\mathcal{M}_i) \;\triangleq\; \frac{1}{s_i^2}. \tag{9}$$

Importantly, w_{ret} and w_{scale} are heuristic reweighting terms and are not guaranteed to yield an unbiased estimate of $\frac{p_{\text{target}}(\mathcal{M})}{p_{\text{train}}(\mathcal{M})}$. That said, we verify empirically in Sec. 5.3, Tab. 3 that both w_{ret} and w_{scale} improve calibration, with w_{ret} having the greater impact on overall calibration, while w_{scale} reduces the calibration gap between small and large instances, leading to more balanced calibration across object scales.

Importance-Weighted Training Objective. Our post-hoc probabilistic extension, shown in Fig. 2, introduces probabilistic heads that predict $\mathbf{\Sigma}_{i,k} = \mathbf{\Sigma}_k(\mathbf{f}_i; \boldsymbol{\theta}_\Sigma)$ from the neck features $\mathbf{f}_i$ of the underlying YOLO model. To learn their parameters $\boldsymbol{\theta}_\Sigma$, we approximate the objective in Eq. (6) using a mini-batch of matched-pair sets obtained from training-time label assignment on training data. We denote this mini-batch by the indexed family $(\mathcal{M}_i)_{i\in\mathcal{B}}$, where $\mathcal{B}$ is the corresponding mini-batch index set:

$$\mathcal{L}^{\mathcal{B}}_{\text{IW-GNLL}} \;=\; \frac{1}{\sum\limits_{i\in\mathcal{B}} w_i\,|\mathcal{K}_i|} \sum_{i\in\mathcal{B}} w_i \left[\sum_{k\in\mathcal{K}_i} \mathcal{L}_{\text{GNLL}}\big(\mathbf{\Sigma}_k(\mathbf{f}_i;\boldsymbol{\theta}_\Sigma);\mathbf{r}_{i,k}\big)\right]. \tag{10}$$

We minimize $\mathcal{L}^{\mathcal{B}}_{\text{IW-GNLL}}$ with respect to $\boldsymbol{\theta}_\Sigma$ while keeping $\mathbf{f}_i$ and $\mathbf{r}_{i,k}$ fixed, i.e., by freezing the underlying YOLO model. Here, $w_i \triangleq \frac{c^{\alpha}_{\text{det},i}}{s_i^2}$ denotes the assignment-based importance weight (see Eqs. (8) and (9)) and $\mathcal{K}_i$ the corresponding index set of labeled keypoints. The normalization by $\sum_{i\in\mathcal{B}} w_i\,|\mathcal{K}_i|$ yields a self-normalized mini-batch estimate, stabilizing the loss scale across batches.

3.2 Calibrating the Predictive Distribution: Dispersion and Tail Behavior

While Eq. (10) provides a principled objective for learning the conditional dispersion structure, the predictions $\mathbf{\Sigma}_{i,k}$ are not necessarily well calibrated on deployment data. In practice, uncertainty estimates can become overconfident at deployment: residuals observed during training are typically smaller than those encountered under real-world conditions, causing $\mathbf{\Sigma}_{i,k}$ to underestimate inference-time errors, as we show in Sec. 5.1, Fig. 4. Moreover, the residual distribution may exhibit distributional misspecification beyond second moments: extreme errors can occur more frequently than implied by Gaussian tails. As we detail in Sec. 3.2.2, this tail behavior is consistent with the robust, outlier-suppressing structure of the OKS loss.

We therefore propose two complementary baseline calibration methods, depending on downstream distributional assumptions, as shown in Fig. 2. When downstream pipelines require a Gaussian likelihood, we apply *Gaussian variance scaling* to calibrate dispersion while preserving downstream compatibility. When this constraint is absent, we instead perform *Student-t dispersion–tail scaling* to preserve distributional fidelity to the empirical residuals, extending Gaussian variance scaling by jointly calibrating dispersion and tail behavior. Importantly, both calibration regimes share the same frozen YOLO-Pose model and trained dispersion model; inference-time switching only requires selecting the corresponding stored calibration parameters.

3.2.1 Gaussian Calibration for Broad Downstream Compatibility

Gaussian likelihoods are a convenient default for downstream applications due to their broad methodological compatibility. To obtain reliable uncertainty estimates at deployment, we calibrate the trained heteroscedastic dispersion model on a held-out calibration set that reflects the intended operating conditions. A systematic comparison of calibration methods is beyond the scope of this work; we therefore employ *Gaussian variance scaling* (Guo et al., 2017; Levi et al., 2022) as a baseline method:

$$\tilde{\mathbf{\Sigma}}_{i,k}(\tau_k) \;\triangleq\; \tau_k^2\,\mathbf{\Sigma}_{i,k}, \qquad \text{with} \qquad \tau_k \;>\; 0. \tag{11}$$

For calibration, we reuse the training objective in Eq. (10) to learn a global vector of keypoint-specific temperatures, $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_K)$. Applying the training-time label-assignment procedure to the held-out calibration data yields an indexed family of matched-pair sets $(\mathcal{M}_i)_{i\in\mathcal{C}}$, where $\mathcal{C}$ denotes the corresponding calibration index set. During calibration, we keep $\mathbf{r}_{i,k}$ and $\mathbf{\Sigma}_{i,k}$ fixed, i.e., freeze both the underlying YOLO model and our probabilistic heads, and optimize only $\boldsymbol{\tau}$ subject to $\boldsymbol{\tau} \in \mathbb{R}^K_{>0}$ by minimizing

$$\mathcal{L}^{\mathcal{C}}_{\text{IW-GNLL}} \;=\; \frac{1}{\sum\limits_{i\in\mathcal{C}} w_i\,|\mathcal{K}_i|} \sum_{i\in\mathcal{C}} w_i \left[\sum_{k\in\mathcal{K}_i} \mathcal{L}_{\text{GNLL}}\big(\tilde{\mathbf{\Sigma}}_{i,k}(\tau_k);\mathbf{r}_{i,k}\big)\right]. \tag{12}$$

Empirically, Gaussian variance scaling provides a useful yet limited calibration baseline because tail mismatch with the residual distribution remains; see Sec. 5.1, Tab. 1. In App. B, we propose *residual distribution fine-tuning* as a complementary upstream step to improve calibration quality under the Gaussian model.

3.2.2 OKS Loss: Heavy Tails by Design

The limitations of the Gaussian model can be understood from the geometry of the OKS loss and its suppression of large residuals. In particular, the OKS loss in Eq. (2) is not a GNLL: it maximizes a Gaussian kernel similarity rather than minimizing the corresponding negative log-likelihood.

To contrast the two objectives, we adopt the perspective of robust M-estimation and consider the ψ-function, $\psi(\mathbf{r}) \triangleq \nabla_{\mathbf{r}} \mathcal{L}(\mathbf{r})$, which quantifies the influence of a single residual $\mathbf{r}$ on the optimization update (Huber & Ronchetti, 2009; Barron, 2019). Differentiating the OKS loss in Eq. (2) yields

$$\psi_{\mathrm{OKS}}(\mathbf{r}) \;=\; \nabla_{\mathbf{r}} \mathcal{L}_{\mathrm{OKS}} \;\propto\; \exp\!\Big(-\tfrac{\mathbf{r}^\top \mathbf{r}}{2\, s^2\, \kappa^2}\Big)\, \mathbf{r}, \tag{13}$$

up to constant factors. In M-estimation terms, $\psi_{\mathrm{OKS}}(\mathbf{r})$ is *redescending*: for $\|\mathbf{r}\|_2 \to \infty$ its magnitude is exponentially suppressed, such that $\|\psi_{\mathrm{OKS}}(\mathbf{r})\|_2 \to 0$ and gross outliers exert vanishing influence during training. By contrast, under the GNLL in Eq. (4), the per-residual influence is

$$\psi_{\mathrm{GNLL}}(\mathbf{r}) \;=\; \nabla_{\mathbf{r}} \mathcal{L}_{\mathrm{GNLL}} \;=\; \mathbf{\Sigma}^{-1}\mathbf{r}. \tag{14}$$

In the heteroscedastic setting, predicting a larger $\mathbf{\Sigma}$ can reduce $\|\psi_{\mathrm{GNLL}}(\mathbf{r})\|_2$, but this is counterbalanced by the $\ln|\mathbf{\Sigma}|$ term in Eq. (4) and thus cannot reproduce the exponential suppression in $\psi_{\mathrm{OKS}}(\mathbf{r})$. Consequently, $\psi_{\mathrm{GNLL}}(\mathbf{r})$ is *non-redescending*: for $\|\mathbf{r}\|_2 \to \infty$, the influence diverges, $\|\psi_{\mathrm{GNLL}}(\mathbf{r})\|_2 \to \infty$.

The OKS loss therefore induces a robust, outlier-suppressing training signal, reducing the incentive to correct large residuals and thereby making heavier-tailed residual behavior more plausible than under a GNLL. More fundamentally, $\mathcal{L}_{\mathrm{OKS}}$ cannot even be interpreted as a valid NLL, as we formalize in Lemma 1.

Lemma 1 (Non-Normalizability)**.** *Since $\mathcal{L}_{\mathrm{OKS}} \in [0,1)$ is bounded, it holds that $\exp(-\mathcal{L}_{\mathrm{OKS}}) \geq e^{-1}$ for all $\mathbf{r}$. For any $R > 0$, let $B_R = \big\{\mathbf{r} \in \mathbb{R}^2 \,\big|\, \|\mathbf{r}\|_2 \leq R\big\}$. Then*

$$\int_{\mathbb{R}^2} \exp(-\mathcal{L}_{\mathrm{OKS}})\, d\mathbf{r} \;\geq\; \int_{B_R} e^{-1}\, d\mathbf{r} \;=\; e^{-1}\, \pi R^2 \;\xrightarrow{R\to\infty}\; \infty.$$

Hence a putative density $p(\mathbf{r}) \propto \exp(-\mathcal{L}_{\mathrm{OKS}})$ is not normalizable; consequently, $\mathcal{L}_{\mathrm{OKS}}$ cannot be interpreted as a negative log-likelihood on $\mathbb{R}^2$ under the standard identification $p(\mathbf{r}) \propto \exp(-\mathcal{L}(\mathbf{r}))$.

To model OKS-induced residuals with a well-defined likelihood, an elliptically contoured heavy-tailed distribution is a natural choice. In particular, the Student-t likelihood provides a normalizable surrogate that more faithfully captures the observed tail behavior, while also admitting a principled covariance interpretation. We justify this modeling choice in App. C and empirically validate it in Sec. 5.1, Tab. 1.

3.2.3 Student-t Calibration for Distributional Fidelity

To account for the heavy-tailed residual distribution, we calibrate a global tail and scale adjustment under a Student-t likelihood while retaining our Gaussian-trained heteroscedastic dispersion model. The Student-t distribution generalizes the Gaussian by introducing a degrees-of-freedom parameter $\nu \in \mathbb{R}_{>0}$, which controls tail heaviness, and a positive definite scale matrix $\mathbf{S}$. For the bivariate case, the implied covariance is $\mathrm{Cov}[\mathbf{r}] = \frac{\nu}{\nu-2}\,\mathbf{S}$ for $\nu > 2$, whereas it is not defined for $\nu \leq 2$. The corresponding NLL is

$$\mathcal{L}_{t\text{-NLL}}(\nu, \mathbf{S}; \mathbf{r}) \;=\; \frac{\nu+2}{2} \ln\!\left(1 + \frac{1}{\nu}\, \mathbf{r}^\top \mathbf{S}^{-1} \mathbf{r}\right) + \frac{1}{2} \ln|\mathbf{S}| + \ln Z_t(\nu), \tag{15}$$

where $\ln Z_t(\nu) = \ln\Gamma\big(\frac{\nu}{2}\big) - \ln\Gamma\big(\frac{\nu+2}{2}\big) + \ln(\nu\pi)$ is the normalization term, with gamma function $\Gamma(\cdot)$.

The Student-t likelihood admits an equivalent Gaussian scale-mixture representation: $\mathbf{r} \mid \lambda \sim \mathcal{N}(\mathbf{0}, {}^{\mathbf{S}}/\lambda)$ with $\lambda \sim \mathrm{Gamma}(\nu/2, \nu/2)$ yields $\mathbf{r} \sim \mathcal{T}_\nu(\mathbf{0}, \mathbf{S})$ (Kotz & Nadarajah, 2004). Hence, the Student-t distribution remains elliptically contoured: $\mathbf{S}$ sets orientation and anisotropy through the Mahalanobis term $\mathbf{r}^\top \mathbf{S}^{-1} \mathbf{r}$, while

ν controls tail heaviness. This justifies using our Gaussian-trained heteroscedastic dispersion model as an *uncalibrated* Student-t scale model. As a baseline calibration method, we introduce *Student-t dispersion–tail scaling*:

$$\tilde{\mathbf{S}}_{i,k}(\tau_k) \;\triangleq\; \tau_k^2\, \mathbf{\Sigma}_{i,k}, \qquad \text{with} \qquad \tau_k \;>\; 0, \tag{16}$$

$$\tilde{\mathbf{\Sigma}}_{i,k}(\nu_k, \tau_k) \;\triangleq\; \frac{\nu_k}{\nu_k - 2}\, \tilde{\mathbf{S}}_{i,k}(\tau_k), \qquad \text{with} \qquad \nu_k \;>\; 2, \tag{17}$$

where $\nu_k > 2$ ensures a well-defined calibrated covariance $\tilde{\mathbf{\Sigma}}_{i,k}$. Consequently, our approach is a generalization of Gaussian variance scaling (cf. Eq. (11)), which is recovered in the limit $\nu_k \to \infty$.

In order to learn the global degrees of freedom $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_K)$ and temperature $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_K)$, we reuse the importance-weighting approach from Eq. (10) and minimize

$$\mathcal{L}^{\mathcal{C}}_{\text{IW-}t\text{-NLL}} \;=\; \frac{1}{\sum\limits_{i \in \mathcal{C}} w_i\, |\mathcal{K}_i|} \sum_{i \in \mathcal{C}} w_i \left[\sum_{k \in \mathcal{K}_i} \mathcal{L}_{t\text{-NLL}}\big(\nu_k, \tilde{\mathbf{S}}_{i,k}(\tau_k); \mathbf{r}_{i,k}\big) \right] \tag{18}$$

subject to $\boldsymbol{\nu} \in \mathbb{R}^K_{>2}$ and $\boldsymbol{\tau} \in \mathbb{R}^K_{>0}$ on a held-out calibration set, while keeping $\mathbf{r}_{i,k}$ and $\mathbf{\Sigma}_{i,k}$ fixed, i.e., by freezing both the underlying YOLO model and our probabilistic heads.

3.3 Pruning Uncertain Keypoints

Keypoints are typically used as intermediate measurements in downstream pipelines rather than as an end product. In such settings, a single badly localized keypoint can dominate the estimate and lead to severe downstream failures. YOLO-Pose models already provide a rejection mechanism for occluded or truncated keypoints beyond reliable inference (Maji et al., 2022). They predict a per-keypoint confidence score trained on visibility flags $v_{i,k}$, which is thresholded at inference time; we refer to this as *visibility-score thresholding*. Pruning keypoints that pass the visibility-score threshold but are likely to be poorly localized would substantially improve the robustness of downstream pipelines.

Our probabilistic extension provides a principled pruning signal: for each instance and keypoint, we obtain a calibrated heteroscedastic covariance $\tilde{\mathbf{\Sigma}}_{i,k}$ that quantifies input-dependent localization uncertainty. Concretely, in the Gaussian case, we set $\tilde{\mathbf{\Sigma}}_{i,k} = \tau_k^2\, \mathbf{\Sigma}_{i,k}$ (Eq. (11)), whereas for Student-t with $\nu_k > 2$, we convert the scale matrix to a covariance, $\tilde{\mathbf{\Sigma}}_{i,k} = \frac{\nu_k}{\nu_k - 2}\, \tau_k^2\, \mathbf{\Sigma}_{i,k}$ (Eq. (17)). We then prune keypoints with large $\tilde{\mathbf{\Sigma}}_{i,k}$, capturing anisotropy and keypoint-specific ambiguity while exposing a simple reliability–coverage trade-off.

Localization Confidence Score. We convert $\tilde{\mathbf{\Sigma}}_{i,k}$ into a scalar localization confidence score by combining standard covariance scalarizations with an OKS-motivated normalization:

$$c_{i,k} \;\triangleq\; -\frac{\phi\big(\tilde{\mathbf{\Sigma}}_{i,k}\big)}{s_i^2\, \kappa_k^2}, \qquad \phi(\mathbf{\Sigma}) \in \Big\{ \operatorname{tr}(\mathbf{\Sigma}),\; \sqrt{\det(\mathbf{\Sigma})},\; \lambda_{\max}(\mathbf{\Sigma}) \Big\}. \tag{19}$$

By default, we use the total variance $\phi(\mathbf{\Sigma}) = \operatorname{tr}(\mathbf{\Sigma})$, i.e., the sum of eigenvalues, and ablate the alternatives bivariate ellipse-area scale $\sqrt{\det(\mathbf{\Sigma})}$ and worst-direction variance $\lambda_{\max}(\mathbf{\Sigma})$ in Sec. 5.3, Tab. 4. We emphasize that the best scalarization can be dataset- and application-dependent.

Pruning-Threshold Selection. We prune keypoints by thresholding $c_{i,k}$ (Eq. (19)). To select a pruning threshold, we apply evaluation-time matching (see Sec. 4.1) on a held-out dataset, such as the calibration set, yielding an indexed family of matched-pair sets $(\mathcal{M}_i)_{i \in \mathcal{H}}$, where $\mathcal{H}$ denotes the corresponding index set. We then cast pruning as a classification problem by defining the binary label

$$z_{i,k} \;\triangleq\; \mathbb{1}[\text{KS}_{i,k} \geq \gamma_{\text{KS}}]\,, \qquad \text{with} \qquad \text{KS}_{i,k} \;=\; \exp\!\left(-\frac{\mathbf{r}_{i,k}^{\top} \mathbf{r}_{i,k}}{2\, s_i^2\, \kappa_k^2} \right), \tag{20}$$

for labeled keypoints $k \in \mathcal{K}_i$, where $\gamma_{\text{KS}} \in (0, 1)$ denotes a prespecified keypoint similarity (KS) threshold. We then select the pruning threshold γ_c, applied to localization confidence scores $c_{i,k}$, as

$$\gamma_c \;=\; \arg\max_{\gamma}\; J\left(\gamma; \big\{ \big(c_{i,k}, z_{i,k}\big) \,\big|\, i \in \mathcal{H},\; k \in \mathcal{K}_i \big\}\right), \tag{21}$$

where $J(\gamma) = \mathrm{TPR}(\gamma) - \mathrm{FPR}(\gamma)$ is Youden's J statistic, a standard threshold-selection criterion based on the receiver operating characteristic (ROC) curve. We ablate the alternative operating-point criteria minimum distance to the ideal ROC point $(0, 1)$ and maximum F1 score; see Sec. 5.3, Tab. 5.

Inference-Time Pruning. The pruning threshold γ_c is fixed after selection and used at inference time. After post-processing, we retain for each instance i,

$$\mathcal{K}_i^{\mathrm{keep}} \;=\; \left\{ k \in \mathcal{K}_i^{\mathrm{vis}} \,\middle|\, c_{i,k} \geq \gamma_c \right\}, \tag{22}$$

where $\mathcal{K}_i^{\mathrm{vis}}$ denotes the set of keypoints after visibility-score thresholding.

4 Evaluation Protocol for Keypoint Uncertainty Estimates

Keypoint detectors are typically evaluated as point estimators using OKS-based average precision (AP) and average recall (AR), as standardized in the COCO evaluation framework (Lin et al., 2014), which prioritize keypoint localization accuracy over uncertainty quantification. Consequently, there is still no widely accepted protocol for assessing keypoint uncertainty, especially for models that parameterize heteroscedastic, anisotropic predictive distributions.

This gap matters in safety-critical pipelines, where uncertainty informs downstream decisions such as gating, tracking, and sensor fusion. Overconfident estimates risk propagating errors, whereas underconfident estimates can lead to unnecessary rejection or overly conservative behavior, failure modes not revealed by AP/AR alone. To this end, we introduce a new COCO-style metric that evaluates keypoint-level reliability ranking (Sec. 4.1), and propose a set of calibration diagnostics tailored to heteroscedastic keypoint regression under Gaussian and the more general Student-t likelihoods (Sec. 4.2).

4.1 Average Keypoint Precision: A COCO-Style Evaluation Metric

The COCO evaluation framework provides no metric for assessing the quality of per-keypoint confidence scores. The reason is structural: AP/AR evaluate keypoint-detection performance at the instance level based on OKS (Eq. (1)), which aggregates per-keypoint similarity over all labeled keypoints of a detected instance. Consequently, the COCO evaluation protocol has no notion of keypoint-level abstention. In particular, AP/AR cannot distinguish between a detector that always predicts all keypoints and one that correctly ranks poorly detected keypoints for pruning.

This is undesirable for downstream pipelines where fewer but more reliable keypoints are preferable to a complete but noisy pose estimate. To address this limitation, we introduce average keypoint precision (AKP), a COCO-style evaluation metric defined at keypoint level. AKP quantifies how well a confidence score ranks keypoints by correctness.

Evaluation-Time Matching. First, we follow the COCO protocol and apply greedy one-to-one matching between post-processed detections and ground-truth instances on the evaluation data. This yields an indexed family of matched-pair sets $(\mathcal{M}_i)_{i \in \mathcal{E}}$, where $\mathcal{E}$ denotes the corresponding evaluation index set. In contrast to AP/AR, we evaluate AKP only on true-positive detections by requiring the match OKS to exceed the lowest COCO threshold, i.e., OKS ≥ 0.5. This decouples keypoint-level reliability from instance-level mislocalization and thus complements AP/AR.

Keypoint-Level Binary Correctness. We then define binary correctness for labeled keypoints $k \in \mathcal{K}_i$ by reusing the keypoint similarity in Eq. (20):

$$z_{i,k}(\gamma_{\mathrm{KS}}) \;\triangleq\; \mathbb{1}[\mathrm{KS}_{i,k} \geq \gamma_{\mathrm{KS}}]\,. \tag{23}$$

Average Keypoint Precision. Analogous to COCO's AP, our proposed metric is based on the 101-point interpolated average precision (Lin et al., 2014), denoted by AP101. We average AP101 over the standard COCO threshold set, applied here to keypoint similarity. For each keypoint-similarity threshold γ_{KS}, AP_{101}

requires a confidence score $c_{i,k}$ for ranking. In general, $c_{i,k}$ may be any keypoint-level confidence score. In our setting, we use the keypoint-level localization confidence from Eq. (19), which is derived from the calibrated covariances $\tilde{\mathbf{\Sigma}}_{i,k}$. Accordingly, we define

$$\mathrm{AKP} \;=\; \frac{1}{|\mathcal{G}|} \sum_{\gamma_{\mathrm{KS}} \in \mathcal{G}} \mathrm{AP}_{101}\big(\big\{\big(z_{i,k}(\gamma_{\mathrm{KS}}),\, c_{i,k}\big) \;\big|\; i \in \mathcal{E}\,,\, k \in \mathcal{K}_i\big\}\big)\,, \tag{24}$$

where $\mathcal{G} = \{0.50, 0.55, \ldots, 0.95\}$ is the standard COCO threshold set.

4.2 Calibration Diagnostics for Heteroscedastic Keypoint Regression

Calibrated uncertainty estimates are essential when keypoints are used in downstream tasks. A comprehensive evaluation of uncertainty calibration requires multiple complementary diagnostics, particularly in the heteroscedastic regression setting induced by our probabilistic extension in Sec. 3, where predictive uncertainty is input-dependent and may vary substantially across object instances as well as their scales and orientations. Although NLL is a proper scoring rule and the standard training objective for probabilistic models, as a single scalar it conflates accuracy and uncertainty and can therefore obscure whether a model is over- or underconfident, especially under likelihood mismatch, where variance calibration and tail coverage may diverge. We therefore introduce a suite of *coverage*, *quantile–quantile (Q–Q)*, and *variance-calibration* diagnostics, comprising both plots and quantitative metrics.

Evaluation-Time Matching. We use the same evaluation-time matching procedure as in Sec. 4.1 to obtain $\mathcal{E}$, thereby restricting calibration analysis to true-positive instances. This isolates the calibration of aleatoric keypoint-localization uncertainty from instance-level detection failures, for which no meaningful localization residual is defined. Moreover, $\mathcal{E}$ may be restricted to subsets of interest, e.g., specific object-scale regimes, while $\mathcal{K}_i$ may be restricted to a single keypoint category to enable per-keypoint calibration analysis.

Coverage Diagnostics. Coverage diagnostics assess whether nominal confidence levels match empirical frequencies (Gneiting et al., 2007; Kuleshov et al., 2018). A model is perfectly calibrated if the empirical coverage $\hat{C}(\alpha)$ at nominal coverage level α satisfies $\hat{C}(\alpha) = \alpha$ for all $\alpha \in (0, 1)$. For our bivariate Gaussian and Student-t predictive distributions, we compute the empirical joint coverage as

$$\hat{C}_{\mathcal{N}}(\alpha) \;=\; \frac{1}{\sum_{i \in \mathcal{E}} |\mathcal{K}_i|} \sum_{i \in \mathcal{E},\, k \in \mathcal{K}_i} \mathbb{1}\Big[\mathbf{r}_{i,k}^{\top} \tilde{\mathbf{\Sigma}}_{i,k}^{-1} \mathbf{r}_{i,k} \le \chi^2_{2,\alpha}\Big]\,, \tag{25}$$

$$\hat{C}_{\mathcal{T}}(\alpha) \;=\; \frac{1}{\sum_{i \in \mathcal{E}} |\mathcal{K}_i|} \sum_{i \in \mathcal{E},\, k \in \mathcal{K}_i} \mathbb{1}\Big[\mathbf{r}_{i,k}^{\top} \tilde{\mathbf{S}}_{i,k}^{-1} \mathbf{r}_{i,k} \le 2\, F^{-1}_{2,\nu_k}(\alpha)\Big]\,, \tag{26}$$

where $\chi^2_{2,\alpha}$ and $F^{-1}_{2,\nu_k}(\alpha)$ denote the corresponding α-quantiles of the χ^2 and F distributions with the appropriate degrees of freedom. The corresponding definitions of marginal empirical coverage for each standardized coordinate are provided in App. D.1. We visualize joint and marginal calibration using coverage plots over a grid of nominal coverage levels, e.g., $\mathcal{A} = \{0.01, 0.02, \ldots, 0.99\}$, and summarize deviations using the average coverage error (ACE),

$$\mathrm{ACE} \;=\; \frac{1}{|\mathcal{A}|} \sum_{\alpha \in \mathcal{A}} \Big|\hat{C}(\alpha) - \alpha\Big|\,, \tag{27}$$

where lower values indicate better calibration. The ACE corresponds to a discrete approximation of the area between the observed coverage curve and the diagonal of perfect calibration. Further details on coverage diagnostics are provided in App. D.1.

Q–Q Goodness-Of-Fit Analysis. Complementary Q–Q analysis assesses whether the full distribution of normalized residual statistics agrees with the corresponding reference law, revealing shape and tail mismatches beyond confidence-region probabilities alone (Wilk & Gnanadesikan, 1968). It is defined analogously to coverage diagnostics based on the same normalized residual statistics; full details are provided in App. D.2. We perform Q–Q analysis jointly and marginally, and report the resulting goodness-of-fit using Q–Q plots.

Variance-Calibration Diagnostics. Coverage and Q–Q diagnostics can indicate perfect calibration even when the predicted uncertainty is uninformative in the heteroscedastic sense (Levi et al., 2022). Variance-calibration diagnostics therefore complement coverage analysis by assessing whether the calibrated covariance predictions $\tilde{\mathbf{\Sigma}}_{i,k}$, obtained either from the Gaussian model (Eq. (11)) or the Student-t model (Eq. (17)), match empirical squared residuals across the range of predicted uncertainty levels. For our multivariate setting, the global moment-matching condition is $\mathbb{E}[\mathbf{r}\mathbf{r}^\top] \approx \mathbb{E}[\tilde{\mathbf{\Sigma}}]$.

For a stratified evaluation across uncertainty levels, we sort predictions by their uncertainty magnitude and compare predicted variances and empirical squared residuals within bins. To this end, we use the total variance $\mathrm{tr}(\tilde{\mathbf{\Sigma}}_{i,k})$ as a scalar uncertainty score for joint variance calibration, analogous to the marginal definitions of Levi et al. (2022), which are detailed in App. D.3. This choice is natural because the trace is a rotation-invariant aggregate of the marginal variances and thus the multivariate counterpart of the scalar variance used by Levi et al. (2022). Accordingly, we sort all evaluated keypoints by $\mathrm{tr}(\tilde{\mathbf{\Sigma}}_{i,k})$ and use (n) for the corresponding sorted index such that $\mathrm{tr}\big(\tilde{\mathbf{\Sigma}}_{(1)}\big) \leq \cdots \leq \mathrm{tr}\big(\tilde{\mathbf{\Sigma}}_{(N)}\big)$, where $N = \sum_{i \in \mathcal{E}} |\mathcal{K}_i|$. We then partition the sorted samples into B equally sized bins $\{\mathcal{B}_b\}_{b=1}^{B}$ and define the mean empirical squared residuals and mean predicted total variance in bin $\mathcal{B}_b$ as

$$\hat{e}_b \;=\; \frac{1}{|\mathcal{B}_b|} \sum_{n \in \mathcal{B}_b} \frac{1}{2} \|\mathbf{r}_{(n)}\|_2^2, \qquad \hat{v}_b \;=\; \frac{1}{|\mathcal{B}_b|} \sum_{n \in \mathcal{B}_b} \frac{1}{2} \,\mathrm{tr}\big(\tilde{\mathbf{\Sigma}}_{(n)}\big)\,, \tag{28}$$

where the factor $\frac{1}{2}$ puts the joint quantity on the same scale as the marginal variance-calibration diagnostics. We visualize variance calibration by plotting $\hat{e}_b$ against $\hat{v}_b$, where perfect calibration is reflected by points lying close to the diagonal $\hat{e}_b = \hat{v}_b$ for all $b \in \{1, 2, \ldots, B\}$. For a scalar summary, we follow Levi et al. (2022) and compute the expected normalized calibration error (ENCE) on the corresponding root-mean quantities,

$$\mathrm{ENCE} \;=\; \frac{1}{B} \sum_{b=1}^{B} \frac{|\sqrt{\hat{e}_b} - \sqrt{\hat{v}_b}|}{\sqrt{\hat{v}_b}}. \tag{29}$$

Lower values indicate better agreement between predicted and empirical variance across uncertainty levels. Further details on joint and marginal variance-calibration diagnostics are provided in App. D.3.

5 Benchmark Experiments on COCO

We evaluate our probabilistic extension on COCO (Lin et al., 2014) to assess its behavior in a standardized and widely used multi-person pose-estimation setting. COCO provides diverse object scales, poses, occlusions, and image conditions, making it well suited for evaluating keypoint uncertainty across a broad distribution of localization errors. We split the original validation set into calibration and evaluation subsets at a 20/80 ratio; all reported metrics are computed on the evaluation subset.

As the underlying YOLO-Pose model, we use SuperGradients' *YOLO-NAS-Pose*, which provides a strong keypoint detection baseline and a practical basis for open-sourcing our probabilistic extension[1]. We evaluate both the Nano and Large variants, representing the most computationally efficient and strongest-performing models, respectively. In all cases, the YOLO-Pose model is kept frozen and only the additional probabilistic heads are trained and calibrated, such that the original keypoint predictions remain unchanged.

We first report the core benchmark results in terms of uncertainty calibration diagnostics and average keypoint precision in Sec. 5.1. We then evaluate whether uncertain keypoints can be identified and pruned reliably, and how this impacts calibration, in Sec. 5.2. Finally, Sec. 5.3 presents ablation studies on the importance-weighting scheme, covariance parameterization, probabilistic-head architecture, and uncertainty scalarization to support the main design choices of our approach.

5.1 Core Results

Qualitative examples of the predicted keypoint distributions are shown in Fig. 3. The uncertainty ellipses visualize the spatial uncertainty estimated by the probabilistic heads and reflect the local ambiguity of the

[1]SuperGradients is Apache-2.0 licensed, supporting open-source redistribution.

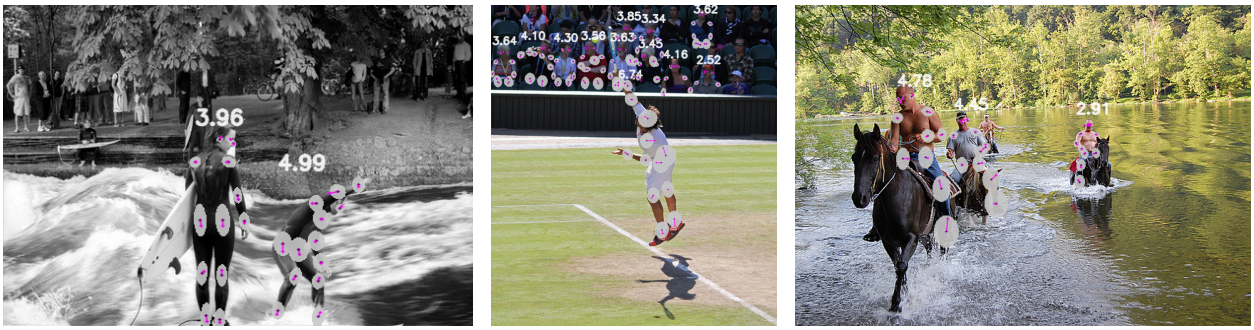


Figure 3: Example keypoint predictions produced by our probabilistic extension. The gray ellipses enclose 95% of the probability mass of the predicted keypoint distributions, while the purple lines visualize the residuals between predicted and ground-truth keypoints. The displayed values represent the associated NLLs. Aggregate calibration metrics are summarized in Tab. 1, corresponding calibration diagnostics are shown in Fig. 4, and per-keypoint calibration metrics are reported in Fig. 5.

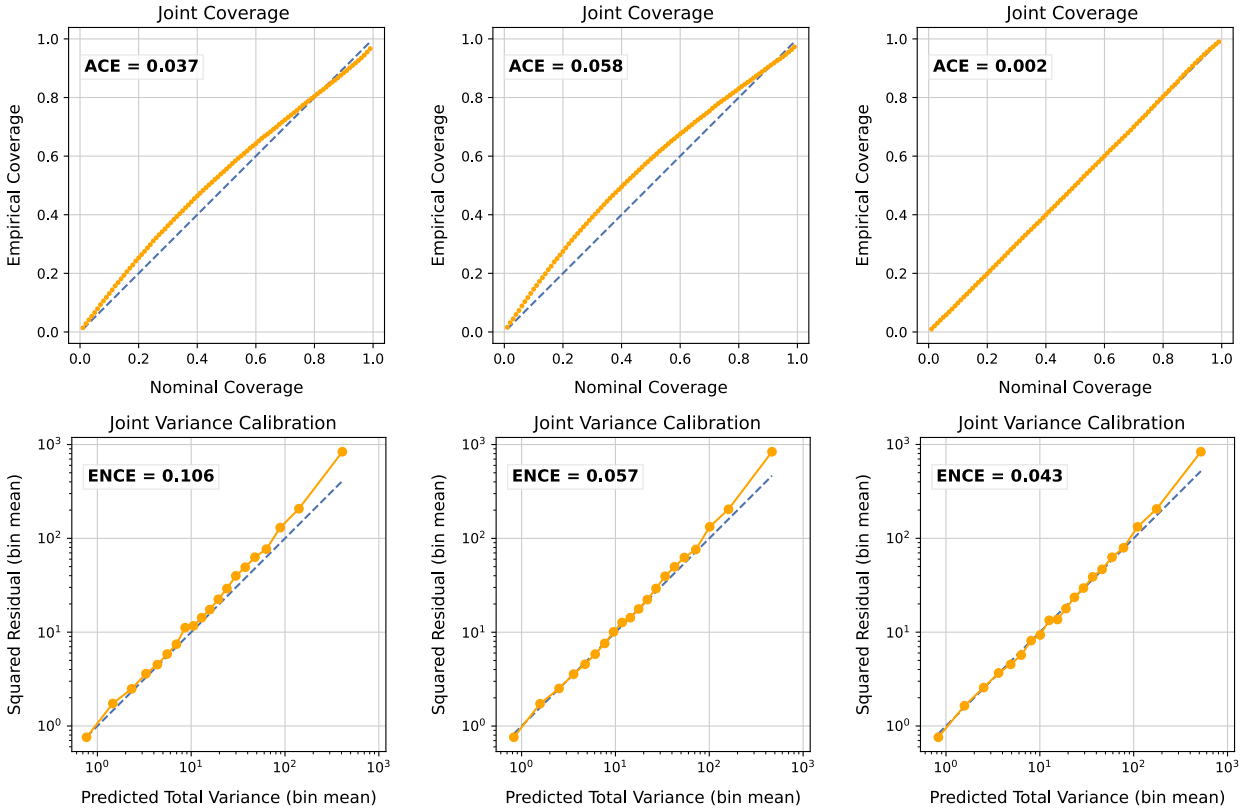
Table 1: Core uncertainty metrics on COCO for YOLO-NAS-Pose Nano and Large. We compare the uncalibrated dispersion model against Gaussian and Student-t calibration. The corresponding full calibration diagnostics are provided in Fig. 4 and App. E, Figs. 13 and 14; per-keypoint calibration metrics are reported in Fig. 5. For reference, we additionally report AKP_{rand}, i.e., the AKP of a random ranking.

Model size	Calibration method	ACE ↓	ENCE ↓	NLL ↓	AKP ↑	AKP_{rand} ↑
Nano	Uncalibrated dispersion model	0.0374	0.1061	5.67	0.935	0.791
	Gaussian calibration	0.0582	0.0570	5.62	0.935	0.791
	Student-t calibration	**0.0016**	**0.0434**	**5.50**	0.935	0.791
Large	Uncalibrated dispersion model	0.0252	0.1538	5.47	0.948	0.831
	Gaussian calibration	0.0602	0.0813	5.42	0.948	0.831
	Student-t calibration	**0.0020**	**0.0770**	**5.27**	0.948	0.831

corresponding keypoints. The core quantitative results are summarized in Tab. 1 for YOLO-NAS-Pose Nano and Large. We compare the uncalibrated dispersion model, Gaussian calibration, and Student-t calibration; the keypoint mean predictions remain unchanged by design. Thus, differences in NLL, ACE, ENCE, and AKP are attributable to the uncertainty estimates rather than changes in keypoint localization accuracy.

Gaussian calibration improves the predicted covariances primarily in terms of second-moment calibration, reducing ENCE for both Nano and Large. At the same time, ACE is larger than for the uncalibrated dispersion model, highlighting that coverage calibration and variance calibration capture complementary aspects of predictive uncertainty under distributional misspecification. Student-t calibration better aligns with both diagnostics by combining calibrated covariance scale with heavier tails. It substantially reduces ACE and also improves or matches ENCE for both model sizes while also improving NLL. Thus, Student-t calibration yields the best overall probabilistic performance, indicating that the empirical residual distribution is better captured by a heavy-tailed Student-t distribution than by a Gaussian approximation. The AKP values remain essentially unchanged across calibration methods for a fixed model size. This is expected because AKP evaluates the uncertainty-based ranking of keypoints, whereas global calibration primarily rescales or reshapes the predictive distributions without changing the underlying keypoint predictions. The corresponding random baselines, AKP_{rand}, are substantially lower, confirming that the predicted uncertainties contain meaningful information about keypoint reliability.

Figure 4 visualizes this behavior for YOLO-NAS-Pose Nano. The uncalibrated dispersion model exhibits a systematic variance-calibration error: predicted variances are consistently smaller than the empirical squared residuals, indicating overconfident second moments. Gaussian temperature scaling largely corrects this covariance-scale mismatch, as reflected by the improved variance-calibration curve. However, because the



(a) Uncalibrated dispersion model (b) Gaussian calibration (c) Student-t calibration

Figure 4: Calibration diagnostics on COCO for our probabilistic extension of YOLO-NAS-Pose Nano, comparing the uncalibrated dispersion model, Gaussian calibration, and Student-t calibration. The corresponding marginal calibration diagnostics, as well as the joint and marginal calibration diagnostics for YOLO-NAS-Pose Large, can be found in App. E, Figs. 13 to 15.

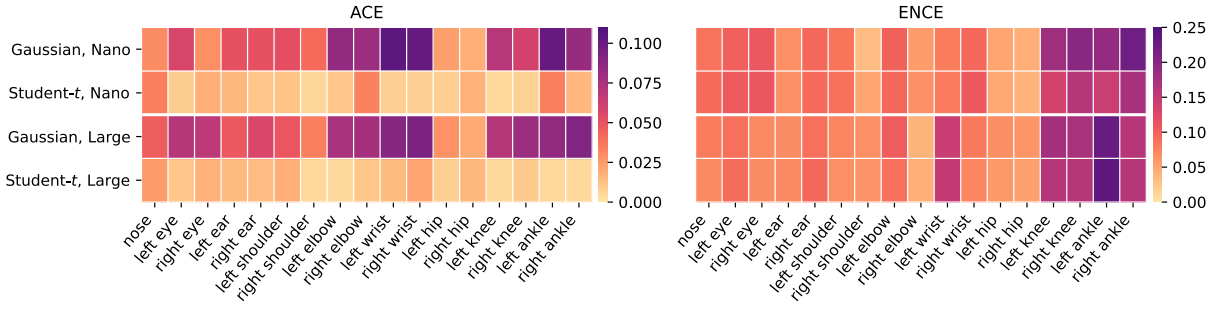


Figure 5: Per-keypoint calibration metrics on COCO. ACE and ENCE are reported for each keypoint class under Gaussian and Student-t calibration for YOLO-NAS-Pose Nano and Large. Lower values indicate better calibration. Student-t calibration improves calibration for most keypoint classes, while calibration quality remains keypoint-dependent, with particularly poor variance calibration for knees and ankles.

empirical residuals deviate from the assumed Gaussian shape, improving the covariance scale does not necessarily improve joint coverage. Indeed, the uncalibrated model can achieve a lower Gaussian coverage error due to compensating effects between overconfident second moments and distributional misspecification. Student-t calibration mitigates this mismatch by combining calibrated covariance scale with heavier tails, yielding near-nominal coverage and improved variance calibration. The corresponding marginal calibration

diagnostics, as well as the joint and marginal calibration diagnostics for YOLO-NAS-Pose Large, can be found in App. E, Figs. 13 to 15.

Finally, the per-keypoint calibration results in Fig. 5 show that calibration quality varies substantially across keypoint classes, with extremities such as knees and ankles remaining more challenging than more stable body keypoints. We therefore evaluate in Sec. 5.2 whether predictive uncertainty can be used to identify and remove unreliable keypoint predictions, thereby improving calibration.

5.2 Pruning Uncertain Keypoints

We next evaluate whether the predicted uncertainties can be used as a post-hoc reliability criterion for selective keypoint prediction. Following Sec. 3.3, we prune keypoints whose predictive uncertainty exceeds the selected threshold and compute the metrics on the retained predictions. This removes individual unreliable keypoints while leaving the retained predictions unchanged. Qualitative examples in Fig. 6 show that pruning primarily suppresses predictions with large spatial uncertainty relative to the object scale. Such cases often arise from occlusions, overlapping persons, or left–right ambiguities, particularly for lower-body keypoints where confused limbs can lead to misplaced knees and ankles.

The quantitative results in Tab. 2 show that pruning consistently improves NLL across all calibrated models, indicating that high-uncertainty keypoints contribute disproportionately to poor likelihood values. The effect on calibration is more metric-dependent: pruning generally improves variance calibration, whereas changes in ACE are smaller and less uniform. For Student-t calibration, pruning can slightly degrade ACE, as the heavy-tailed likelihood already accommodates large residuals and selective removal perturbs the calibrated tail behavior of the retained distribution. This suggests that uncertainty-based pruning

(a) Before keypoint pruning

(b) After keypoint pruning

Figure 6: Example keypoint predictions before and after pruning uncertain keypoints. The gray ellipses enclose 95% of the probability mass of the predicted keypoint distributions, while the purple lines visualize the distance between prediction and ground truth. The displayed value corresponds to the associated NLLs. We observe that left and right legs are frequently confused, leading to misplaced ankle and knee predictions. Uncertainty-based keypoint pruning effectively mitigates these ambiguities, leading to improved calibration for ankles and knees, cf. Fig. 7.

Table 2: Calibration improvements from pruning uncertain keypoints on COCO for YOLO-NAS-Pose Nano and Large. The reported metric values are obtained after applying uncertainty-based keypoint pruning introduced in Sec. 3.3. Changes are computed relative to the corresponding models without keypoint pruning reported in Tab. 1; negative changes indicate improvement. The corresponding calibration plots are shown in App. E, Figs. 16 and 17, while changes in the per-keypoint calibration metrics are reported in Fig. 7.

Model size	Calibration method	ACE ↓		ENCE ↓		NLL ↓	
		Pruned	Change	Pruned	Change	Pruned	Change
Nano	Gaussian calibration	0.0539	**−0.0043**	0.0467	**−0.0103**	5.35	**−0.27**
	Student-t calibration	0.0049	+0.0033	0.0488	+0.0054	5.25	**−0.25**
Large	Gaussian calibration	0.0604	+0.0002	0.0608	**−0.0205**	5.20	**−0.22**
	Student-t calibration	0.0040	+0.0020	0.0560	**−0.0210**	5.08	**−0.19**

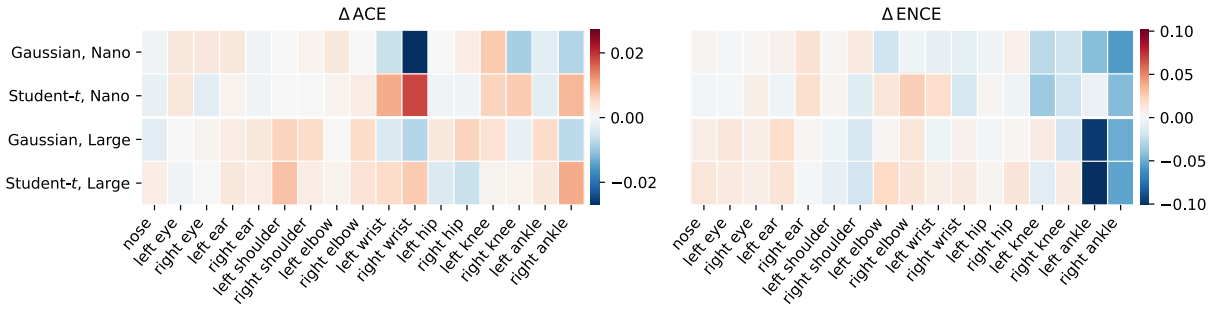


Figure 7: Per-keypoint impact of keypoint pruning on COCO calibration metrics, corresponding to the unpruned per-keypoint results in Fig. 5. We report Δ ACE and Δ ENCE for each keypoint under Gaussian and Student-t calibration for YOLO-NAS-Pose Nano and Large. Negative values indicate improved calibration after pruning, while positive values indicate degradation. Our proposed keypoint-pruning method notably improves variance calibration for knees and ankles by suppressing unreliable predictions, as shown in Fig. 6.

primarily removes predictions with poorly matched residual magnitude and covariance scale, rather than the full distributional shape, while coverage calibration is less directly affected by selective removal. The corresponding full calibration diagnostics are provided in App. E, Figs. 16 and 17.

The per-keypoint analysis in Fig. 7 confirms that the largest improvements occur for knees and ankles, where pose ambiguity is most pronounced, consistent with the qualitative examples in Fig. 6. Overall, uncertainty-based pruning provides a simple mechanism for suppressing unreliable keypoint predictions, particularly in cases of limb confusion, and improving the calibration of the retained pose estimates.

5.3 Ablation Studies

We ablate the main components of our probabilistic extension on COCO. Unless stated otherwise, all variants are evaluated after Gaussian calibration, such that differences reflect the underlying training objective, dispersion parameterization, and probabilistic-head design rather than differences in post-hoc calibration.

Tab. 3 summarizes the main ablation results for YOLO-NAS-Pose Nano and Large. The default configuration provides the best overall trade-off across ACE, ENCE, NLL, and AKP. Removing object-scale importance weights (see Eq. (9)) degrades calibration, while omitting importance weighting altogether, i.e., both object-scale and retention weights (see Eq. (8)), leads to a further degradation. This supports the proposed weighting scheme for learning calibrated uncertainty estimates across both model sizes and metrics. The analysis in Fig. 8 further shows that object-scale importance weights lead to more balanced calibration across object scales and substantially improve calibration for smaller objects.

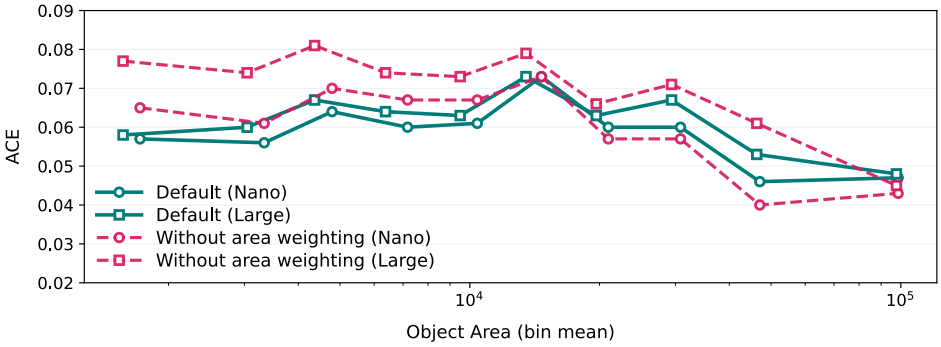


Figure 8: Calibration across object scales on COCO. We ablate object-area importance weighting and show that it balances calibration across object sizes, substantially improving calibration for smaller objects.

Table 3: Ablation study on COCO for YOLO-NAS-Pose Nano and Large. All configurations are evaluated after Gaussian calibration. Our default configuration uses full heteroscedastic dispersion matrices, the importance-weighted GNLL objectives for training and calibration defined in Eqs. (10) and (12), and the proposed probabilistic head architecture introduced in Fig. 2.

Model size	Ablation	ACE ↓	ENCE ↓	NLL ↓	AKP ↑
Nano	**Default configuration**	**0.0582**	0.0570	**5.62**	0.935
	No object-scale importance weights ($w_{\text{scale}} = 1$)	0.0596	0.0579	5.64	**0.936**
	No importance weighting ($w_{\text{ret}} = 1, w_{\text{scale}} = 1$)	0.0689	0.0637	5.64	0.935
	Diagonal dispersion parametrization	0.0617	0.0629	5.67	0.934
	Isotropic dispersion parametrization	0.0652	0.0603	5.71	0.935
	Last-layer probabilistic heads	0.1017	**0.0536**	5.85	0.916
Large	**Default configuration**	**0.0602**	0.0813	**5.42**	**0.948**
	No object-scale importance weights ($w_{\text{scale}} = 1$)	0.0699	0.0861	5.43	**0.948**
	No importance weighting ($w_{\text{ret}} = 1, w_{\text{scale}} = 1$)	0.0796	0.0868	5.43	0.947
	Diagonal dispersion parametrization	0.0658	0.0778	5.44	**0.948**
	Isotropic dispersion parametrization	0.0678	**0.0762**	5.48	**0.948**
	Last-layer probabilistic heads	0.0770	0.0811	5.56	0.933

Table 4: Ablation study on COCO of covariance scalarization methods for deriving the keypoint uncertainty scores used by AKP. All variants use the default configuration and differ only in the scalarization method.

Scalarization	AKP ↑			
	Nano		Large	
	Gaussian	Student-t	Gaussian	Student-t
Trace $\mathrm{tr}(\boldsymbol{\Sigma})$ (**default**)	**0.935**	**0.935**	**0.948**	**0.948**
Determinant $\sqrt{\det(\boldsymbol{\Sigma})}$	**0.935**	0.934	**0.948**	**0.948**
Max-eigenvalue $\lambda_{\max}(\boldsymbol{\Sigma})$	0.933	0.933	0.946	0.946

Table 5: Threshold-selection ablation for uncertainty-based keypoint pruning on COCO. We compare Youden's J, the minimum-distance criterion to the ideal ROC point, and maximum F1 selection. Lower values indicate better calibration and likelihood.

Model size	Threshold selection	ACE ↓		ENCE ↓		NLL ↓	
		Gaussian	Student-t	Gaussian	Student-t	Gaussian	Student-t
Nano	Youden's J (**Default**)	**0.0539**	0.0049	0.0467	0.0488	**5.35**	**5.25**
	Min. distance to $(0, 1)$	0.0540	0.0049	**0.0465**	0.0488	5.36	**5.25**
	Max. F1	0.0580	**0.0016**	0.0584	**0.0446**	5.65	5.50
Large	Youden's J (**Default**)	**0.0604**	0.0040	**0.0608**	**0.0560**	**5.20**	**5.08**
	Min. distance to $(0, 1)$	**0.0604**	0.0040	**0.0608**	**0.0560**	**5.20**	**5.08**
	Max. F1	0.0602	**0.0021**	0.0815	0.0771	5.42	5.26

The dispersion and architecture ablations support the proposed design choices. Restricting the predicted covariance to diagonal or isotropic form generally degrades probabilistic performance, particularly in terms of NLL, showing that anisotropic uncertainty is beneficial for keypoint localization. For the isotropic parameterization, the calibration diagnostics in App. E, Fig. 19, further show that acceptable joint variance calibration can mask poor marginal calibration along the individual coordinate axes, highlighting the im-

portance of evaluating both joint and marginal diagnostics. Replacing the proposed probabilistic heads with last-layer heads, which take the penultimate pose-head features as input and use only a single nonlinear prediction layer, leads to a clear degradation. This suggests that uncertainty estimation benefits from the additional feature processing capacity of the proposed probabilistic heads.

Tab. 4 compares covariance scalarizations for deriving the keypoint uncertainty scores used by AKP. Trace, determinant, and maximum-eigenvalue scalarization yield very similar results, indicating that AKP is robust to the specific scalarization choice in this setting. We therefore use the trace as the default because it is simple and directly reflects the total predicted marginal variance.

Finally, Tab. 5 compares threshold-selection strategies for uncertainty-based keypoint pruning. Youden's J statistic and the minimum-distance-to-ideal-ROC-point criterion select non-trivial pruning thresholds and yield similar results, indicating that the pruning behavior is not driven by a fragile operating-point choice. By contrast, maximizing F1 removes only a few keypoints, resulting in metrics close to the unpruned baseline and making it less suitable for inducing meaningful selective prediction. We therefore use Youden's J as the default balanced criterion for separating reliable from unreliable keypoint predictions.

6 Vision-Based Aircraft Landing

Vision-based landing (VBL) refers to vision-based runway landing guidance or navigation during approach and landing, where visual information from an onboard camera is used to support runway-relative guidance or position estimation (Balduzzi et al., 2021). As detailed in App. F.1, VBL provides a complementary landing-navigation modality that is independent of the ground-based infrastructure required by the instrument landing system (ILS) and does not rely directly on the global navigation satellite system (GNSS). In our system, the aircraft's 3D position relative to the runway is estimated from detected image-plane keypoints of runway landmarks with known georeferenced coordinates, such as the four runway corners. The resulting vision-based position estimate can then be used as an independent measurement source within an extended Kalman filter (EKF)-based sensor-fusion module, as described by Roumeliotis & Bekey (2000).

Since this measurement is fused in an extended Kalman filter (EKF)-based sensor-fusion module, uncertainty quantification is a functional requirement rather than a diagnostic add-on: each vision update must be associated with a well-calibrated covariance estimate compatible with the Gaussian measurement model. Overconfident covariances can cause the filter to overweight unreliable vision updates, whereas underconfident covariances reduce the contribution of the vision sensor to the fused state estimate. This is consequential in civil aviation due to certification requirements for deep learning-based systems (European Union Aviation Safety Agency, 2024a;b; 2025; Federal Aviation Administration, 2024).

VBL also imposes strict real-time constraints, making the single-stage YOLO architecture attractive for runway-keypoint localization. In this setting, uncertainty estimates must not compromise the latency and prediction interface of the underlying YOLO-Pose model. Our post-hoc probabilistic extension is designed for this constraint: it keeps the YOLO-Pose model frozen, preserves its original keypoint predictions, and adds only lightweight probabilistic heads for calibrated covariance prediction, as shown in Fig. 2.

System Details. An overview of the VBL system is shown in Fig. 9. In the following, we consider a single detected runway and omit the instance index i for clarity. In this application, we restrict attention to the Gaussian model, since the downstream navigation pipeline is formulated around Gaussian uncertainty representations. The VBL model predicts, for each retained keypoint $k \in \mathcal{K}^{\text{keep}}$ (see Eq. (22)), a 2D keypoint location $\hat{\mathbf{y}}_k \in \mathbb{R}^2$ together with a calibrated covariance $\tilde{\mathbf{\Sigma}}_k \in \mathbb{S}^2_{++}$, where $\mathbb{S}^2_{++}$ denotes the set of symmetric positive definite 2×2 matrices. We estimate the aircraft's 3D position $\hat{\mathbf{t}} \in \mathbb{R}^3$ by solving a covariance-weighted Perspective-n-Point (PnP) problem (Ferraz et al., 2014) from $\hat{\mathbf{y}}_k$, $\tilde{\mathbf{\Sigma}}_k$, and the georeferenced runway keypoint coordinates; see App. F.2. We then propagate the image-plane uncertainty through the PnP mapping using a first-order linearization (Valentin et al., 2024), yielding a 3D position covariance $\mathbf{\Sigma_t} \in \mathbb{S}^3_{++}$; see App. F.3. The resulting pair $(\hat{\mathbf{t}}, \mathbf{\Sigma_t})$ is subsequently fused with other navigation sensors; see App. F.4. VBL therefore provides a realistic application scenario for our central claim: calibrated keypoint uncertainty is valuable not only for image-plane diagnostics, but because it directly affects the reliability of the downstream pipeline.

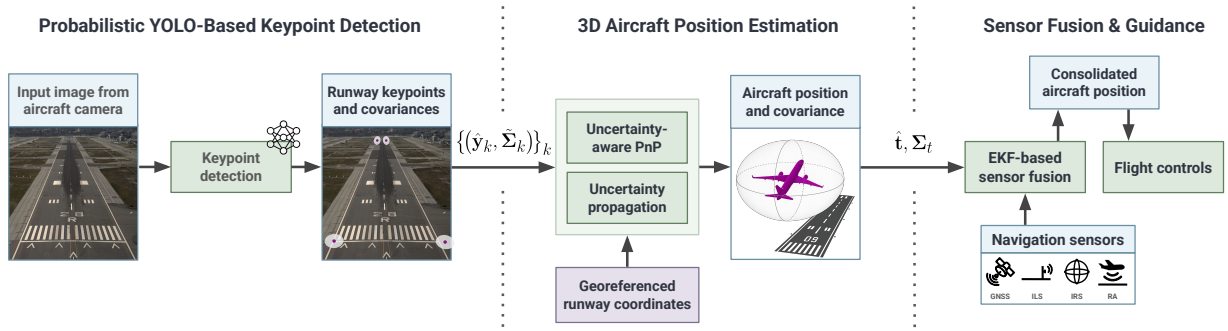


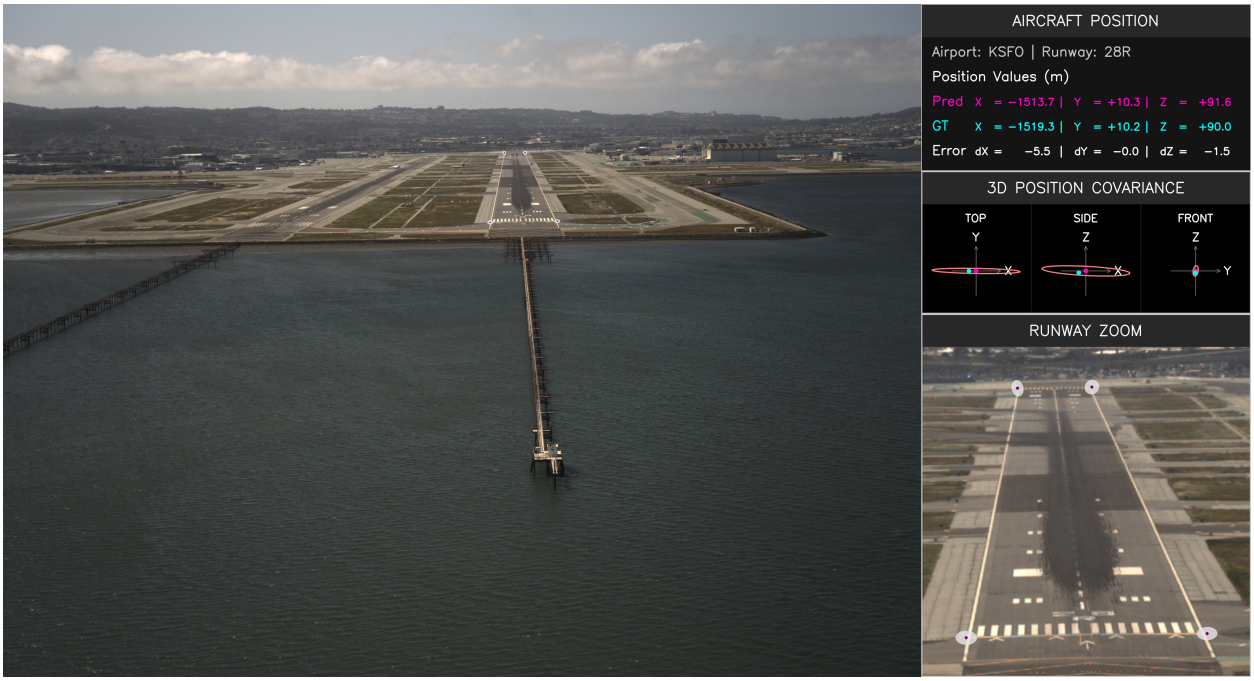
Figure 9: Vision-based aircraft landing. A YOLO-Pose model, augmented with our post-hoc probabilistic extension, predicts runway keypoints together with calibrated 2D covariances. These predictions are used in a covariance-weighted PnP step to estimate the 3D position of the aircraft, and the corresponding 3D covariance is obtained by propagating the 2D covariances through the PnP mapping. The resulting vision-based measurement is then integrated with other navigation sensors in a downstream EKF-based sensor-fusion module to support autonomous landing.

Experimental Setup. We instantiate our approach on SuperGradients' *YOLO-NAS-Pose*, where we adopt the Nano variant to meet the stringent latency requirements of VBL in real-time operation. The YOLO-NAS-Pose Nano is trained on a large-scale internal aircraft-landing dataset comprising over 10^6 data points. We use a standard train/validation/test split and reserve a separate calibration set for uncertainty calibration. The post-hoc probabilistic extension is trained without using data augmentations. This design choice ensures that the learned residual distribution closely reflects the distribution encountered at inference time. Residuals are obtained via SuperGradients' label assignment procedure, and training is performed using the proposed importance-weighted GNLL objective in Eq. (10). In line with the application setting described above, we restrict attention to the Gaussian model, since the downstream pipeline is designed around Gaussian uncertainty representations. At inference time, we use the SuperGradients' default post-processing and optionally apply keypoint pruning as described in Sec. 3.3.

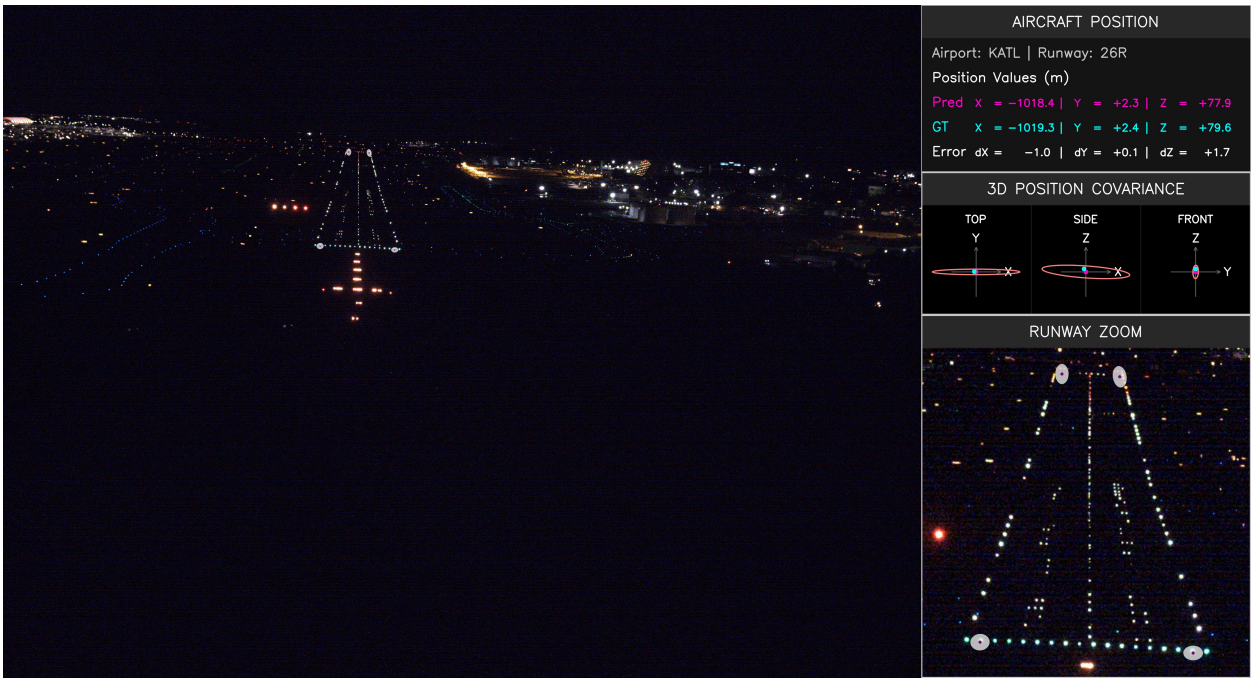
Results. We use VBL to assess whether improved keypoint uncertainty estimation transfers to a realistic downstream pipeline. Fig. 11 shows that Gaussian calibration provides a strong fit on the VBL data, with joint coverage close to the diagonal and good agreement between predicted and empirical variance. Compared with the COCO results in Fig. 4, the Gaussian model is substantially better calibrated, which is plausible because runway keypoints follow a more constrained projective geometry with limited pose, appearance, visibility, and occlusion variation. Pruning uncertain keypoints further improves coverage calibration by excluding predictions that tend to exhibit large residuals, reducing the risk of poorly localized keypoints affecting the position estimate. The scale-wise ACE results further show that the proposed importance-weighted GNLL yields balanced calibration across object areas, a key requirement for VBL given the wide range of runway scales encountered during approach. Further examples for San Francisco International Airport (KSFO), Los Angeles International Airport (KLAX), and Hartsfield–Jackson Atlanta International Airport (KATL) are provided in App. F.5, Figs. 22 to 24.

Propagating the calibrated image-plane covariances through PnP yields informative 3D position covariances, but we observed that their calibration is less accurate than the image-plane calibration. This is expected because 3D covariance calibration also depends on the uncertainty-propagation approximation and on unmodeled uncertainty in camera intrinsics and extrinsics. These downstream effects are outside the scope of this work, which focuses on calibrated image-plane covariances for keypoint detection.

Finally, Fig. 25 in App. F.5 shows vision-based landing in foggy conditions at Charlotte Douglas International Airport (KCLT), representing a distribution shift relative to the training data. The VBL model fails in several frames, and the predicted covariances do not consistently capture this failure. This illustrates that calibrated keypoint covariances learned in the in-distribution regime do not account for epistemic uncertainty associated with out-of-distribution inputs. Robust handling of such cases requires additional mechanisms for detecting

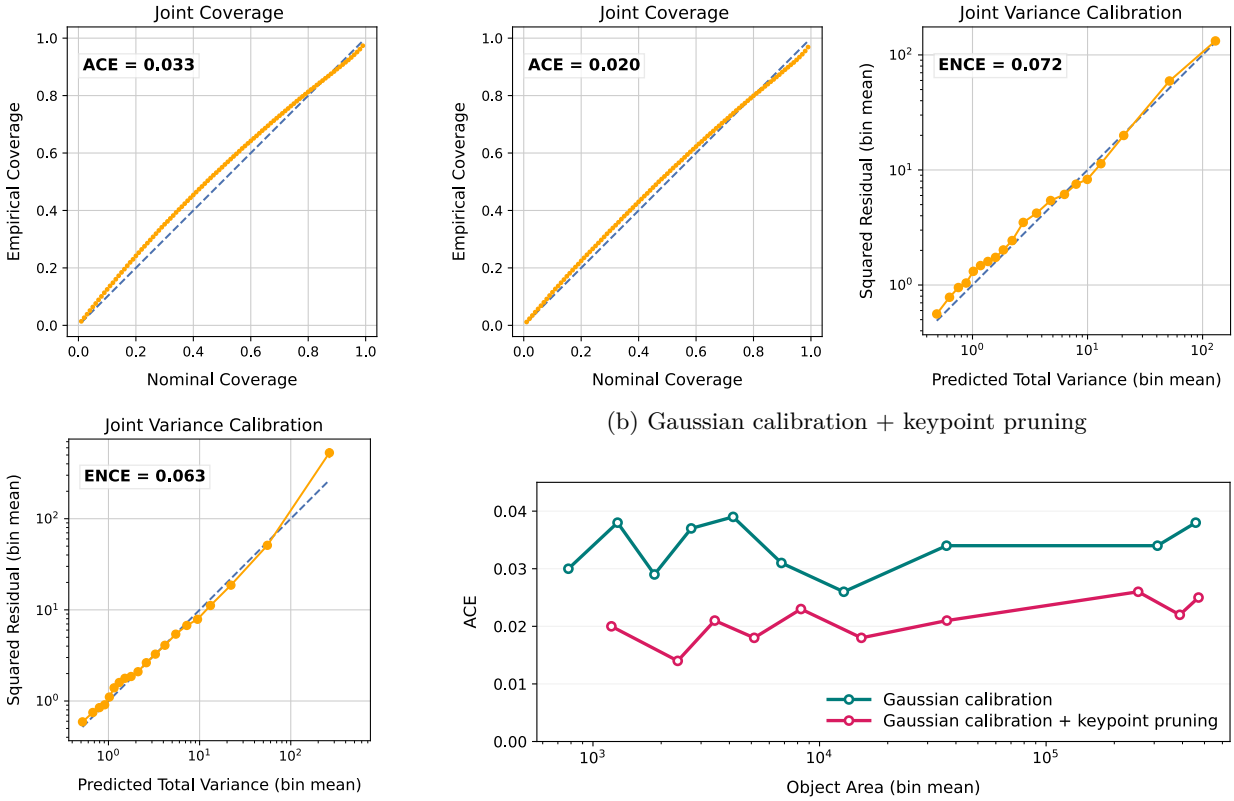


(a) San Francisco International Airport (KSFO), see App. F.5, Fig. 22 for additional KSFO examples.



(b) Hartsfield–Jackson Atlanta International Airport (KATL), see App. F.5, Fig. 24 for additional KATL examples.

Figure 10: Vision-based aircraft landing examples. Detected runway keypoints define the mean locations of the calibrated predictive distributions, visualized as 95% probability ellipses. The 2D–3D matches between image keypoints and georeferenced runway corners are used in a covariance-weighted PnP formulation to estimate the aircraft position relative to the runway. The propagated 3D position uncertainty is represented by a 95% probability ellipsoid, shown through top, side, and front projections. Positions are reported in the runway coordinate system, where x, y, and z denote longitudinal, lateral, and vertical directions, respectively. *GT* denotes the differential-GPS reference position, *Pred* the VBL prediction, and *Diff* their difference.



(b) Gaussian calibration + keypoint pruning

(a) Gaussian calibration

(c) Balanced calibration across object scales

Figure 11: Calibration diagnostics for vision-based aircraft landing, comparing the Gaussian-calibrated model before and after pruning uncertain keypoints. The scale-wise ACE plot shows that our importance-weighted GNLL yields balanced calibration across object areas, a key requirement for VBL. Compared with COCO (cf. Fig. 4), the Gaussian model is substantially better calibrated on the VBL data, reflecting the more constrained landing geometry, limited pose and appearance variation, and fewer visibility or occlusion ambiguities of runway keypoints. Additional marginal calibration diagnostics and example predictions before and after keypoint pruning are provided in App. F.5, Figs. 26 to 28.

or accounting for distribution shift, such as epistemic-uncertainty modeling that exposes shift-induced model uncertainty and supports out-of-distribution detection. This remains an important direction for future work, especially in the context of safety-critical applications.

7 Conclusion

We presented a lightweight post-hoc probabilistic extension for YOLO-Pose models that augments a trained YOLO-Pose model with calibrated predictive distributions over keypoint localization errors while preserving its original predictions. This design is motivated by the task-specific optimization of modern YOLO models through architectures, assignment strategies, losses, and post-processing pipelines. Rather than replacing these components with a new training objective, which may compromise prediction quality or require revalidating already trained or certification-constrained models, our approach learns heteroscedastic dispersion around the fixed keypoint predictions and calibrates the resulting predictive distribution in a subsequent step. Based on this predictive distribution, we further introduced uncertainty-based keypoint pruning as a post-hoc mechanism for removing high-uncertainty keypoint predictions.

Complementing the probabilistic extension, we proposed an uncertainty evaluation protocol for assessing both keypoint reliability ranking and distributional calibration. For reliability ranking, we introduced AKP, a

keypoint-level extension of the COCO AP protocol that evaluates whether confidence scores rank individual keypoints according to localization correctness. For distributional calibration, we evaluate the predicted bivariate keypoint distributions using joint and marginal coverage, Q–Q, and variance-calibration diagnostics. Together, the protocol assesses whether uncertainty estimates are calibrated and useful for selecting, pruning, or weighting keypoints in downstream tasks.

Our benchmark experiments on COCO show that the separation between point prediction, dispersion learning, and calibration is both practically useful and empirically important. The same trained dispersion model supports Gaussian and Student-t calibration regimes through stored calibration parameter sets: Gaussian predictions provide covariance compatibility for downstream pipelines, while Student-t predictions better preserve the heavy-tailed residual structure observed for OKS-trained keypoint detectors. Gaussian calibration improves variance calibration and NLL, but remains limited by likelihood mismatch; Student-t calibration better captures heavy-tailed residuals and provides the strongest overall calibration performance. Uncertainty-based keypoint pruning removes high-uncertainty, empirically unreliable keypoints while leaving retained predictions unchanged, improving calibration of the retained predictions. The ablation studies confirm the importance of the proposed weighting scheme: retention weighting has the larger effect on overall calibration, while object-scale weighting yields more balanced calibration across object scales.

Finally, the experiments on vision-based aircraft landing demonstrate the downstream relevance of calibrated keypoint uncertainty estimates. The Gaussian-calibrated model provides well-calibrated image-plane covariances that can be propagated through uncertainty-aware PnP and used as measurement uncertainty in downstream EKF-based sensor fusion. At the same time, the foggy landing sequence highlights a fundamental limitation of the present approach: uncertainty estimates learned and calibrated from in-distribution residuals do not, by themselves, capture epistemic uncertainty under distribution shift. Future work should prioritize epistemic uncertainty estimation and out-of-distribution detection while also exploring more expressive calibration methods.

References

Shay Aharon, Louis-Dupont, Ofri Masad, Kate Yurkova, Lotem Fridman, Lkdci, Eugene Khvedchenya, Ran Rubin, Natan Bagrov, Borys Tymchenko, Tomer Keren, Alexander Zhilko, and Eran-Deci. Super-gradients, 2023. URL https://zenodo.org/record/7789328.

Giovanni Balduzzi, Martino Ferrari Bravo, Anna Chernova, Calin Cruceru, Luuk van Dijk, Peter de Lange, Juan Jerez, Nathanaël Koehler, Mathias Koerner, Corentin Perret-Gentil, Zoltan Pillio, Ruben Polak, Hugo Silva, Romeo Valentin, Ian Whittington, and Grigory Yakushev. Neural network based runway landing guidance for general aviation autoland. Technical Report DOT/FAA/TC-21/48, Federal Aviation Administration, William J. Hughes Technical Center, 2021.

Jonathan T. Barron. A general and adaptive robust loss function. In *CVPR*, pp. 4331–4339, 2019. doi: 10.1109/CVPR.2019.00446.

John A. Caldwell, Melissa M. Mallis, J. Lynn Caldwell, Michel A. Paul, James C. Miller, and David F. Neri. Fatigue Countermeasures in Aviation. *Aviation, Space, and Environmental Medicine*, 80(1):29–59, 2009. doi: 10.3357/ASEM.2435.2009.

Jiwoong Choi, Dayoung Chun, Hyun Kim, and Hyuk-Jae Lee. Gaussian YOLOv3: An accurate and fast object detector using localization uncertainty for autonomous driving. In *ICCV*, pp. 502–511, 2019. doi: 10.1109/ICCV.2019.00059.

European Union Aviation Safety Agency. EASA artificial intelligence (AI) concept paper issue 2: Guidance for level 1 & 2 machine-learning applications. Concept paper, European Union Aviation Safety Agency (EASA), March 2024a. Issue 02. Published 6 Mar 2024 (document dated March 2024). Accessed 25 Feb 2026.

European Union Aviation Safety Agency. EASA research — machine learning application approval (MLEAP): Final report (MLEAP-D4 public report, issue 01). Research Project Final Report EASA.2021.C38 (MLEAP-D4), European Union Aviation Safety Agency (EASA), May 2024b. Dated 28 May 2024. Accessed 25 Feb 2026.

European Union Aviation Safety Agency. NPA 2025-07: Detailed specifications and associated acceptable means of compliance and guidance material — artificial intelligence trustworthiness. Notice of Proposed Amendment (NPA) 2025-07, European Union Aviation Safety Agency (EASA), November 2025. Issued 10 Nov 2025. Includes Part (A) Explanatory Note and Part (B) Proposed DS.AI with AMC & GM. Accessed 25 Feb 2026.

Federal Aviation Administration. Roadmap for artificial intelligence safety assurance (version i). Roadmap, Federal Aviation Administration (FAA), July 2024. Published 23 Jul 2024. Accessed 25 Feb 2026.

Chengjian Feng, Yujie Zhong, Yu Gao, Matthew R. Scott, and Weilin Huang. TOOD: Task-aligned one-stage object detection. In *ICCV*, pp. 3510–3519, 2021.

Luis Ferraz, Xavier Binefa, and Francesc Moreno-Noguer. Leveraging feature uncertainty in the PnP problem. In *BMVC*. BMVA Press, 2014. doi: 10.5244/C.28.83.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *ICML*, pp. 1050–1059. PMLR, 2016.

Johannes Gilg, Torben Teepe, Fabian Herzog, and Gerhard Rigoll. The box size confidence bias harms your object detector. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1471–1480, 2023.

Tilmann Gneiting. Making and evaluating point forecasts. *Journal of the American Statistical Association*, 106(494):746–762, 2011. doi: 10.1198/jasa.2011.r10138.

Tilmann Gneiting, Fadoua Balabdaoui, and Adrian E. Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B*, 69(2):243–268, 2007.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *ICML*, pp. 1321–1330. PMLR, 2017.

Yihui He, Chenchen Zhu, Jianren Wang, Marios Savvides, and Xiangyu Zhang. Bounding box regression with uncertainty for accurate object detection. In *CVPR*, pp. 2888–2897, 2019.

Giles Hooker and James O. Ramsay. Learned-loss boosting. *Computational Statistics & Data Analysis*, 56 (12):3935–3944, 2012. doi: 10.1016/j.csda.2012.05.019.

Peter J. Huber and Elvezio M. Ronchetti. *Robust Statistics*. John Wiley & Sons, New York, second edition, 2009. doi: 10.1002/9780470434697.

International Civil Aviation Organization. European Guidance Material on Management of ILS Critical and Sensitive Areas. EUR Doc 040, International Civil Aviation Organization, European and North Atlantic Office, Paris, France, June 2020.

Sebastián Jiménez, Mira Jürgens, and Willem Waegeman. Position: Epistemic uncertainty estimation methods are fundamentally incomplete, 2025.

Glenn Jocher, Ayush Chaurasia, Jing Qiu, et al. Ultralytics YOLO, 2025. URL `https://zenodo.org/records/18147344`.

Alex Kendall and Yarin Gal. What uncertainties do we need in Bayesian deep learning for computer vision? In *NeurIPS*, volume 30, 2017.

Samuel Kotz and Saralees Nadarajah. *Multivariate t-Distributions and Their Applications*. Cambridge University Press, 2004.

Florian Kraus and Klaus Dietmayer. Uncertainty estimation in one-stage object detection. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 53–60. IEEE, 2019. doi: 10.1109/ITSC.2019.8917494.

Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2796–2804. PMLR, 2018.

Fabian Küppers, Jan Kronenberger, Amirhossein Shantia, and Anselm Haselhoff. Multivariate confidence calibration for object detection. In *CVPRW*, pp. 1322–1330. IEEE, June 2020. doi: 10.1109/CVPRW50498.2020.00171.

Dan Levi, Liran Gispan, Niv Giladi, and Ethan Fetaya. Evaluating and calibrating uncertainty prediction in regression tasks. *Sensors*, 22(15):5540, 2022. doi: 10.3390/s22155540.

Jiefeng Li, Siyuan Bian, Ailing Zeng, Can Wang, Bo Pang, Wentao Liu, and Cewu Lu. Human pose regression with residual log-likelihood estimation. In *ICCV*, pp. 11025–11034, 2021.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common objects in context. In *ECCV*, pp. 740–755, 2014. doi: 10.1007/978-3-319-10602-1_48.

Debapriya Maji, Soyeb Nagori, Manu Mathew, and Deepak Poddar. YOLO-Pose: Enhancing YOLO for multi person pose estimation using object keypoint similarity loss. In *CVPRW*, pp. 2637–2646, 2022. doi: 10.1109/CVPRW56347.2022.00297.

Ruben Morales-Ferre, Philipp Richter, Emanuela Falletti, Alberto de la Fuente, and Elena Simona Lohan. A Survey on Coping With Intentional Interference in Satellite Navigation for Manned and Unmanned Aircraft. *IEEE Communications Surveys & Tutorials*, 22(1):249–291, 2020. doi: 10.1109/COMST.2019.2949178.

Tim Murphy, Matt Harris, Young Shin Park, and Sam Pullen. GBAS Differentially Corrected Positioning Service Ionospheric Anomaly Errors Evaluated in an Operational Context. In *Proceedings of the 2010 International Technical Meeting of The Institute of Navigation*, pp. 394–410, San Diego, CA, USA, January 2010. Institute of Navigation.

Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

Stergios I. Roumeliotis and George A. Bekey. An extended Kalman filter for frequent local and infrequent global sensor data fusion. In *Proceedings of SPIE: Sensor Fusion and Decentralized Control in Robotic Systems*, 2000.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *ICLR*, 2020.

Ranjan Sapkota, Rahul Harsha Cheppally, Ajay Sharda, and Manoj Karkee. YOLO26: Key architectural enhancements and performance benchmarking for real-time object detection, 2025.

Maximilian Seitzer, Arash Tavakoli, Dimitrije Antic, and Georg Martius. On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks. In *ICLR*, 2022.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000. ISSN 0378-3758. doi: 10.1016/S0378-3758(00)00115-4.

Hao Song, Tom Diethe, Meelis Kull, and Peter Flach. Distribution calibration for regression. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5897–5906. PMLR, 2019.

Andrew Stirn, Harm Wessels, Megan Schertzer, Laura Pereira, Neville Sanjana, and David Knowles. Faithful heteroscedastic regression with neural networks. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 5593–5613. PMLR, 2023.

Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *JMLR*, 8(35):985–1005, 2007.

Alexander Vakhitov, Luis Ferraz, Antonio Agudo, and Francesc Moreno-Noguer. Uncertainty-aware camera pose estimation from points and lines. In *CVPR*, pp. 4659–4668, 2021.

Romeo Valentin, Sydney M. Katz, Joonghyun Lee, Mansur Arief, Matthew Sorgenfrei, Don Walker, and Mykel J. Kochenderfer. Probabilistic parameter estimators and calibration metrics for pose estimation from image features. In *2024 AIAA DATC/IEEE 43rd Digital Avionics Systems Conference (DASC)*, 2024. doi: 10.1109/DASC62030.2024.10748707.

Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, and Guiguang Ding. YOLOv10: Real-time end-to-end object detection. In *NeurIPS*, volume 37, pp. 107984–108011, 2024. doi: 10.52202/079017-3429.

Samuel S. Wilk and Ramakrishna Gnanadesikan. Probability plotting methods for the analysis of data. *Biometrika*, 55(1):1–17, 1968.