Two-Stage World-Space Pose Refinement for Precise Soccer Player Localization

Parthsarthi Rawat
GameChanger by Dick's Sporting Goods
sarthi.rawat@gc.com

Abstract

We present a two-stage detection-and-refinement pipeline for sub-metre soccer player localization in world coordinates from broadcast 4K images. The first stage employs a YOLO26x pose model operating at 1920 px resolution on full 4K frames to produce player bounding boxes and coarse ground-projected keypoint estimates. The second stage extracts a padded crop around each detection and applies a second YOLO26x pose model at 640 px crop resolution to regress the ground-projected keypoint with sub-pixel precision. To bridge pixel-space training and metric-space evaluation, we derive a differentiable coordinate transform—reversing letterbox scaling, crop offsets, and perspective camera projection—and introduce a multi-scale LoCSim loss *that jointly penalises world-space error at $\tau \in \{0.25, 0.50, 1.0\}$ m. The loss is injected only into the one-to-many detection branch, leaving the one-to-one inference head unaffected while directing gradients toward small, hard-to-localise players. On the SpiideoSynLoc challenge set our method achieves **94.05%** mAP-LocSim at τ=1 m and **98.90%** at τ=5 m.*

1. Method

1.1. Stage 1: Full-Resolution Player Detection and Coarse Pose Estimation

Stage 1 processes the raw 4K image (3840×2160) using a YOLO26x pose model [1] trained at `imgsz`=1920. Each detected player is represented by a bounding box, a confidence score, and two coarse keypoints: the pelvis point (kpt 0) and its orthogonal projection onto the ground plane (kpt 1). These coarse kpt 1 predictions serve as fallbacks when Stage 2 fails to fire.

1.2. Stage 2: Crop-Level Keypoint Refinement

For each Stage-1 detection we extract a crop around the player bounding box with α=0.40 fractional padding:

$$\text{pad} = \alpha \cdot \max(b_w, b_h), \tag{1}$$

where b_w, b_h are the bounding box width and height. An additional 64-pixel border is added to the padded image before cropping to prevent out-of-bounds accesses near image edges. The resulting crop is fed to a YOLO26x pose model with `kpt_shape`=$[1, 3]$, trained at `imgsz`=640 to regress a single ground-projected keypoint (kpt 1). The crop coordinate of the predicted keypoint $\hat{p}_{\text{crop}}$ is then mapped back to 4K space:

$$\hat{p}_{4K} = \hat{p}_{\text{crop}} + (c_{x1},\, c_{y1}), \tag{2}$$

where (c_{x1}, c_{y1}) is the crop's top-left corner in the (edge-padded) 4K image. Both stages are trained on the SpiideoSynLoc training set. Stage 1 is trained first with the standard YOLO pose loss; the world-space LoCSim loss is added after a warm-up period to sharpen the coarse kpt 1 estimates before Stage 2 training begins. Stage 2 is first trained with the standard pose loss to obtain a strong keypoint baseline, then further fine-tuned with the world-space loss described below.

1.3. World-Space LoCSim Loss

Coordinate transform. Training images for Stage 2 are letterboxed to 640×640. Given crop dimensions (W_c, H_c), the letterbox scale and padding are

$$s_{\text{lb}} = \min\!\left(\tfrac{640}{W_c},\, \tfrac{640}{H_c}\right), \quad p_w = \tfrac{640 - W_c s_{\text{lb}}}{2},\; p_h = \tfrac{640 - H_c s_{\text{lb}}}{2}, \tag{3}$$

where s_{lb} is the letterbox scale factor and p_w, p_h are the horizontal and vertical pixel padding added during letterboxing. A predicted pixel coordinate $\hat{x}_{640}$ is back-projected to 4K via

$$\hat{x}_c = \tfrac{\hat{x}_{640} - p_w}{s_{\text{lb}}}, \qquad \hat{x}_{4K} = \hat{x}_c + c_{x1}, \tag{4}$$

and then normalised for the camera projection function [2]:

$$\tilde{x} = \frac{\hat{x}_{4K} - (W_{4K} - 1)/2}{W_{4K}}, \quad \tilde{y} = \frac{\hat{y}_{4K} - (H_{4K} - 1)/2}{W_{4K}}, \tag{5}$$

where W_{4K}=3840 and H_{4K}=2160 are the 4K frame dimensions. The full transform $\pi : (\tilde{x}, \tilde{y}) \mapsto (X, Y)_{\text{world}}$ uses the

per-image camera matrix and radial distortion polynomial provided with the dataset [2], and is implemented as a differentiable PyTorch operation so that gradients flow back to $\hat{p}_{640}$.

Multi-scale LoCSim loss. The LoCSim score at threshold τ for world-space error d m is

$$\mathrm{LS}(d;\tau) = \exp\big(\ln(0.05)\cdot d^2/\tau^2\big). \quad (6)$$

Rather than optimising at a single τ, we use a weighted multi-scale combination

$$\mathcal{L}_{\text{world}} = 1 - \big(0.2\,\mathrm{LS}_{1.0} + 0.3\,\mathrm{LS}_{0.5} + 0.5\,\mathrm{LS}_{0.25}\big), \quad (7)$$

which provides a strong gradient for moderate errors ($\sim$0.5 m) via τ=0.25 while still rewarding coarse improvements via τ=1.0.

Adaptive sample weighting. Two complementary weights are applied per sample. First, an *error-margin* weight

$$w_e = \sigma\big((d-\delta_{\text{lo}})/T\big), \quad \delta_{\text{lo}} = 0.13\,\text{m},\ T = 0.03\,\text{m}, \quad (8)$$

where σ is the sigmoid function, suppresses gradients for already-accurate predictions ($d < 0.13$ m) and focuses training on the failure modes. Second, an *area* weight

$$w_a = \min\left(\sqrt{2000/A_{\text{orig}}},\ 4.0\right) \quad (9)$$

upweights small players, where A_{orig} is the estimated player bounding-box area in 4K pixels (inferred from the crop size as $A_{\text{orig}} \approx (W_c/1.8)(H_c/1.8)$). Players with $A_{\text{orig}} \geq 2500\,\text{px}^2$ are excluded from the world loss entirely; they are already well-localised, and applying noisy world-space gradients empirically degrades their accuracy.

Training protocol. YOLO26 employs dual detection heads: a one-to-many branch used during training and a one-to-one branch used at inference [1]. The world-space loss is applied exclusively through the one-to-many branch, ensuring the one-to-one inference head remains unaffected. Each stage is trained for up to 50 epochs with early stopping. The Stage 1 LoCSim warm-up uses Ultralytics default training hyperparameters. The world-space fine-tuning stage uses $\mathrm{lr}_0 = 2\times10^{-4}$ with cosine decay and all geometric augmentations disabled (crop offsets must be exact for the coordinate transform in Eq. (4)–(5)); photometric augmentations (HSV jitter) are retained.

2. Experiments

Dataset. We train and evaluate on SpiideoSynLoc [2], a synthetic soccer dataset with full camera calibration per frame. The training split contains 65 k frames with approximately 668 k annotated player instances; results are reported on the validation, test, and challenge splits.

Evaluation metric. mAP-LocSim is COCO-style mean average precision where the per-detection IoU is replaced by the LoCSim score (Eq. (6)) at τ=1 m, thresholded at ten score levels from 0.50 to 0.95 (in steps of 0.05) and integrated over recall [2].

Results. Table 1 reports the full set of competition metrics across all three evaluation splits. The proposed method achieves **94.34%** mAP-LocSim at τ=1 m on the validation set, **93.42%** on the test set, and **94.05%** on the challenge set, demonstrating consistent performance across splits. At the looser threshold τ=5 m the system achieves at least **98.85%** mAP-LocSim on all three splits.

Table 1. Results on the SpiideoSynLoc validation, test, and challenge sets.

	Val		**Test**		**Challenge**	
Metric	τ_1	τ_5	τ_1	τ_5	τ_1	τ_5
mAP-LocSim	**94.34**	**98.94**	**93.42**	**98.85**	**94.05**	**98.90**
Precision	98.98	99.07	98.29	98.47	98.67	98.82
Recall	97.00	98.00	97.00	98.00	97.00	98.00
F1	97.98	98.53	97.64	98.24	97.83	98.41
Frame Acc.	75.39	79.12	75.97	80.70	75.92	80.49

Size analysis. Evaluating on the validation set by player bounding-box area reveals that large players ($>8\,\text{k px}^2$) achieve 98.3% at LoCSim ≥ 0.95 (τ=1 m), while small players (500–2 k px^2, the most populous group at $\sim$45% of instances) reach 86.2% at the same threshold. The world-space loss and area weighting (Eq. (9)) provide the largest gains in this small-player bucket, which dominates the overall mAP.

References

[1] Ranjan Sapkota, Rahul Harsha Cheppally, Ajay Sharda, and Manoj Karkee. YOLO26: Key architectural enhancements and performance benchmarking for real-time object detection, 2026. URL https://arxiv.org/abs/2509.25164. 1, 2

[2] Håkan Ardö, Mikael Nilsson, Anthony Cioppa, Floriane Magera, Silvio Giancola, Haochen Liu, Bernard Ghanem, and Marc Van Droogenbroeck. Spiideo SoccerNet SynLoc — single frame world coordinate athlete detection and localization with synthetic data. In *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, pages 278–285. SciTePress, 2025. ISBN 978-989-758-728-3. doi: 10.5220/0013108200003912. 1, 2