From Broadcast to Minimap: Achieving State-of-the-Art SoccerNet Game State Reconstruction

Supplementary Material

A. Camera Parameters Network Architecture

See Figure 1 for a detailed diagram of the camera parameters prediction model architecture.

B. Pitch Localization

Accurate pitch localization is essential for mapping athletes on the image to their 3D positions on the pitch. Our method employs a multi-stage approach that first leverages a custom SegFormer model to generate an initial estimate of the camera parameters. Following this, a ResNet50-based segmentation network detects keypoints on the field, such as intersections of pitch lines with grass lines. These keypoints are then used in an optimization process to refine the estimated parameters, ensuring a more precise camera alignment with the real-world pitch.

You can find visualization of the pitch localization pipeline in Figure 2.

C. Camera Parameters Dataset

The histograms in Figure 3 compare key distributions, such as camera position coordinates (X, Y, Z), field of view, pan, and tilt angles. The real dataset is naturally constrained by physical camera placements, while the synthetic dataset spans a wider range of configurations, enabling the model to learn robust representations. You can see examples of synthetic images in Figure 4.

D. Camera Parameters Loss

Given the camera parameters params $= \{I, R, t\}$, we define a mapping function $P = F(\text{params})$ that transforms 3D world coordinates X into 3D camera coordinates in Normalized Device Coordinates (NDC) space:

$$x_{\text{camera}} = P(X) = IR(X - t),$$

where:

- X is the 3D world coordinates $[X, Y, Z]^T$.
- I is the intrinsic matrix, encoding focal length and principal point (which is set to zero in the case of NDC coordinates).
- R is the rotation matrix representing the camera's orientation.
- t is the translation vector representing the camera's position.
- x_{camera} is the resulting 3D point $[x_c, y_c, z_c]^T$ in NDC space.

This 3D-to-3D transformation (from world coordinates to Normalized Device Coordinates, or NDC) offers three key advantages: (1) it ensures resolution invariance by decoupling the loss from the input image size, (2) it eliminates the need for a perspective divide, thereby maintaining a smooth and stable gradient flow during optimization, and (3) it retains invertibility, enabling consistent reconstruction of 3D world coordinates from camera coordinates.

The inverse mapping is derived as follows. Starting from the forward transformation:

$$x_{\text{camera}} = IR(X - t),$$

we derive the inverse as follows:

- Multiply both sides by $(IR)^{-1}$ to isolate $X - t$:

$$(IR)^{-1} x_{\text{camera}} = X - t.$$

- Solve for X by adding t to both sides:

$$X = (IR)^{-1} x_{\text{camera}} + t.$$

- Since R is an orthogonal matrix, $R^{-1} = R^T$, giving the final expression:

$$X = R^T I^{-1} x_{\text{camera}} + t.$$

Thus, our inverse mapping is:

$$P^{\text{inv}}(x_{\text{camera}}) = R^T I^{-1} x_{\text{camera}} + t.$$

This formulation plays a critical role in our training loss, allowing symmetric penalization of both forward and inverse transformations between world and NDC camera coordinates.

E. Camera parameters data preparation

Figure 5 illustrates the projection of keypoints and coordinate heatmaps into image space using a homography matrix computed from the camera parameters.

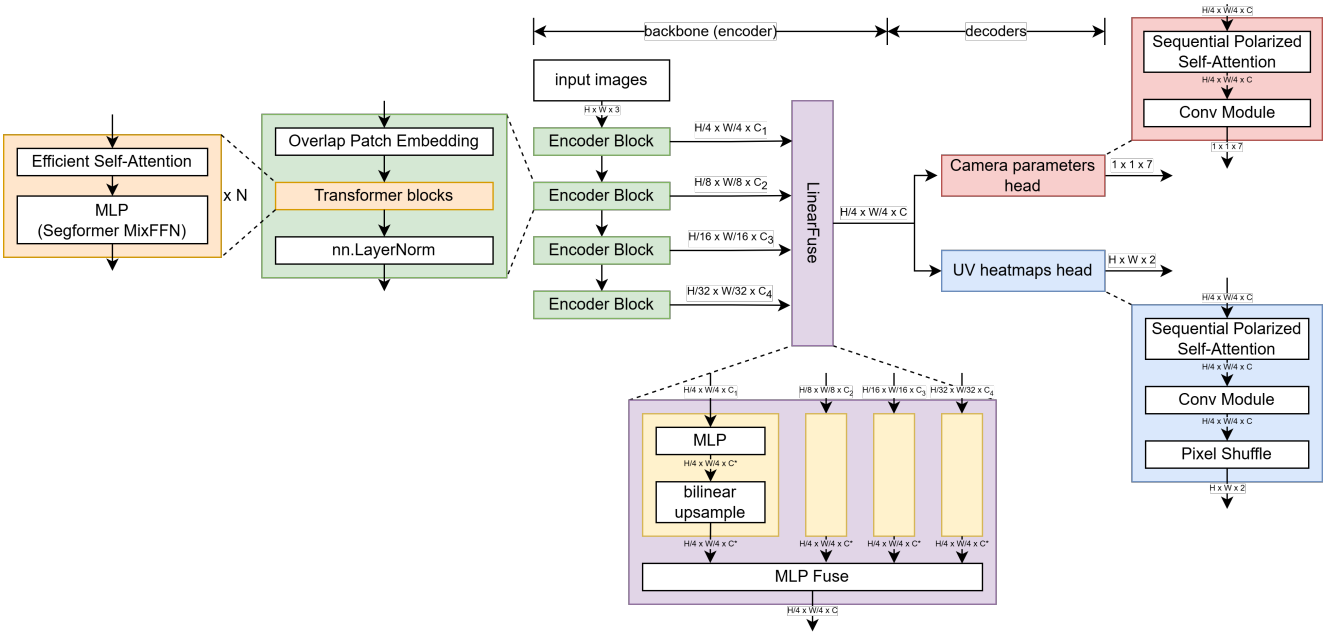


Figure 1. Camera Parameters Model. This figure illustrates the architecture of our custom SegFormer-based camera parameter estimator. The model consists of an encoder-decoder structure, where the encoder is based on the SegFormer architecture and the decoder includes two heads: one for predicting camera parameters (position, orientation, and field of view) and another for generating UV heatmaps.

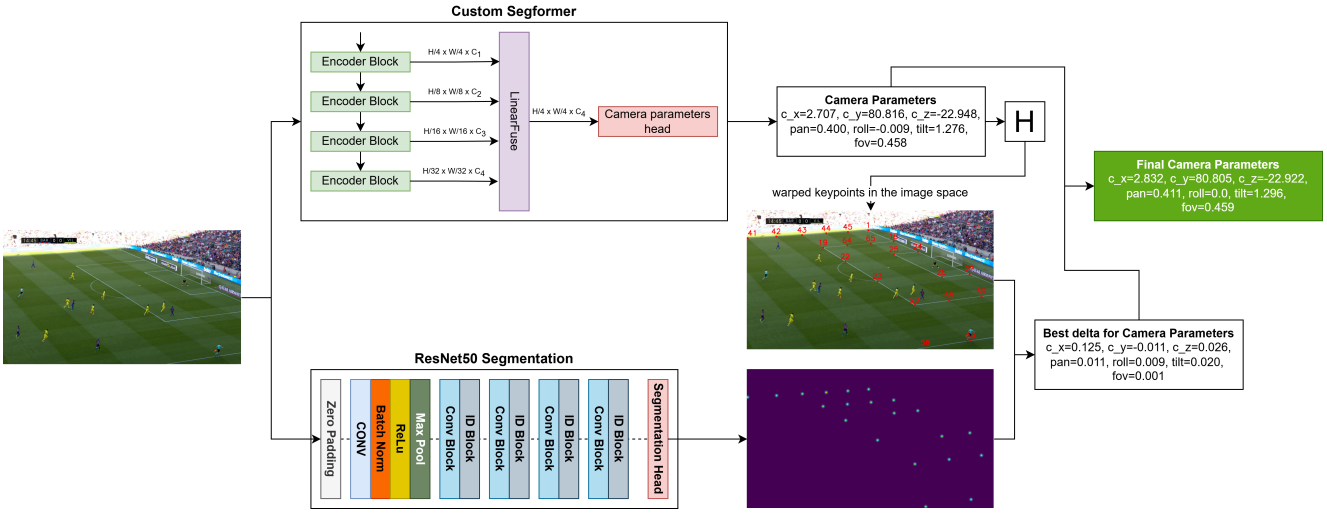


Figure 2. The pipeline estimates camera parameters by combining a custom SegFormer model for initial predictions and a ResNet50-based segmentation for keypoint detection. The parameters are refined using keypoint alignment to obtain the final camera pose.

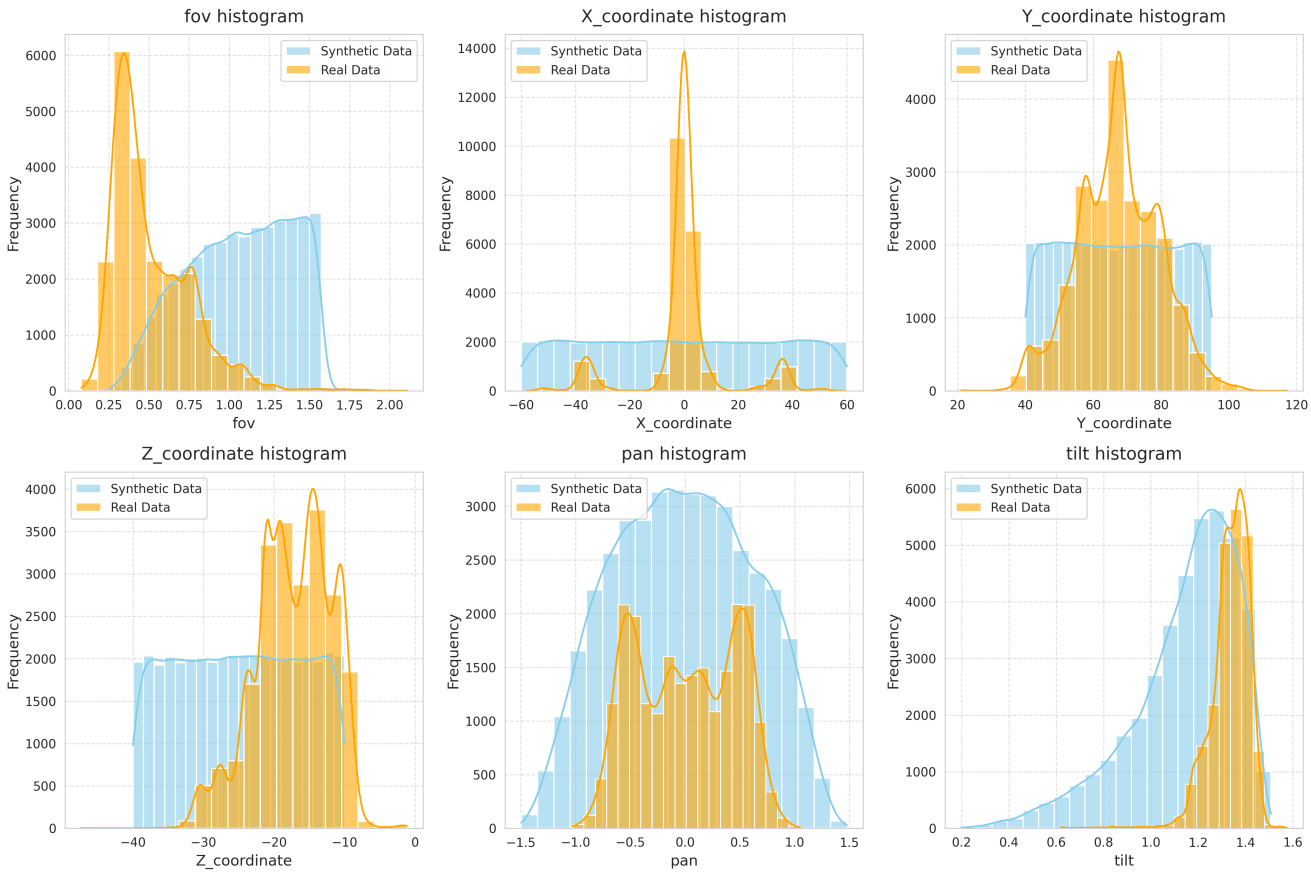
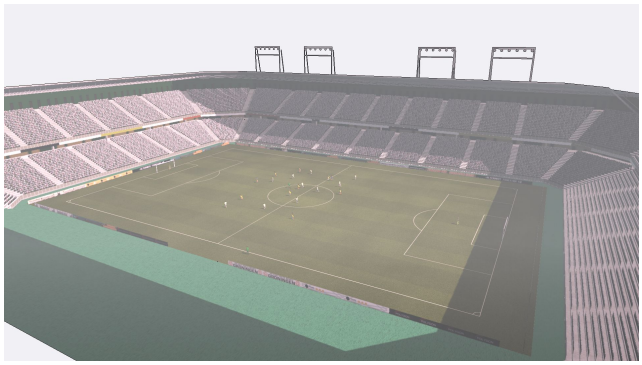


Figure 3. Real and synthetic data statistics. The histograms compare the distributions of key camera parameters and coordinate values between real and synthetic datasets. The X, Y, and Z coordinates represent camera positions with respect to the center of the football field, which serves as the origin (0,0,0). The FIFA standard field dimensions are 105 meters (length) × 68 meters (width). The field of view (FoV), pan, and tilt angles illustrate differences in camera configurations across datasets, while roll is fixed at 0 for all images. The synthetic data (blue) shows a broader and more uniform distribution, while the real data (orange) exhibits a more concentrated range of values, indicating the constrained nature of real-world camera placements.

(a) FoV: 0.86, c_x: -48.19, c_y: 72.27, c_z: -13.31, Pan: -0.06, Tilt: 1.35, Roll: 0.0

(b) FoV: 0.47, c_x: -11.75, c_y: 74.62, c_z: -34.18, Pan: -0.12, Tilt: 1.16, Roll: 0.0

(c) FoV: 0.78, c_x: 26.77, c_y: 44.59, c_z: -34.42, Pan: -0.71, Tilt: 1.23, Roll: 0.0



(d) FoV: 1.29, c_x: 57.28, c_y: 94.18, c_z: -39.87, Pan: -0.49, Tilt: 1.25, Roll: 0.0

Figure 4. Examples from the synthetic dataset with corresponding camera parameters.

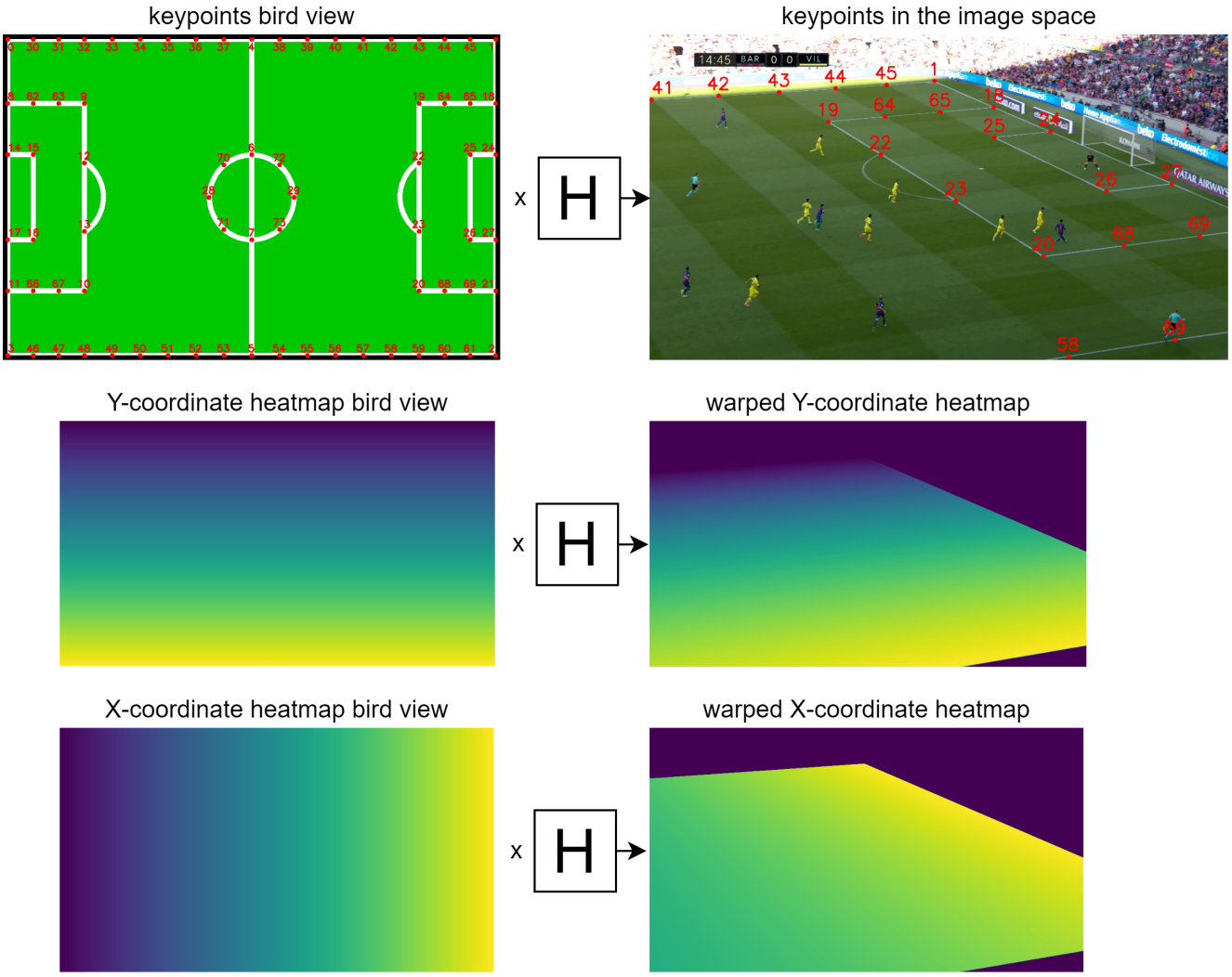


Figure 5. Keypoints, Y-coordinate heatmap (bird's-eye view), and X-coordinate heatmap are projected into image space using a homography matrix derived from the camera parameters.