(Supplementary) TranSPORTmer: A Holistic Approach to Trajectory Understanding in Multi-Agent Sports

Guillem Capellera[1,2], Luis Ferraz[1], Antonio Rubio[1], Antonio Agudo[2], and Francesc Moreno-Noguer[2]

[1] Kognia Sports Intelligence, Barcelona, Spain
{guillem.capellera,luis.ferraz,antonio.rubio}@kogniasports.com
[2] Institut de Robòtica i Informàtica Industrial CSIC-UPC, Barcelona, Spain
{gcapellera,aagudo,fmoreno}@iri.upc.edu

1 Soccer baselines

The task of forecasting (or imputing) player trajectories based on the future movement of the ball or the opposing team has been seldom explored by state-of-the-art methods. To our knowledge, there are few baseline methods that address this aspect in forecasting tasks [2,24]. Therefore, we implemented these pipelines to showcase the results for the soccer dataset presented in Table 1 of the main paper. The baselines used for comparison are described as follows:

Velocity: As a sanity check, we adopted this baseline, projecting agent predictions linearly based on observed velocity.

RNN: This baseline utilizes an encoder with LSTM, employing shared weights to capture input representations of each agent, along with an MLP decoder for prediction [4].

GRNN: This is a non-variational version of GVRNN [18, 24], generating trajectories without sampling. The training process involves using the ADE loss ($\mathcal{L}_{\text{ADE}}$). Yeh et al. [24] demonstrated superior results of GRNN over GVRNN in a soccer context. We use the implementation from Teranishi et al. [19].

GRNN + Att: Similar to the previous baseline but using a Graph Attention Network (GAT) instead of GNNs, inspired by [5, 10].

Transformer: Inspired by [2,3], this baseline uses the same pipeline as ours but incorporates attention through the flattened temporal and social dimensions. It employs a 2D positional encoder [20], making it a non-equivariant baseline.

2 Training procedure

The models were trained on an NVIDIA RTX A6000 GPU for 100 epochs, using a batch size of 64 samples. We employed the AdamW optimizer [12, 14] with a learning rate of 0.001 and an epsilon value of 1×10^{-4}. The learning rate was reduced by a factor of 0.5 every 20 epochs. To prevent gradient explosion, we applied gradient clipping with a threshold of 5, ensuring stable optimization. Model weights were initialized using the Xavier normal distribution [8]. All experiments have been trained separately.

The hyperparameters for our method and were set as follows: 128-dimensional embeddings (d), 16 heads in each Multi-Head-Attention (MHA) module, a hidden dimension of 512 for each Set Attention Block (SAB). In TranSPORTmer with state classification, the weight for the Cross-Entropy (CE) loss was set to $\lambda = 4$ to balance the magnitudes of the two losses. It is worth noting that without state classification, $\lambda = 0$.

3 Basketball experiments

3.1 Forecasting with conditioning

In Table 2 of the main paper, we conduct a comparative analysis between our method and forecasting-based models using the Basketball-VU dataset. The majority of these models are generative stochastic approaches primarily focused on imitative tasks. However, they are generally sub-optimal at handling a significant number of agent interactions. As a result, their ability to accurately forecast trajectories for offensive and defensive players jointly is hindered, especially when trying to model trajectories for both teams simultaneously. Consequently, these models often resort to separately modeling these trajectories. In contrast, our method can model offensive and defensive players' trajectories simultaneously using a single model.

Furthermore, our approach can generate trajectories based on the movement of the opponent players and/or the ball. This capability allows us to predict their future movements much more accurately. To illustrate this point, we would like to present an additional experiment similar to Table 1 in our main paper but focused on the basketball context. This experiment showcases the results of forecasting basketball trajectories for players or the offensive/defensive team when conditioned on the other team and/or the ball. The results of this experiment are presented in Table 1, demonstrating the efficacy of our model in encoding these interactions and providing more accurate results as the conditioning agents increase. It's important to note that defense predictions show significant improvement when conditioned on the offensive team alone, more so than when conditioned on the ball. This is due to the intrinsic nature of basketball, particularly in one-on-one defense situations.

Predict P	**Players**		**Offense**				**Defense**			
Condition	None	Ball	None	Ball	Defense	Ball+Defense	None	Ball	Offense	Ball+Offense
$\text{ADE}_P \downarrow$	7.75	7.05	9.19	8.47	4.29	3.96	6.31	5.64	3.67	3.14
$\text{FDE}_P \downarrow$	11.65	11.13	14.24	13.78	7.44	7.05	9.04	8.48	5.78	4.76

Table 1: Evaluation in Basketball-VU dataset in player trajectory forecasting with TranSPORTmer w/o CLS. Predictions are generated with a time horizon of 8s using a prior of 2s. All metrics are in feet.

3.2 Unified imputation and forecasting

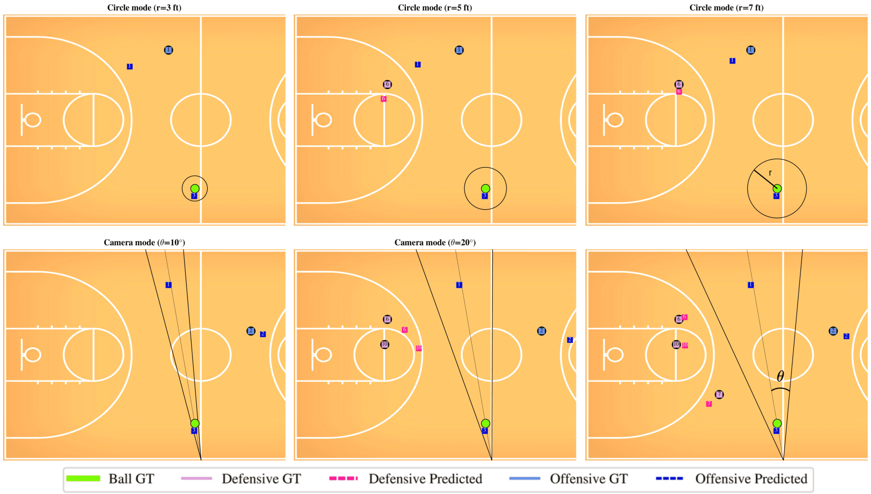

Fig. 1: Initial timestep example of Basketball-TIP dataset for "circle mode" and "camera mode". The depicted (predicted) players are those with at least one observation inside the circle/camera view during the imputation task.

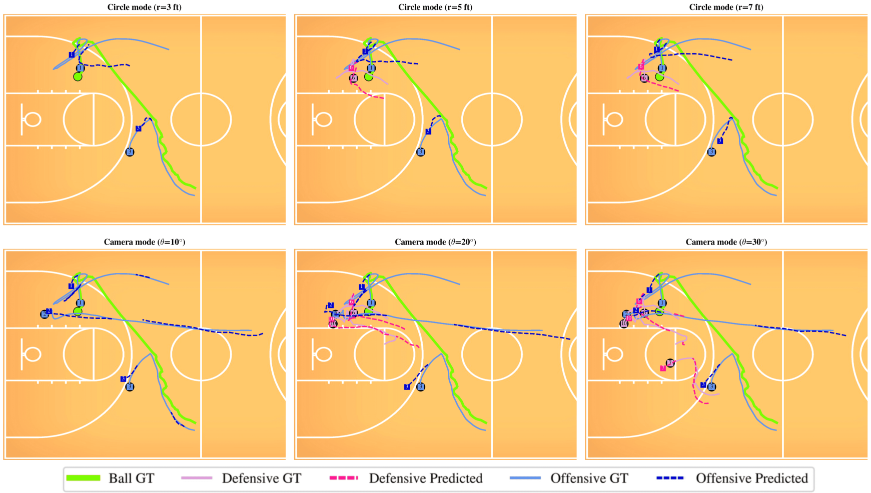

Fig. 2: Qualitative evaluation in Basketball-TIP dataset. Imputation during first 6.4s and forecasting during the subsequent 1.6s. Refer to the supplemental video to view the animated version.

In Table 3 of the main paper, we present a comparison against the state-of-the-art in unified imputation and forecasting tasks using the Basketball-TIP dataset. Here, we include two figures to provide visual support for the concepts of "circle mode" and "camera mode", showing one test sample solved for each strategy. Refer to Fig. 1 to view the initial time-step of the task, where the simulated view is depicted. Refer to Fig. 2 to view the same sequence solved until the final step with the predicted trajectories.

4 Training with missing data

In this section, we present two experiments conducted using the soccer dataset, focusing on handling missing data for goalkeepers. The first experiment involves generating the adapted dataset, enabling us to compare our approach against *ballradar* [23]. The second experiment assesses our model's capability to predict ball movements using the original dataset.

4.1 Adapted dataset: goalkeepers imputation and inference

We outline the methodology for goalkeeper trajectory imputation and inference, specifically tailored for creating the adapted dataset to facilitate a comparison between TranSPORTmer and *ballradar* [11] in ball trajectory tasks. The *ballradar* baseline relies on the positions of both goalkeepers for optimal performance in ball prediction tasks.

We initiate the process with the original soccer dataset discussed in Section 5.1 of the main paper. Since the observations are derived from optical tracking, several goalkeeper data points are missing. To address this, we initially filter and retain sequences containing at least one observation of a goalkeeper. Subsequently, we train TranSPORTmer in goalkeeper trajectory prediction using these adapted sequences. A random mask is applied, obscuring 97% to 100% of the goalkeepers' observations. The goalkeepers' unavailable observations are ignored by our model using the NaN-mask during training.

The evaluation metrics for goalkeepers' trajectory inference and imputation in the test samples are presented in Table 2. In the inference task, all available observations are concealed (100% mask), and trajectory imputation is performed with 97% of available observations hidden (97% mask). Figure 3 showcases two test samples solved with all ground truth observations hidden (100% mask). Using this trained model, we construct the adapted dataset by imputing the positions of missing observations for goalkeepers with at least one observation. In cases where one goalkeeper has no observations throughout the entire sequence, we manually set its position to a standard field position. Consequently, the new adapted dataset comprises 73,595 sequences for training, 6,628 for validation, and 5,725 for testing.

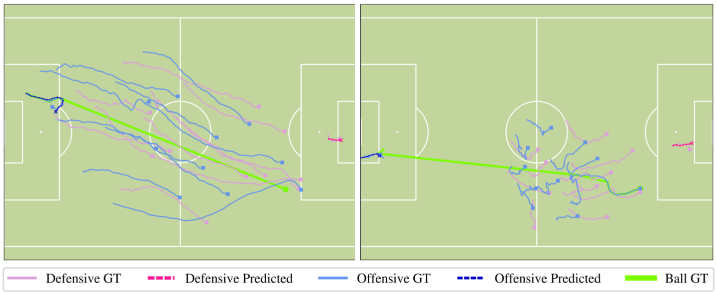

Fig. 3: Goalkeepers inference through the full 9.6s sequence. All available goalkeepers' observations are hidden.

Task	Mask	ADE ↓	MaxErr ↓
Inference	100%	1.97	3.32
Imputation	97%	0.82	1.96

Table 2: Evaluation of goalkeepers' imputation and inference on 9.6s sequences. The model utilized is TranSPORTmer w/o CLS. All metrics are in meters.

4.2 Ball inference

To ensure a fair comparison, in the main paper, we present the results of our method trained and evaluated using the adapted soccer dataset, which contains fewer sequences but includes inferred goalkeeper positions. Here, we aim to demonstrate the effectiveness of our model in predicting ball location using the original dataset, which lacks some goalkeeper observations but has approximately 10,000 (12.71%) more sequences for training. We present the results for ball inference in Table 3, using both the adapted dataset without missing data (adapted dataset w/o missing data) and the original dataset with missing data (original dataset w missing data). It's important to note that our model can be trained with incomplete data, allowing us to train on more sequences and leading to improved results.

	adapted dataset w/o missing data			**original dataset w missing data**	
	ballradar (KDD'23)	Ours w/o CLS	Ours	Ours w/o CLS	Ours
ADE ↓	3.89	2.89	2.71	2.73	2.57
MaxErr ↓	8.79	7.78	7.39	7.58	7.22
Acc (%) ↑	-	-	80.84	-	81.64

Table 3: Evaluation in ball inference in soccer. Predictions are generated through the full 9.6s sequence. All metrics, except Acc, are in meters.

5 Coarse-to-fine ablation

In this section, we perform an ablation regarding the number of the encoders of the proposed architecture. The considered task is the ball inference and we compare *Our w/o CLS*, which utilize two encoders that act as a coarse-to-fine manner, against utilizing one-single encoder and three encoders. A single encoder yelds ADE and MaxErr metrics of 4.40m and 9.83m, respectively, compared to our results of 2.71m and 7.39m (see Table 4 main paper). Figure 4 here shows attention maps for Seq1 and Seq2 using a single encoder, exhibiting noisier focus compared to our fine encoder (see "Ball Attn in second SAB_S" in Fig.4-right main paper), leading to suboptimal results. Using three encoders fails to converge.

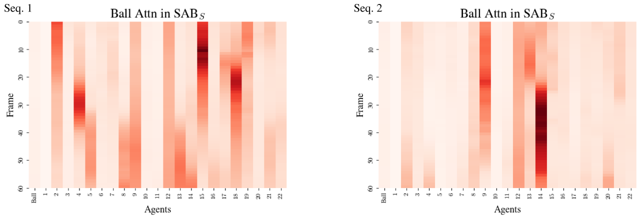

Fig. 4: Results compared to Fig.4-right of the main paper using only one encoder.

6 Pedestrian forecasting

Although analyzing urban pedestrian scenes is outside the main scope of this paper, for completeness we have included an experiment with the benchmark dataset ETH-UCY [13, 15]. This dataset comprises five different subsets: ETH, Hotel, Univ, Zara1, and Zara2. We follow the established convention of leave-one-out training [9] and employ the task of forecasting 12 future time-steps based on 8 preceding time-steps, with a frame rate of 2.5Hz. The results of our experiment against deterministic state-of-the-art models are presented in Table 4. It is worth pointing out that our approach is on a par with the most recent architectures, many of which are specifically tailored for pedestrian contexts, and achieves strong results in three subsets (ETH, Zara1, Zara2) and on Average. We achieve a 4.3% improvement in ADE on the ETH subset.

Model		ETH	Hotel	Univ	Zara1	Zara2	Average
S-LSTM [1]	CVPR'16	1.09/2.35	0.79/1.76	0.67/1.40	0.47/1.00	0.56/1.17	0.72/1.54
SGAN-ind [9]	CVPR'18	1.13/2.21	1.01/2.18	0.60/1.28	0.42/0.91	0.52/1.11	0.74/1.54
TransF [7]	ICPR'20	1.03/2.10	0.36/0.71	0.53/1.32	0.44/1.00	0.34/0.76	0.54/1.17
Trajectron++ [17]	ECCV'20	1.02/2.00	0.33/0.62	0.53/1.19	0.44/0.99	0.32/0.73	0.53/1.11
MemoNet [21]	CVPR'22	1.00/2.08	0.35/0.67	0.55/1.19	0.46/1.00	0.32/0.82	0.55/1.15
Autobots [6]	ICLR'22	1.02/1.89	0.32/0.60	0.54/1.16	0.41/0.89	0.32/0.71	0.52/1.05
EqMotion [22]	CVPR'23	0.96/1.92	**0.30**/**0.58**	**0.50**/**1.10**	**0.39**/**0.86**	**0.30**/**0.68**	**0.49**/**1.03**
Social-Transmotion [16]	ICLR'24	0.93/**1.81**	0.32/0.60	0.54/1.16	0.42/0.90	0.32/0.70	0.51/**1.03**
Our w/o CLS		**0.89**/1.87	0.36/0.73	0.57/1.22	0.40/0.87	0.31/0.69	0.51/1.08

Table 4: Evaluation on ETH-UCY dataset in pedestrian forecasting (ADE/FDE). The observation is performed during 8 time-steps (3.2s) while the forecasting is performed during the subsequent 12 time-steps (4.8s). Results are extracted from previous works [16, 22].

References

1. Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., Savarese, S.: Social lstm: Human trajectory prediction in crowded spaces. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 961–971 (2016) 6
2. Alcorn, M.A., Nguyen, A.: baller2vec++: A look-ahead multi-entity transformer for modeling coordinated agents. arXiv preprint arXiv:2104.11980 (2021) 1
3. Alcorn, M.A., Nguyen, A.: baller2vec: A multi-entity transformer for multi-agent spatiotemporal modeling. arXiv preprint arXiv:2102.03291 (2021) 1
4. Becker, S., Hug, R., Hubner, W., Arens, M.: Red: A simple but effective baseline predictor for the trajnet benchmark. In: Proceedings of the European Conference on Computer Vision (ECCV) Workshops. pp. 0–0 (2018) 1
5. Ding, D., Huang, H.H.: A graph attention based approach for trajectory prediction in multi-agent sports games. arXiv preprint arXiv:2012.10531 (2020) 1
6. Girgis, R., Golemo, F., Codevilla, F., Weiss, M., D'Souza, J.A., Kahou, S.E., Heide, F., Pal, C.: Latent variable sequential set transformers for joint multi-agent motion prediction. arXiv preprint arXiv:2104.00563 (2021) 6
7. Giuliari, F., Hasan, I., Cristani, M., Galasso, F.: Transformer networks for trajectory forecasting. In: 2020 25th international conference on pattern recognition (ICPR). pp. 10335–10342. IEEE (2021) 6
8. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. pp. 249–256. JMLR Workshop and Conference Proceedings (2010) 2
9. Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A.: Social gan: Socially acceptable trajectories with generative adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2255–2264 (2018) 6
10. Huang, Y., Bi, H., Li, Z., Mao, T., Wang, Z.: Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6272–6281 (2019) 1
11. Kim, H., Choi, H.J., Kim, C.J., Yoon, J., Ko, S.K.: Ball trajectory inference from multi-agent sports contexts using set transformer and hierarchical bi-lstm. arXiv preprint arXiv:2306.08206 (2023) 4
12. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 1
13. Lerner, A., Chrysanthou, Y., Lischinski, D.: Crowds by example. Computer graphics forum **26**(3), 655–664 (2007) 6
14. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017) 1
15. Pellegrini, S., Ess, A., Schindler, K., Van Gool, L.: You'll never walk alone: Modeling social behavior for multi-target tracking. In: 2009 IEEE 12th international conference on computer vision. pp. 261–268. IEEE (2009) 6
16. Saadatnejad, S., Gao, Y., Messaoud, K., Alahi, A.: Social-transmotion: Promptable human trajectory prediction. arXiv preprint arXiv:2312.16168 (2023) 6
17. Salzmann, T., Ivanovic, B., Chakravarty, P., Pavone, M.: Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control. arXiv preprint arXiv:2001.03093 **2** (2020) 6
18. Sun, C., Karlsson, P., Wu, J., Tenenbaum, J.B., Murphy, K.: Stochastic prediction of multi-agent interactions from partial observations. arXiv preprint arXiv:1902.09641 (2019) 1

19. Teranishi, M., Tsutsui, K., Takeda, K., Fujii, K.: Evaluation of creating scoring opportunities for teammates in soccer via trajectory prediction. In: International Workshop on Machine Learning and Data Mining for Sports Analytics. pp. 53–73. Springer (2022) 1
20. Wang, Z., Liu, J.C.: Translating math formula images to latex sequences using deep neural networks with sequence-level training (2019) 1
21. Xu, C., Mao, W., Zhang, W., Chen, S.: Remember intentions: Retrospective-memory-based trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6488–6497 (2022) 6
22. Xu, C., Tan, R.T., Tan, Y., Chen, S., Wang, Y.G., Wang, X., Wang, Y.: Eqmotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1410–1420 (2023) 6
23. Xu, Y., Bazarjani, A., Chi, H.g., Choi, C., Fu, Y.: Uncovering the missing pattern: Unified framework towards trajectory imputation and prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9632–9643 (2023) 4
24. Yeh, R.A., Schwing, A.G., Huang, J., Murphy, K.: Diverse generation for multi-agent sports games. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4610–4619 (2019) 1