Hoop-MSSL: Multi-Task Self-supervised Representation Learning on Basketball Spatio-Temporal Data

Xing Wang
Universidad Politecnica de Madrid
wxing.inef@gmail.com

Jianchong Shao
Tsinghua University
shaojianchong@gmail.com

Chunyang Huang
Tsinghua University
cy-huang20@mails.tsinghua.edu.cn

Zitian Tang
Brown University
zitian_tang@brown.edu

Miguel Ángel Gómez Ruano
Universidad Politecnica de Madrid
miguelangel.gomez.ruano@upm.es

Shaoliang Zhang
Tsinghua University
ZSL.INEF@gmail.com

Konstantinos Pelechrinis
University of Pittsburgh
kpele@pitt.edu

Abstract

Observing and identifying on-court behaviors by basketball players, engaging in intricate spatio-temporal interactions with their teammates and the opponent players, have long been considered challenging tasks for machines. Early approaches focused on supervised learning to capture spatio-temporal information and role relationships between players. These frameworks relied on labeled data and were unable to be generalized to other tasks. To addressed these limitations, some recent works has drawn inspiration from the field of autonomous driving to develop self-supervised learning frameworks for trajectory data. However, these frameworks mainly focus on single tasks such as trajectory reconstruction or prediction and do not take into account the domain knowledge in basketball. In this work, we propose Hoop-MSSL, a multi-task self-supervised representation learning framework to handle complex interactions and dependencies among spatio-temporal data on basketball court. Specially, Hoop-MSSL integrates masking augmentation and three pre-training tasks for (i) motion reconstruction, (ii) player-role identification and (iii) contrastive learning, to capture the spatio-temporal features and the role relationships across multiple dimensions. To evaluate the efficacy of Hoop-MSSL, we conducted extensive line-probing experiments on three downstream tasks. Our results demonstrate that the synergistic interaction among all of the Hoop-MSSL components helps the model to learn more general spatio-temporal representations, allowing it to achieve better performance on all downstream tasks as compared to using only subsets of the components. Finally, a high masking ratio (80%) can further enhance significantly the model's ability to learn useful representations.

1 Introduction

Learning useful representations for multi-agent behavior has received considerable attention recently from sports analytics researchers and practitioners. Sports such as, basketball, soccer and hockey, are multi-agent systems where the players interact within a shared environment, following an underlying dynamical process that may be stochastic and often infeasible to characterize analytically due to

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

the complex interactions involved. To capture the nuanced patterns and dependencies from these complex interactions, self-supervised representation learning (SSL) has been used to derive low-dimensional features from large amounts of high-dimensional data. A key advantage of these methods is that SSL algorithms can directly learn through supervision signals automatically extracted from unlabeled tracking data, thus, eliminating the need for manual annotation. Most existing multi-agent behavior SSL frameworks employ single pretext tasks. These are focused on rather narrow objectives of trajectory prediction or reconstruction modeling and are widely used in fields like robotics and autonomous driving. In this type of applications, modeling motion information based on past trajectories is a reasonable approach. However, in team sports analysis this single SSL task is not sufficient to deal with important discriminative tasks for coaches and players, such as identifying offensive set plays and defensive formations.

To overcome this drawback, in this work we propose the intergration in our framework of two additional self-supervised elements/tasks on the raw spatio-temporal data. The first is a "Player-role Identification" task (PI). While it might be relatively easy for humans to visually identify whether a player is on offense or defense, it is rather challenging for a machine to automatically do this. In fact, even for humans it might can get hard if we just show them the trajectories of the players, without any team identifiers (e.g., jersey colors, ball possession etc.), as seen in Figure 1. In previous studies [1, 7, 8], this role information was used as one of the input features of the corresponding models to maintain the consistency of each player's positional data over time and to avoid misalignments. However, based on the current Transformer architecture [6, 9], It is possible to align players' trajectories without role information, thus *forcing* the model to learn to discriminate between offensive and defensive roles of anonymous player trajectories. This approach could enable the model to generate more generalized representations of the underlying interactions between players through their spatio-temporal trajectories.

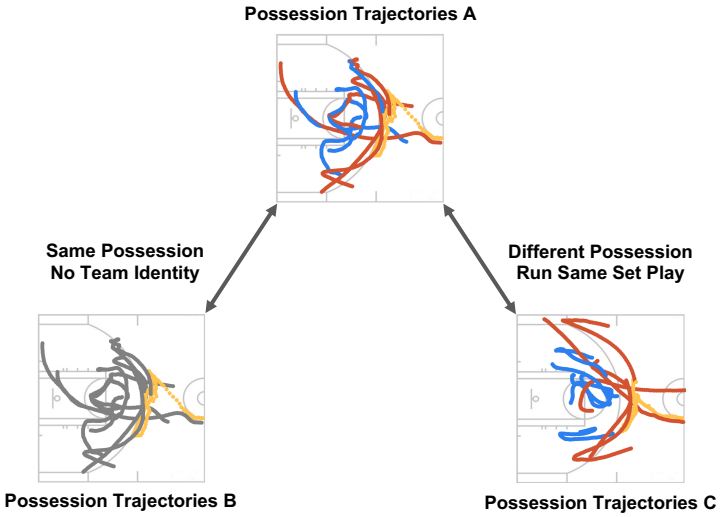


Figure 1: The top row depicts a possession from our data where the offense runs a specific set play called "Weave". The identify of the team is encoded on the color of the trajectories, with red being the offense and blue the defense. The possession trajectories B figure depicts the same possession but without team identity (i.e., player trajectories are grey). Visually, it is hard to identify the offense and defense from the grey trajectories. However, using the interaction spatio-temporal features from these trajectories, will facilitate the identification of the team. The "Player-role Identification" task helps our model to extract interaction features from these trajectories. The right figure in the second row depicts another possession where the offense runs the same "Weave" set play. Contrastive learning will help Hoops-MSSL to bring these two possessions closer together in latent space.

The second element we add in our framework is "Contrastive Learning" task (CL). CL enables models to map similar instances close together in latent space while pushing apart those that are dissimilar. It has proven to be effective in computer vision (CV) and natural language processing (NLP) for tasks such as image classification, object detection, sentence embedding, and text classification. In team sports, the agents' trajectories within a possession encapsulate complex patterns and movements that

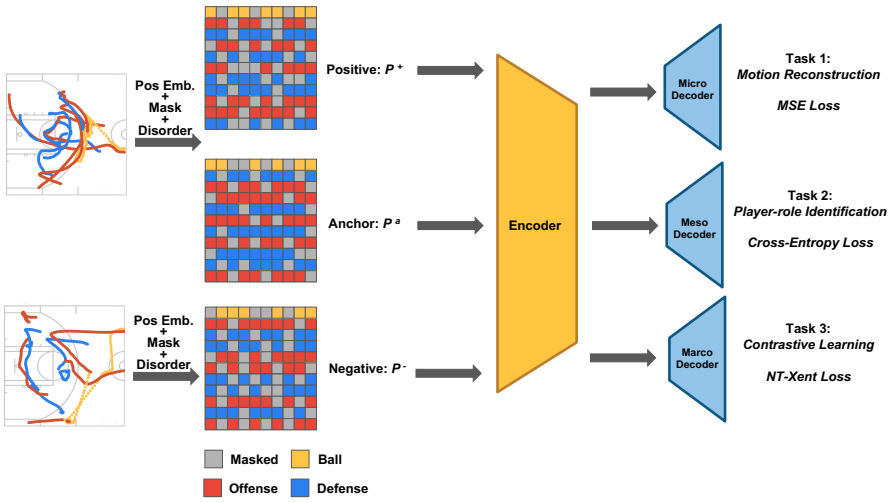


Figure 2: Given a set of N possessions containing the trajectories of 10 players (no team identity) and ball, generate 2N samples by random masking and shuffling augmentation. Each sample and its corresponding augmented version are considered positive pairs, while all other samples are treated as negative pairs. All samples were encoded using the Transformer-based encoder. The encoded representations are then processed through three different decoders corresponding to three pretext tasks.

reflect macro-level tactical intentions, akin to how visual or textual data carry high-level semantic information. Particularly in basketball, where there is a fixed time for offense, the offensive players move according to some pre-designed, structured offensive strategies (called *set plays*) designed by coaches, which can be viewed as analogous to images or sentences. By using CL, possessions where the offense runs the same set play will be mapped close to each other in the latent space, while possessions that run different set plays will be driven farther apart. This approach enhances the model's ability to understand and predict strategic plays, providing deeper insights into the team's offensive dynamics and facilitating better tactical analysis and decision-making.

With these tasks set, we explore different combinations of training strategies on basketball tracking data to learn the representation model. This will allow us to fully utilize the representations learnt to successfully complete a variety of downstream tasks. We provide a comprehensive evaluation and analysis in our experiments that clearly demonstrate the benefits achieved from Hoop-MSSL. Our contributions can be summarized in the following:

- We propose a multi-task self-supervised learning framework for multi-agent trajectories representation learning in sports. The framework's objective is to learn comprehensive and general feature representations without the need of labeled data. Hoop-MSSL utilizes 3 separate self-supervised elements; "Motion Reconstruction", "Player-role Identification" and "Contrastive Learning".
- We find that masking a high proportion of the input trajectories, e.g., 80%, optimizes the performance of each self-supervised learning task.
- We perform a thorough evaluation of Hoops-MSSL on basketball data including player spatio-temporal data. Our experiments validate the capacity of the learned representations to successfully complete a diverse set of downstream tasks.

2 Methodology

We introduce Hoop-MSSL, a self-supervised representation learning framework for basketball, that is able to capture interaction and motion embeddings using three pretext tasks. These pretext tasks are designed using basketball domain knowledge and correspond to three different abstraction levels.

We follow the axial-attention based Encoder-Decoder architecture from HoopTransformer [9] and set a separate Decoder for each pretext task. We begin by sampling N offensive possessions from dataset $\mathcal{D}$ to construct a mini-batch $I = \{P_i | 1 < i \leq N, i \in \mathbb{N}\}$, each containing the trajectories $\tau = [(x_0, y_0), (x_1, y_1), ...]$ of M players and ball $P = \{\tau_a | 1 \leq a \leq M, a \in \mathbb{N}\}$. Each offensive possession undergoes position embedding, masking, and disorder operations in the preprocessing module, generating two mutually positive instances $I' = \{P_i^{postive}, P_i^{anchor} | 1 < i \leq M, i \in \mathbb{N}\}$ that are then fed into the Encoder. The position embeddings are the same as the Transformer, which enforces the temporal structure in the self-attention operations. The random masking operation are the same as in BERT, combined with the disorder operation to generate two different views of one possession sample. Each possession sample is a tensor [A,T,D] (A: 11 agents; T: 121 timesteps; D: 3 x-coordinate, y-coordinate, and velocity). The three pretext tasks (detailed in what follows) are trained simultaneously and share the same parameters of the encoder. After training Hoop-MSSL can model all the observed player trajectories as spatio-temporal embeddings and implicitly provide useful tactical information when applied on downstream task (as we detail in Section 4).

2.1 Hoop-MSSL's Self-Supervised Tasks

As aforementioned, Hoop-MSSL integrates three pretext tasks for interaction and motion modeling: 1) motion reconstruction, 2) player-role identification, and 3) contrastive learning. For motion reconstruction, Hoop-MSSL learns micro level information from the raw trajectories interacting on the basketball court through the encoder. For the PI task, the encoder captures information related to the *social* relationships between the agents, regardless whether they are teammates or opponents. For the contrastive learning element, the encoder captures the tactical information (set plays), typically designed prior to the game.

Task 1: Motion Reconstruction (MR) Compared to motion prediction task, motion reconstruction entails bidirectional information transfer and its objective is to reconstruct the full trajectory from the incomplete trajectory, which means that we can predict the missing parts in between based on past and future trajectories. The encoder operates only on the visible subset of trajectory, while the decoder reconstructs the original trajectory from the latent representation and mask tokens. We use the mean squared error (MSE) as the loss function for this task.

$$L_{MR} = \frac{1}{n} \sum_{a=1}^{n} (y_a^t - \hat{y}_a^t)^2$$

Task 2: Player-role Identification The offensive and defensive players in a possession have different objectives, which guide their behaviors. The relationship between these behaviors can generally be classified into two categories: cooperative (i.e., players on the same team) and adversarial (i.e., players on different teams). We define a classification problem for player roles, which allows us to capture the latent relationship between players through their latent representation. The decoder contains a pooling layer and a classification header. The pooling layer is used to generate the player embedding, while the classification header performs binary classification with cross-entropy as the loss function.

$$L_{PI} = -(y \log(p) + (1 - y) \log(1 - p))$$

Task 3: Contrastive Learning The key to training in CL lies in constructing effective positive and negative sample pairs. These pairs help the model learn to distinguish between similar and dissimilar data points in the embedding space. There are various data augmentation methods for constructing positive samples, such as rotation, cropping and dropout. In our work, we directly applied the masking strategy from the previous motion reconstruction task as our data augmentation method. Specifically, we performed two random mask operations on each of the N possessions in a minibatch, resulting in 2N masked possessions. Each masked possession is trained to identify its counterpart among the $2 \cdot (N - 1)$ in-batch negative samples. Following previous studies [2, 10], we adopt the normalized temperature-scaled cross-entropy loss (NT-Xent) as the contrastive objective.

$$L_{CL} = -\log \frac{\exp(\text{sim}(r_i, r_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(r_i, r_k)/\tau)}$$

Having defined all 3 elements of our framework, in the final stage we define the overall objective function of Hoop-MSSL as:

$$L_{pre} = L_{MSE} + \lambda_1 L_{PI} + \lambda_2 L_{CL}$$

It is worth noting that the masking ratio plays a crucial role in pre-training. In BERT [3], only 15% of the words in the input sequence are masked. However, in MAE [5], the authors found that hat a higher masking ratio of 75% is more effective for image. Therefore, in our experiments we will experiment with different mask ratios.

Acknowledgments and Disclosure of Funding

References

[1] Michael A Alcorn and Anh Nguyen. baller2vec++: A look-ahead multi-entity transformer for modeling coordinated agents. *arXiv preprint arXiv:2104.11980*, 2021.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Christoph Feichtenhofer, Yanghao Li, Kaiming He, et al. Masked autoencoders as spatiotemporal learners. *Advances in neural information processing systems*, 35:35946–35958, 2022.

[5] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[6] Jiquan Ngiam, Benjamin Caine, Vijay Vasudevan, Zhengdong Zhang, Hao-Tien Lewis Chiang, Jeffrey Ling, Rebecca Roelofs, Alex Bewley, Chenxi Liu, Ashish Venugopal, et al. Scene transformer: A unified architecture for predicting multiple agent trajectories. *arXiv preprint arXiv:2106.08417*, 2021.

[7] Shayegan Omidshafiei, Daniel Hennes, Marta Garnelo, Zhe Wang, Adria Recasens, Eugene Tarassov, Yi Yang, Romuald Elie, Jerome T Connor, Paul Muller, et al. Multiagent off-screen behavior prediction in football. *Scientific reports*, 12(1):8638, 2022.

[8] Kuan-Chieh Wang and Richard Zemel. Classifying nba offensive plays using neural networks. In *Proceedings of MIT Sloan sports analytics conference*, volume 4, 2016.

[9] Xing Wang, Zitian Tang, Jianchong Shao, Sam Robertson, Miguel-Ángel Gómez, and Shaoliang Zhang. Hooptransformer: Advancing nba offensive play recognition with self-supervised learning from player trajectories. *Sports Medicine*, pages 1–11, 2024.

[10] Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. Consert: A contrastive framework for self-supervised sentence representation transfer. *arXiv preprint arXiv:2105.11741*, 2021.

A Experiments

In this section we start by describing the dataset we used and the implementation of our framework. We then describe in detail the downstream tasks that we will use to evaluate Hoops-MSSL and the results, including an ablation study for the framework's 3 elements.

A.1 Dataset

As aforementioned, to build Hoops-MSSL we use a unique dataset composed of 632 NBA games from the 2015-16 regular season collected from the SportVU system[1]. Of primary interest is the optical tracking data which provide the location of each player using rectangular coordinates along the (x, y)-plane, as well as, the court location and height (3-dimensional) of the ball. During each second of gameplay, this information is recorded 25 times at evenly spaced intervals. The data is extensively annotated allowing for labeling of specific events and possession outcomes. Using this annotation we can segment the game into team possessions possessions, that is segments during which one team maintains control of the ball. Additionally, the data provide information about the specific players that are on the court as well as how much time remains on the shot-clock (which regulates the remaining maximum length of the possession).

Teams start their set plays once the ball crosses the mid point of the court in the x-dimension (wide) and hence, to better capture the corresponding tactical movement features on the court, we eliminated the parts of the possessions where the ball is still on the defensive half of the team. In addition, all possessions and location data were mapped to the left half court and subsampled at 5 HZ, as in previous studies [9]. Ultimately, a total of 90,524 possessions were used in our study. In the pre-training process, the dataset was divided into 80% training data and 20% validation data.

A.2 Implementation Details

The Encoder of Hoop-MSSL consists of 5 sets of axis-factorized attention layers, with each set containing one temporal attention layer and one spatial attention layer, following previous literature [9, 7, 6]. In the Encoder, the data are passed in the form of a tensor with the shape [11, 121, 256], where the temporal attention focuses on the time dimension and the spatial attention focuses on the agent dimension. The Decoder for the MR task consists of 2 sets of axis-factorized attention layers and one multilayer perceptron (MLP) layer that predicts the masked data points using the MSE loss. The Decoder for the PI task consists of a 1D pooling layer for the temporal dimension and an MLP layer to generate "player embedding" and predict the player's role using the cross-entropy loss as described above. The Decoder of the CL consists of a 2D pooling layer and an MLP layer to generate "possession embedding" and calculate cosine distance between possessions.

We applied masking ratios ranging from 10% to 90% and trained 50 epochs for each model. During these sessions, we set both regularization parameters λ_1 and λ_2 to 100, with the learning rate of 5e–8.

A.3 Evaluations

To reiterate we assess the model's performance using linear probing on three different levels of downstream tasks:
Play-level task: This task involves identifying the offensive set plays of the team on offense, which includes 14 different offensive set plays such as 'elevator', 'floppy', and 'horns' (more details see Appendix D). This task requires obtaining a *possession embedding* that represents the spatio-temporal information of all players within the possession. This task is analogous to the image classification task in CV.
Action-level task: This task focuses on recognizing three main offensive actions within a possession: pick&roll, off-ball screen, and hand-off. This task requires obtaining an *action embedding* that represents only the spatial information of offensive players within the possession. This task employs focal loss, similar to its application in object detection tasks within computer vision.
Player-level task: This task focuses on identifying the role of the players involved in a pick&roll of ball-handler, screener, ball-handler's defender and screener's defender in a pick&roll scenario. It requires the development of a *player embedding* that captures only the temporal information relevant to each player's role. This task is analogous to pixel-level tasks in CV.

A.4 Results

We first start by evaluating the impact of masking on Hoop-MSSL's performacne. Figure 4 shows the influence of the masking ratio on the three downstream tasks. The optimal ratio is surprisingly

[1] `https://www.statsperform.com/team-performance/basketball/optical-tracking/`

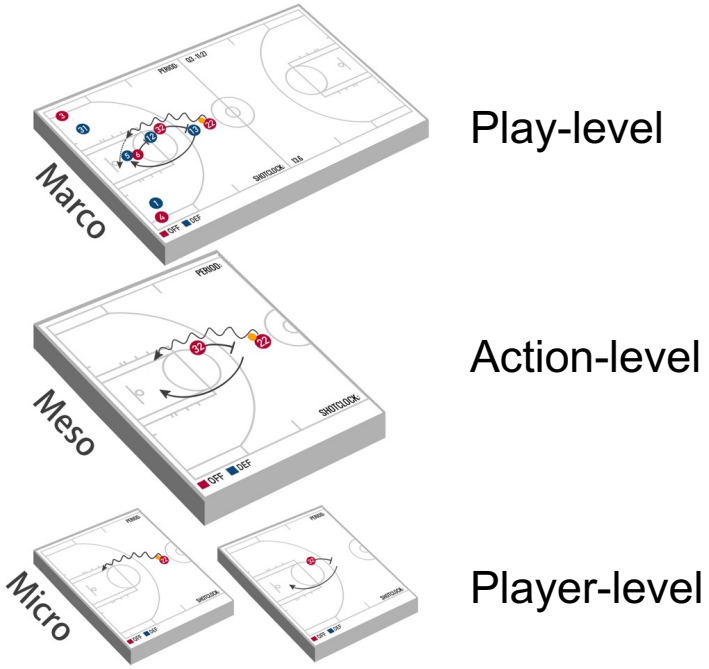


Figure 3: Illustration of three downstream task across different levels. The play-level downstream task involves identifying the offensive set plays of the offense team. The action-level downstream task focuses on recognizing three main offensive actions within a possession. The Player-level is to identify the role of players in a pick&roll scenario.

high for every task. For the play-level task and the action-level task, the accuracy of top-1 at the validation set peaks when the mask ratio is 80%. In the player-level task, the optimal mask ratio is 90%. This result is consistent with the typical masking ratio of 75% on images [5] and 90% on videos [4]. Previous literature [5, 4] has hypothesized that the high masking ratio is related to the high information redundancy in the data, and, we believe that the 80% masking rate observed in our study to provide optimal performance is a reasonable and expected outcome. A lot of the player movements during a play are "decoy" actions, with the goal of misdirecting the defense, and only a small part of the trajectory is the the core set play. Additionally, the varying trends observed across

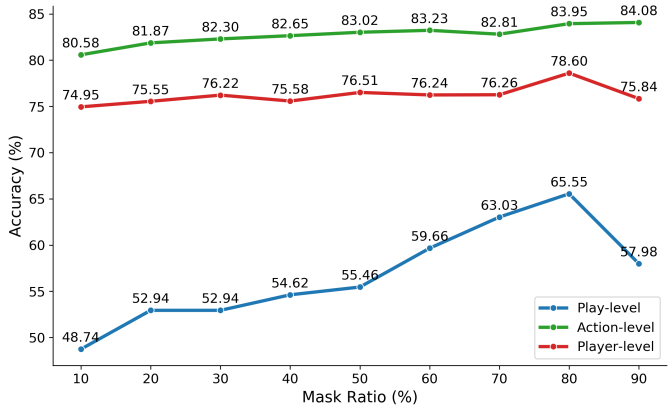


Figure 4: Masking ratio. A high masking ratio (80%) works well for all three downstream tasks. The y-axes are top-1 validation accuracy (%).

the three tasks indicate that higher difficulty downstream tasks are more sensitive to the masking

ratio. In play-level task, the accuracy increases steadily with the masking ratio: the accuracy gap is up to 20%(48.74% vs. 65.55%). In action-level and player-level tasks, the results are less sensitive to the ratios, and a wide range of masking ratios work well. This may be related to the difficulty of the downstream tasks, as the play-level task requires robust representation of both spatial and temporal dimensions.

A.5 Ablation Study

To further evaluate the contribution of each element of Hoop-MSSL, we compare the performance of of our framework when trained on different combinations of its components (with a masking ratio of 80%).

Table 1: Ablation study of pretext tasks on play-level downstream task.

Combination of pretext tasks	Top-1	Top-3
MR + PI	51.26	74.79
MR + CL	61.34	84.87
CL + PI	63.87	84.03
MR + CL + PI	65.55	89.92

Table 1 shows the top-1 and top-3 accuracy at the play-level downstream task. As we can see, the performance of the framework when we remove the CL element (MR + PI) is significantly worse compared to the other three models. These results demonstrates that the CL element plays an essential role at the play-level classification task. Without CL task, the model fails to effectively distinguish between different *possession embeddings* in the representation space. Furthermore, the model incorporating all pretext tasks shows (as expected) the highest accuracy, achieving 65.55% in the top-1 metric and 89.92% in the top-3 metric. This also suggests that while CL is the most important piece for the play-level identification task, both the MR and the PI pretext tasks provide additional useful information for this task.

Table 2 shows the results for the player-level downstream task. As we can see when the model incorporates all pretext tasks it performs better, achieving 78.60% in the top-1 accuracy and an F1-score of 69.26%. Note that for the player and action-level downstream tasks we use the F1-score instead of top-3 accuracy, since there are only 4 labels to be predicted (while for the play-level downstream task there are 14 labels). As expected, the lowest performance among the models was observed when the PI pre-text task was removed from the framework. However, the gap between the models is smaller (as compared to the play-level downstream task) and this is most probably due to the fact that role relationships between players are also learned in the MR and CL pretext tasks.

Table 3 presents the results for the action-level downstream task. As we can see, the CL pretext task had almost no effect on the model's performance. This most probably is related to the nature of the action-level task, which primarily relies on localized spatial information rather than the temporal semantic context that CL typically enhance. Therefore, the benefits of contrastive learning may not be as pronounced in this context.

Overall, this ablation study shows that each of the pre-text tasks that we incorporated in Hoop-MSSL provide useful information for a diverse set of downstream tasks. While there is some degree of information overlap among the representations learned from each pre-text task, the optimal performance is achieved when all of the components are combined.

Table 2: Ablation study of pretext tasks on player-level downstream task.

Combination of pretext tasks	Top-1	F1
MR + PI	74.62	65.42
MR + CL	73.48	64.05
CL + PI	75.12	64.30
MR + CL + PI	78.60	69.26

Table 3: Ablation study on pretext tasks using action-level downstream task.

Combination of pretext tasks	Top-1	F1
MR + PI	83.66	75.36
MR + CL	79.99	75.36
CL + PI	82.89	77.16
MR + CL + PI	83.95	78.71

B Conclusions

This work presents Hoop-MSSL that employed multi-task self-supervised representation learning based on Transformers, to learn better representations for performing the multi-level downstream tasks in basketball in the basketball domain. The pre-training task of Hoop-MSSL enhances the model's ability to handle complex interactions and dependencies among players. By designing the pretext tasks of MR, PI and CL, along with masking augmentation, Hoop-MSSL can capture the spatio-temporal features and the social relationships within multi-players tracking data from multiple dimensions. Quantitative experiments that Hoop-MSSL outperforms other pretext task combinations across three different levels of downstream tasks in the basketball domain and a high masking ratio (80%) can significantly enhance the model's ability to learn useful representations. Future work involves enhancing the representation capabilities of Hoop-MSSL by incorporating graph neural network architectures and various data augmentation techniques. Additionally, we aim to extend the application of this model to other sports domains.

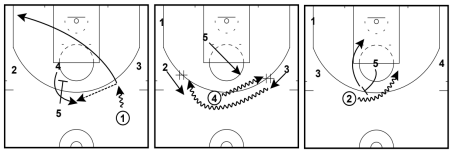
C Model Details

Meta-Arch	Layer	Input	Operation	Across	Atten Matrix	Output Size
Encoder						
	A	Agents	MLP + BN	-	-	[A,T,D]
	B	A	Transformer	Time	[A,T,T]	[A,T,D]
	C	B	Transformer	Agents	[T,A,A]	[A,T,D]
	D	C	Transformer	Time	[A,T,T]	[A,T,D]
	E	D	Transformer	Agents	[T,A,A]	[A,T,D]
	F	E	Transformer	Time	[A,T,T]	[A,T,D]
	G	F	Transformer	Agents	[T,A,A]	[A,T,D]
	H	G	Transformer	Time	[A,T,T]	[A,T,D]
	I	H	Transformer	Agents	[T,A,A]	[A,T,D]
	J	I	Transformer	Time	[A,T,T]	[A,T,D]
	K	J	Transformer	Agents	[T,A,A]	[A,T,D]
Decoder MR						
	L	K	Tile + MLP	-	-	[F,A,T,D]
	M	L	Transformer	Time	[A,T,T]	[F,A,T,D]
	N	M	Transformer	Agents	[T,A,A]	[F,A,T,D]
	O	N	Transformer	Time	[A,T,T]	[F,A,T,D]
	P	O	Transformer	Agents	[T,A,A]	[F,A,T,D]
	Q	P	Layer Norm	-	-	[F,A,T,D]
	R	Q	MLP + BN	-	-	[F,A,T,4]
Decoder PI						
	S	K	Pooling + MLP	Time	-	[A,D]
	T	S	MLP	-	-	[A,2]
Decoder CL						
	U	K	Pooling	Time	-	[A,D]
	V	U	Reshape	-	-	[B,11,D]
	W	V	Pooling	Agents	-	[B,D]
	X	W	MLP	-	-	[B,128]
Transformer heads			4			
Length of Time (T)			121			
Feature Dimensions (D)			256			
Optimizer			Adam $\alpha = 1 \times e^{-7}, \beta_1 = 0.9, \beta_2 = 0.999$			
Learning Rate Schedule			Total epochs: 50; OneCycleLR: $max\, lr = 5 \times e^{-7}$			
Batch size (B)			16			
Future classification weight			0.1			
Position classification weight			1.0			
Laplace Target Scale			1.0			
Temperature (τ)			0.1			

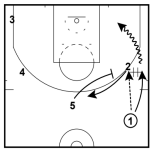
Table C.1: Model architecture. The model receives as input of A agents across T time steps and K features. K is the total number of input features (x-coordinate, y-coordinate and velocity). A subset of these inputs are masked. The input is fed into the encoder and mapped through the MLP layer into [A, T, D] (D = 256). After 10 axial-attention layers, the tensor is fed into the decoder and output a shape of [F, A, T, 4] to calculate loss with masked ground truth (F = 6). All layers employ ReLU nonlinearities.

D Set Play Details (Play-level task Labels)

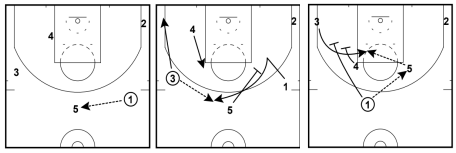
Weave



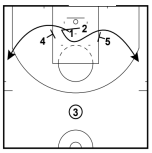
21 Pistol



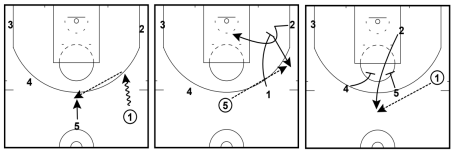
Horns Elbow



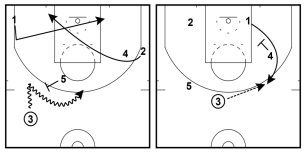
Floppy



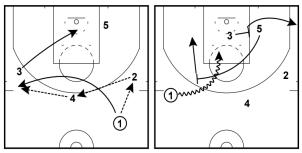
Elevator



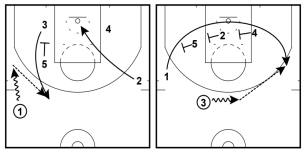
Drag Pindown



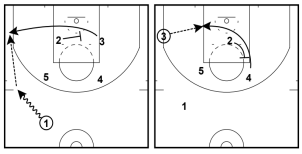
Motion Weak



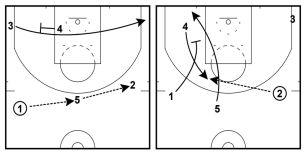
Zip Loop



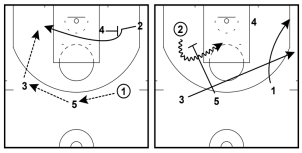
Cyclone



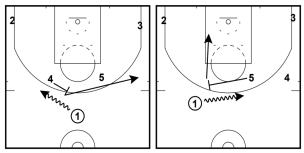
4 Pop



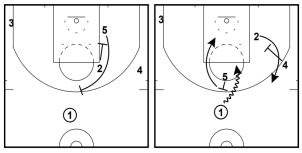
2 Down



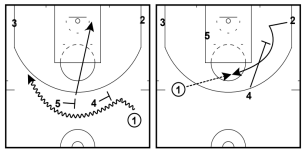
Horns Twist



Ram Exit



Double Drag



NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and/or introduction clearly state the claims made.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [No]

   Justification: The limitation of this study is the size of the dataset, resulting in results that are not necessarily robost

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All the theorems, formulas, and proofs in the paper was numbered and cross-referenced.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We give the experimental results and details in appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We would release the code and dataset.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The details is in appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Error bars are not reported because it would be too computationally expensive

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We reported the computer resources we used.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper complies with the NeurIPS ethical guidelines in all respects

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: there is no societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: the paper poses no such risks

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cited the relevant literature

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.