Unsupervised modeling of the movement of basketball players using a Deep Generative Model

David Acuna
University of Toronto
davidj@cs.toronto.edu

Abstract

We present a deep generative model that is able to synthesized the movement of a given basketball player conditioning on the trajectory of the ball. For that, we used the freely available NBA SportVU tracking data and a Conditional Variational Autoencoder (CVAE). Similar to previous approaches, we capture the movement of the player and the ball using pictorial representations. The temporal aspects of the trajectories are encoded using 'fading'. We show that our architecture is able to correctly model the movement of the players, the movement of a given player, and the movement of a given player conditioning on the trajectory of the ball. To the best of our knowledge, this work constitutes one of the first attempts to synthesize the movement of basketball players using a deep generative approach.

1 Introduction

Basketball is fundamentally a game of effective movement, precision passing, communication and team chemistry. However, when it comes to pure effectiveness on the basketball court, it is the ability to know when and where to go what separates the great players from the rest. As a result, STATS SportVU [1] utilizes a six-camera system to track the real-time positions of players and the ball 25 times per second during the game.

This combination of player and ball tracking statistics has led to an incredible amount of raw data for basketball analytics. This, together with the extraordinary success of Deep Neural Networks in areas such as Computer Vision and Natural Language Processing [2; 3], and the capacity of such models to find insights into large volume of data [4] , has significantly increased the attention of neural networks based architecture for basketball analytics.

For example, in [5] , the authors applies Recurrent Neural Networks [6] in the form of sequence modeling to predict whether a three-point shot is successful. They show that their model is capable of learning the trajectory of a basketball without any knowledge of physics. Similarly in [7], the authors predict the likelihood of a player making a shot. For that, they use pictorial representation of the movement of the players and a Convolutional Neural Network (CNN) [2]. They also capture the temporal aspect of the trajectories using fading.

Wang and Zemel in [8] also use pictorial representations and neural networks to classifying NBA Offensive Plays. In addition to the accuracy of their model, they showed that their approach achieves good recognition rates when trained on one season and tested on the next.

Inspired by these previous works and the capacity of generative models to synthesized images that captures the overall coherence of the original examples, we propose to model the movement of each basketball player and the ball using pictorial representations.

Generative modeling loosely refers to a branch of unsupervised learning techniques which build a model of the data [9], for instance $p(image)$, such that we can sample from it. This is in contrast

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

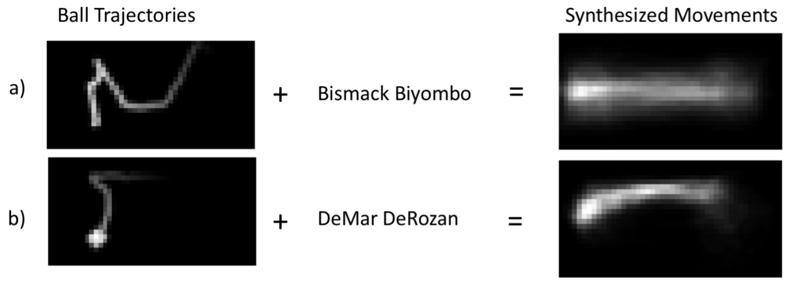

Figure 1: Synthesized offensive movement of two given players conditioning on the trajectory of the ball. Not only does the model has learned to pay attention to the ball and the basketball position of the players, but it has also learned to model the temporal aspects of the movement encoded as 'fading'. a) Player Position: Center b) Player Position: Shooting Guard

to discriminative modeling, such as regression or classification, which tries to estimate conditional distributions such as $p(class|image)$. State-of-the-art generative models include Variational Auto-encoders(VAE) [10] and Generative Adversarial Networks (GAN) [11]. These models have been proved capable of synthesized data, mostly images, that captures the overall coherence of the original examples.

In this paper, we show the capacity of deep generative models to synthesized the movement of basketball players with and without conditioning them, on the player id, the trajectory of the ball or both. For that, we firstly train a vanilla VAE to generate pictorial representation of the movement of basketball players. Then we extend the model and use a Conditional VAE (CVAE) [12; 13] to also synthesized pictorial representation of the players movements, however, this time conditioned on a given player id. Finally, we illustrates the results we obtained by conditioning the CVAE on both the player and the trajectory of ball.

2 Method

2.1 Data representation

The data used in this study comes from the publicly available SportVu tracking and play-by-play data for 42 Toronto Raptors games played in Fall 2015. SportVu [1] is an optical tracking system installed by the National Basketball Association (NBA) in all 30 courts to collect real-time data. The tracking system records the spatial position of the ball and players 25 times a second during a game. Along with the coordinates, it provides a unique player identifier. Based on that, we use each player coordinates to create pictorial representation of each player and ball positions.

The number of images generated was proportional to the number of offensive events per game per player plus the trajectory of the ball in the offensive event of those same games. Figures 2a and 2b show an example of the player and ball's trajectory for the same offensive event. To simplify our input representation further, the generated pictorial representations were set to be 24x46 gray-scale images. Note also that, in order to reduce the computational complexity of the training process, we only model the movement of the Toronto Raptors players.

2.2 The Generative Architecture

Let's define our dataset containing pictorial representation of each of the player's movement to be $X = \{x^{(i)}\}_{i=i}^{N}$ where each of these images $\mathbf{x}^{(i)} \in \mathbb{R}^{24\times 46}$ represents the movement of a given player during an offensive event. Let's also assume that the data is generated by some random process, involving an unobserved continuous random variable $\mathbf{z}$.

In the VAE framework, we can define $\mathbf{z}$ to be a latent representation of the data, which is of lower dimensionality than the input data, and it is the output of a neural network $q_\phi(\mathbf{z}|\mathbf{x})$.This is typically referred to as a 'bottleneck' because the encoder must learn an efficient compression of the data into this lower-dimensional space [10].

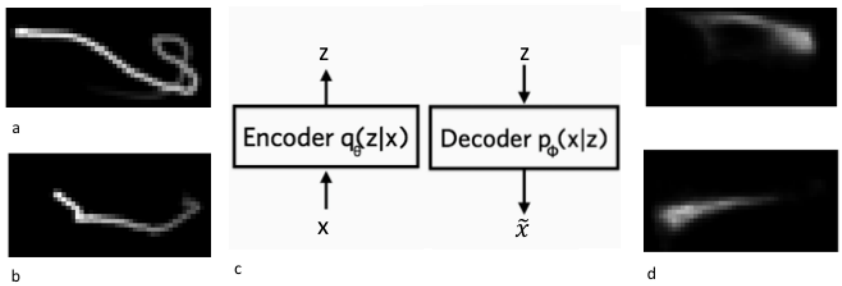

Figure 2: a) Pictorial representation of the movement of a basketball player for a particular offensive event. b) Representation of the trajectory of the ball for the same offensive event. c) VAE framework. The encoder and decoder networks are learned end-to-end by maximizing the evidence lower bound (ELBO). In the case of the conditional VAE, both probability distributions q_θ and p_ϕ are conditioned on the ball and the player id. d) Synthesized movement of two different players obtained by sampling z from a Gaussian prior and feeding it to the decoder.

Following on that, we can also define a another neural network that model $p_\theta(\mathbf{x}|\mathbf{z})$ and ideally will reconstruct the data $\mathbf{x}$ given the latent representation $\mathbf{z}$. In practice, however, information is lost because it goes from a smaller to a larger dimensionality. Figure 2c shows the architecture of the described model.

The beauty of the architecture is that we can learned the parameters of the two networks,namely θ and ϕ, in an end-to-end fashion [10; 13]. For that, we just need to trained the networks to maximize the evidence lower bound (ELBO) shows in equation 1.

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{q_\theta(\mathbf{z}|\mathbf{x})}[\log p_\phi(\mathbf{x}|\mathbf{z})] - KL(q_\theta(\mathbf{z}|\mathbf{x})||p(z)) \tag{1}$$

Note that, after learning the decoder network, we can easily generate pictorial representation of the movement of the players by sampling $\mathbf{z}$ from the Gaussian prior $p(z) \sim \mathcal{N}(0, \mathbf{I})$ and feeding the vector into the decoder.

2.3 The Conditional Generative Architecture

The conditional Variational Autoencoder(CVAE) works similarly to the VAE described before. The main difference is that during training a condition $\mathbf{c}$ is fed to both the encoder and the decoder [12; 13]. In terms of optimization, we can use the same ELBO just conditioning the likelihood and the variational posterior on $\mathbf{c}$. Equation 2 shows the ELBO for the Conditional VAE.

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{q_\theta(\mathbf{z}|\mathbf{x},\mathbf{c})}[\log p_\phi(\mathbf{x}|\mathbf{z}, \mathbf{c})] - KL(q_\theta(\mathbf{z}|\mathbf{x}, \mathbf{c})||p(z)) \tag{2}$$

To generate fake data, we just sample z from the Gaussian prior $p(z) \sim \mathcal{N}(0, \mathbf{I})$ and feed this vector and the condition $\mathbf{c}$ into the decoder. For the purpose of this paper, c constitutes either the player id , the pictorial representation of trajectory of the ball or both. In this last case, c is a combination of them.

3 Experiments and Results

As described before, we trained three generative models, the first one was a vanilla VAE trained to model the trajectory of the basketball players. In this model, we didn't make any distinction between players. We trained for 800 epochs on the 45635 pictorial representation of the players movements. We used Adam optimizer and we kept the mini-batch size to be 64. During training, we used a test set of 1000 pictorial representations to evaluate the average loss of the reconstruction and based on that, we determined the hyper-parameters mentioned before.

The encoder architectures for this model was composed of two simple fully connected layers. The first one reduced the dimension of the image to 400 and the second one to 20. We used a similar number

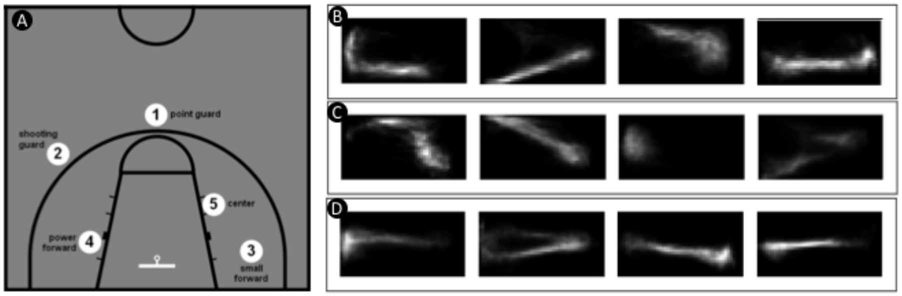

Figure 3: a) Basketball positions in the court. b, c, d) Synthetically generated images conditioning on the movement of the following NBA players: Kyle Lowry - PG , DeMar DeRozan - SG , Bismack Biyombo - C. Not only is the model able to model the movement of the player during an offensive event, but it is also able to associate, in a completely unsupervised fashion, their trajectories to their given positions.

of layer in the decoder just in the reverse order. Figures 2a and 2d compare random samples generated from the model with original data. We can see how the VAE learned the pictorial representation of the movement of the players, and how it is able to generate sensible trajectories that resembles the original ones.

The second model that we trained was condition on the player id. For simplicity, we kept the same encoder/decoder architecture and we used Adam to minimize the negative of equation 2. We also kept the same hyper-parameters mentioned before.

After training, we sampled from the model conditioning on the player id. The main idea was to see whether the synthetically generated representations were able to capture the movement of that given player during an offensive event. Figures 3b, 3c and 3d show three randomly trajectories conditioning on the following three NBA players: Kyle Lowry, DeMar DeRozan and Bismack Biyombo. It can be seen from the picture that in addition to model the movement of the players during an offensive event, the model is also able to associate, in a completely unsupervised fashion, their trajectories to their given positions. For example, it can be seen from figure 3d that the samples corresponding to Bismack Biyombo mostly reflect movements in the middle of the court. This behavior was expected and desired as he spent most of his time playing as a center during Fall 2015. The basketball positions in the court are shown in figure 3a to give the reader the possibility to freely compare.

Our final model was conditioned on both the player id and the pictorial representation of the trajectory of the ball. It is important to highlight that in this case, we had to increase the deepness of the encoder/decoder neural networks. For the encoder, we added an initial fully connected layer that reduced the dimensionality of the image to 800 and kept the other two layers identically. As stated before, it is exactly the same for the decoder just in the reverse order. Everything else was remained the same, including the optimization hyper-parameters and the optimizer.

Figure 1 illustrates an example of two synthetically generated movement representations. This time conditioning on the two different player ids and ball trajectories. Note how the model learned to pay attention to the ball, to the basketball position of the players and to the temporal aspects of the movement (encoded as 'fading' in the original representations).

4 Limitations

Although our approach was able to generate synthetic images that model the trajectory of the basketball players, we noticed that in some cases the generated trajectories were blurry and in others we couldn't see the trajectory at all. We hypothesize that this happens because the simplicity of our encoder/decoder didn't give to the model complexity enough. For example, we noticed a significantly increase in the performance of the CVAE when we just added another fully connected layer to the architecture. However, we couldn't extend more the capacity of the networks as we didn't have enough computational resources.

5 Conclusions

We have presented a novel generative approach that is able to model the movement of the basketball players in the court. Based on the VAE and CVAE frameworks, we trained three models on the freely available NBA SportVU tracking data and used pictorial representation to capture the trajectories of the players and the ball. We illustrated that in all the tested cases, the proposed architectures was able to synthesized trajectories that resembles the original movement of the players. In the case of the conditional VAE, we showed how the generated trajectories defined patterns in terms of the given player. When we also conditioned on the trajectory of the ball, we were able to see how the model implicitly learned to pay attention to it.

References

[1] S. SportVU, "Basketball player tracking," *https://www.stats.com/sportvu-basketball/*, 2017.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[3] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pp. 6645–6649, IEEE, 2013.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[5] R. Shah and R. Romijnders, "Applying deep learning to basketball trajectories," *CoRR*, vol. abs/1608.03793, 2016.

[6] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, pp. 3104–3112, 2014.

[7] M. Harmon, P. Lucey, and D. Klabjan, "Predicting shot making in basketball learnt from adversarial multiagent trajectories,"

[8] K.-C. Wang and R. Zemel, "classifying nba offensive plays using neural networks," in *Proc. MIT SLOAN Sports Analytics Conf*, 2016.

[9] G. E. Hinton and Z. Ghahramani, "Generative models for discovering sparse distributed representations," *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, vol. 352, no. 1358, pp. 1177–1190, 1997.

[10] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[12] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," in *Advances in Neural Information Processing Systems*, pp. 3581–3589, 2014.

[13] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Advances in Neural Information Processing Systems*, pp. 3483–3491, 2015.

Supplemental Material
Unsupervised modeling of the movement of basketball players using a Deep Generative Model

David Acuna
University of Toronto
davidj@cs.toronto.edu

Abstract

In this supplementary material we include a larger set of additional experiments and visualizations. We start by showing synthesized offensive movement of random players. Then, we provide visualization of the movement of a particular players. Finally, we show a greater variety of generated offensive plays, this time the images are conditioning on both the player and the trajectory of the ball.

1 Visualization

Figure 1 illustrates synthesized offensive plays. Note that in this case , the images don not correspond to any player in particular. These images were obtained by sampling from the vanilla VAE trained on the pictorial representation of players movement. Figures 2 - 6, on the other hand, show synthesized images that resemble the movement of a particular player during an offensive event. In the first case (figure 2) the sampled pictorial representations are also conditioned on the trajectories of the ball. These conditional representations were obtained by sampling from the CVAE.

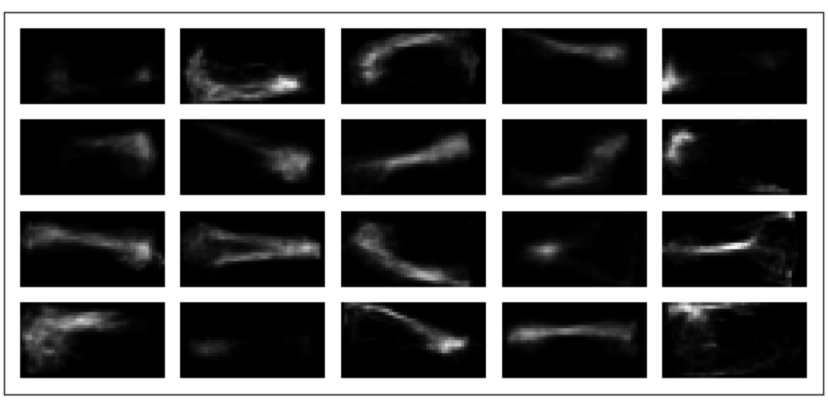

Figure 1: Synthesized offensive movement of players. The temporal aspects of the movement encoded as 'fading'.

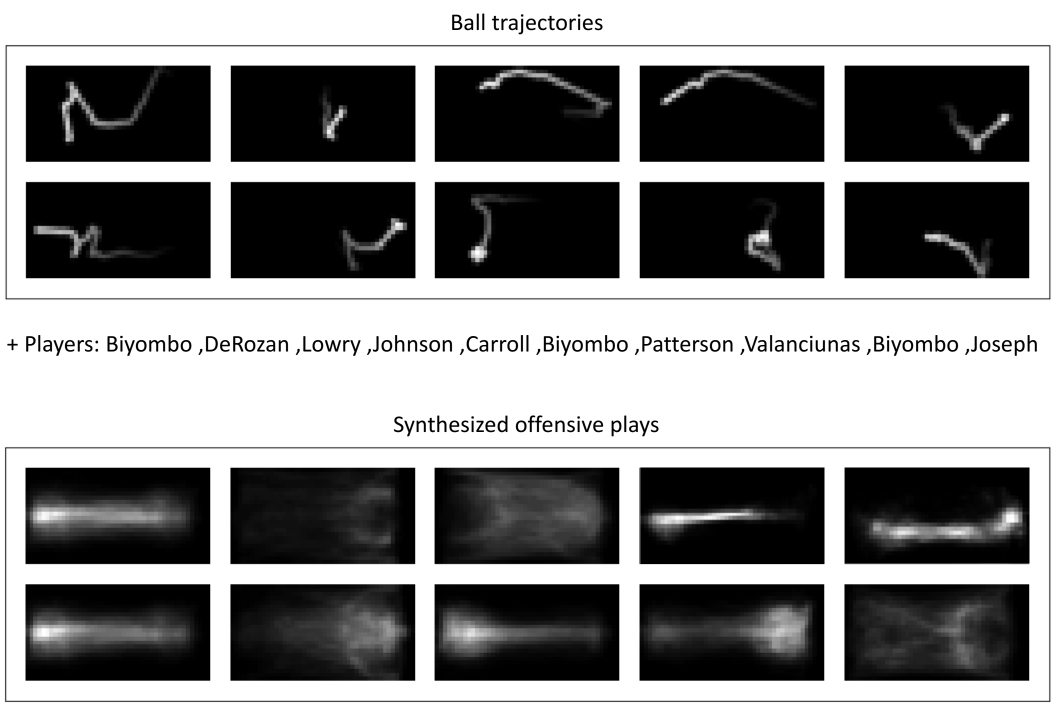

Figure 2: Synthesized offensive movement of the given players conditioning on the trajectories of the ball

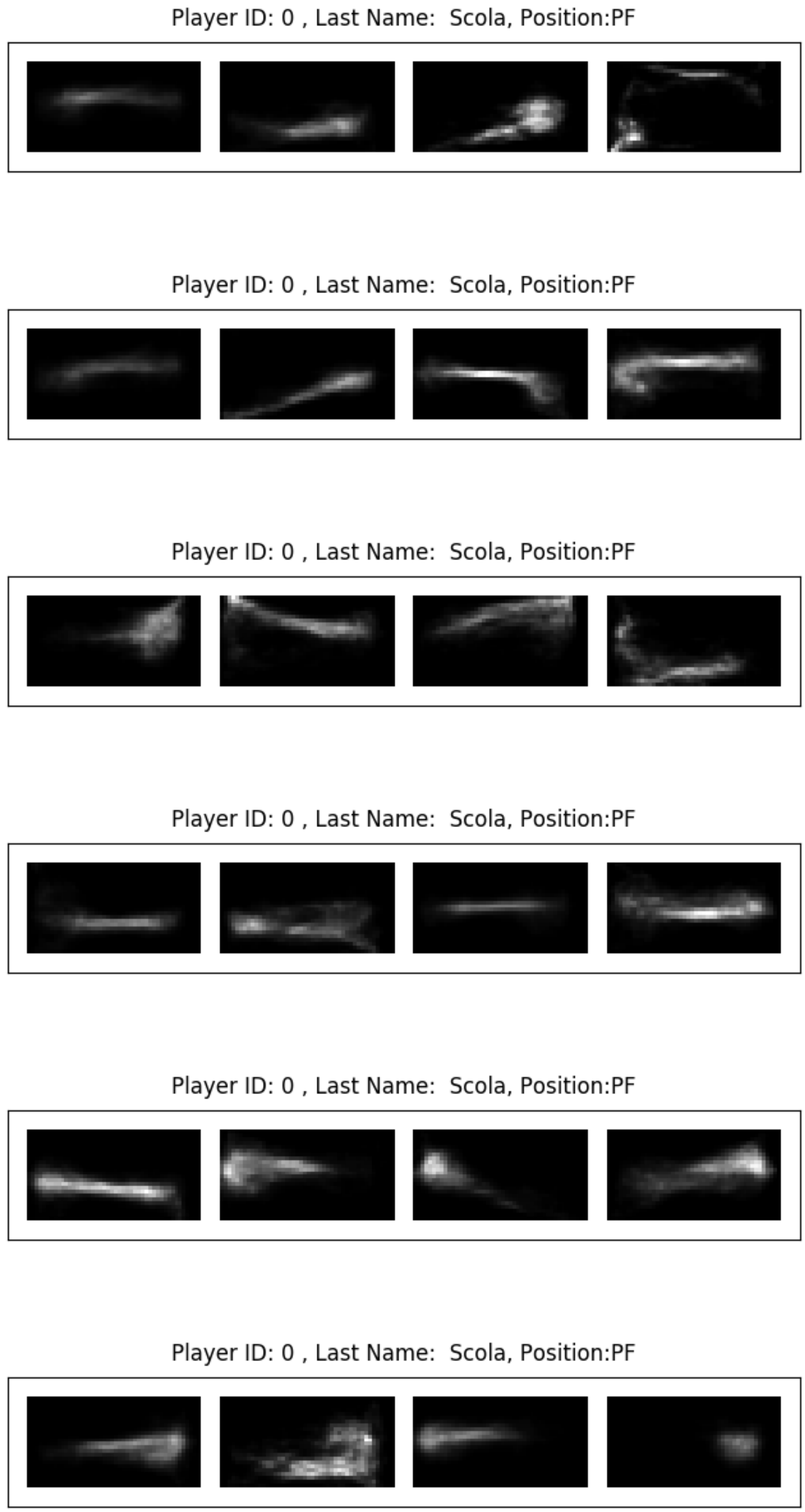

Figure 3: Synthesized movement conditioning on the player “Scola”.Position: PF

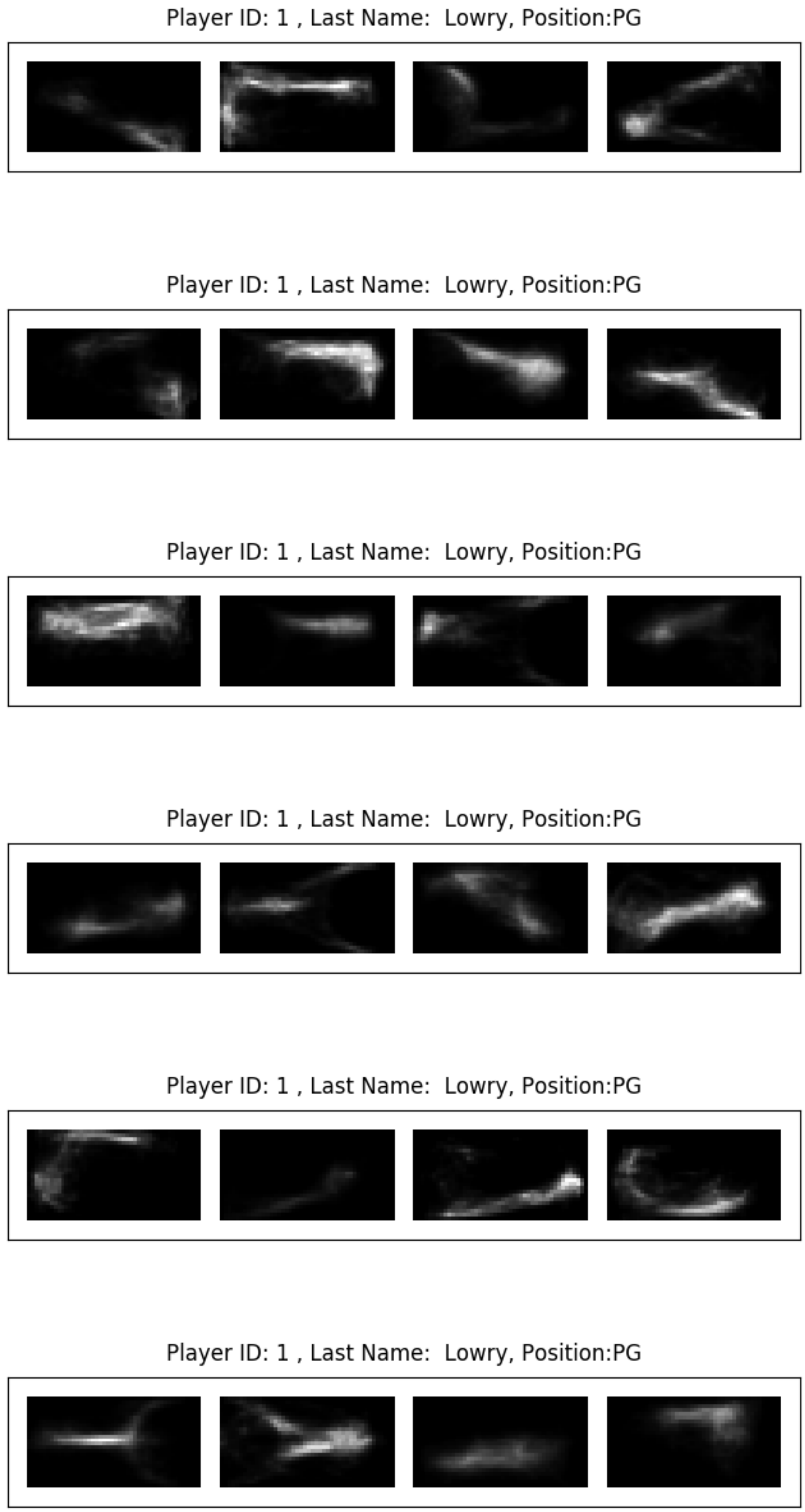

Figure 4: Synthesized movement conditioning on the player “Lowry”.Position: PG

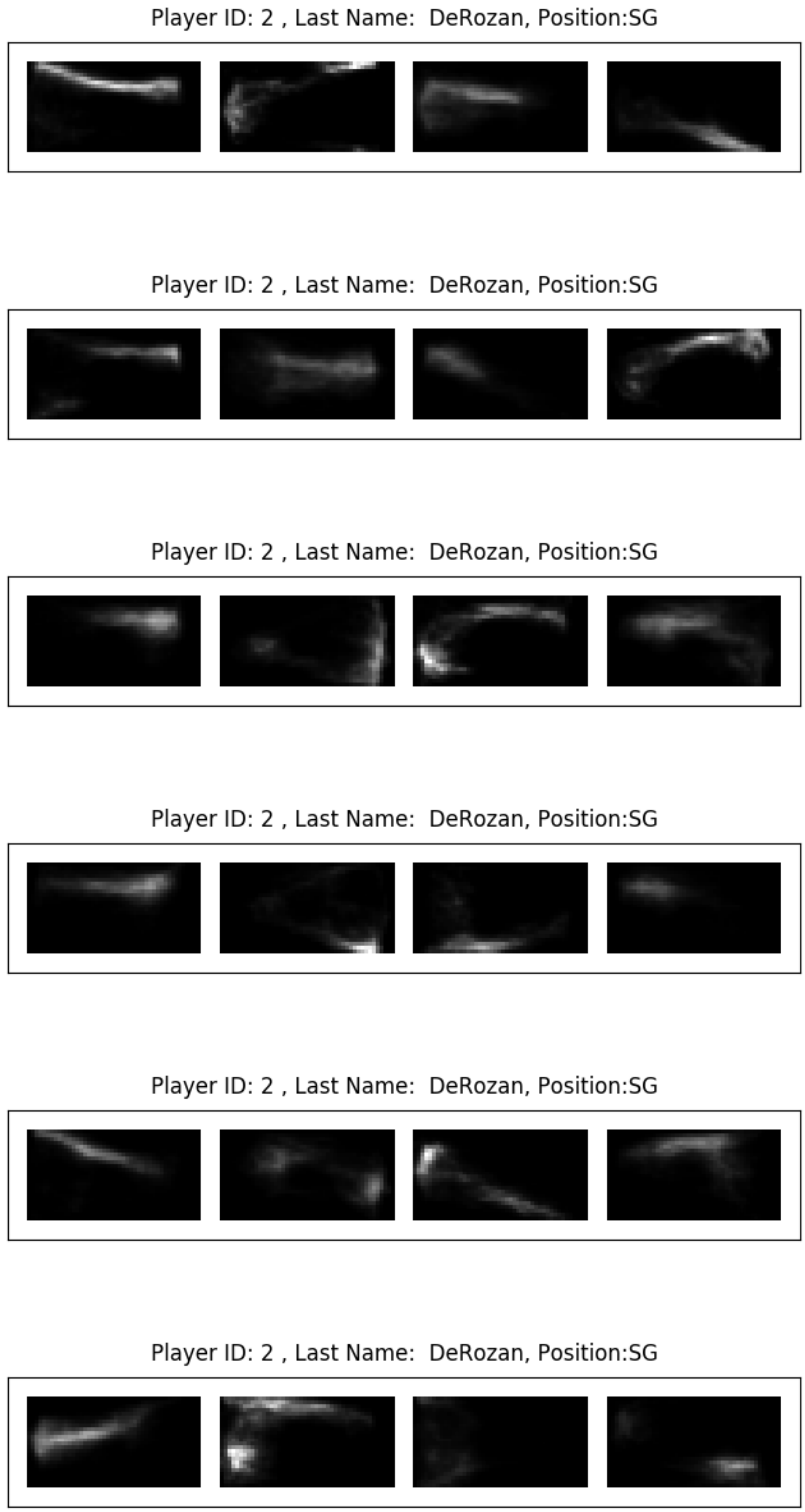

Figure 5: Synthesized movement conditioning on the player “DeRozan”.Position: SG

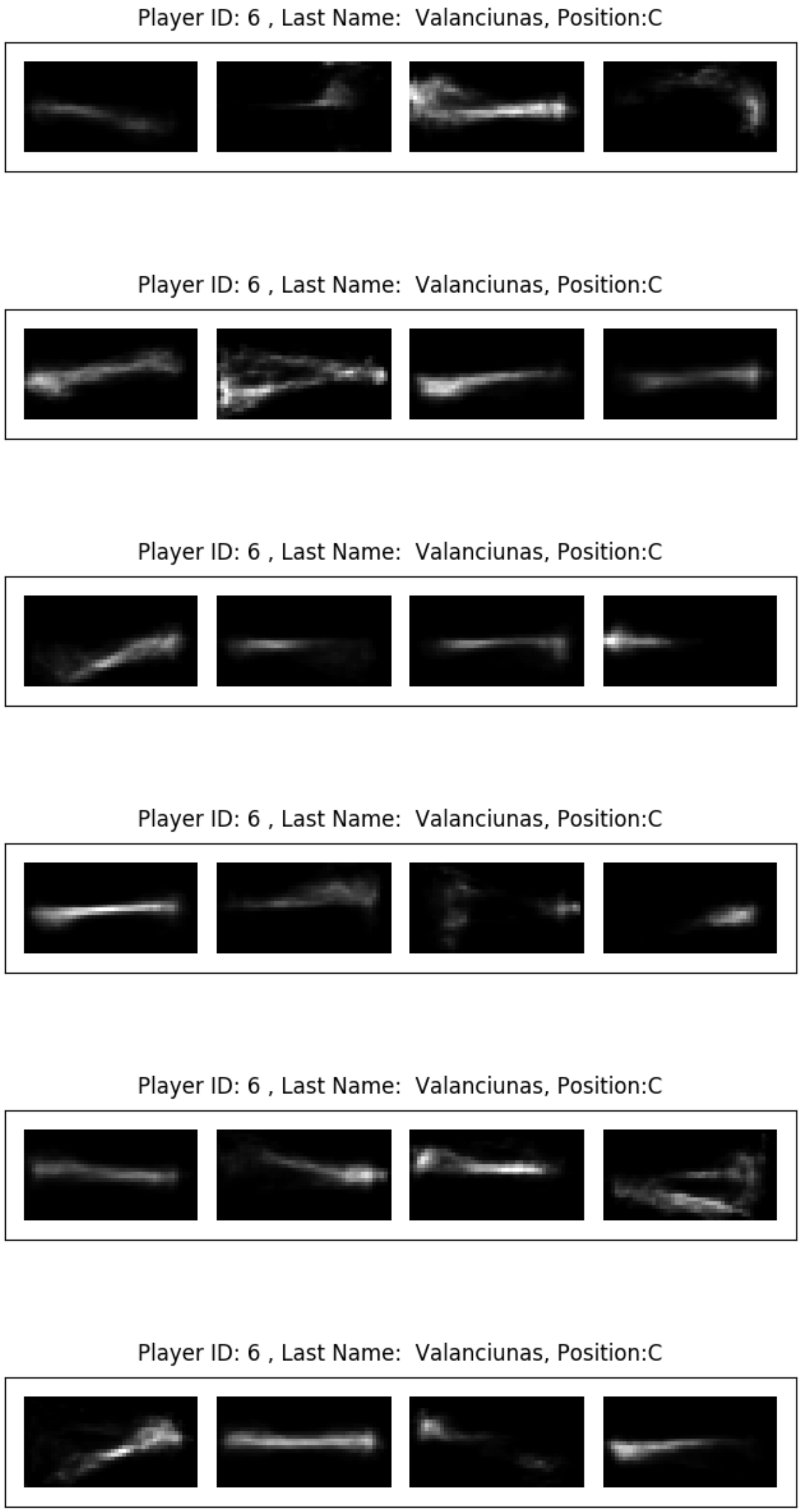

Figure 6: Synthesized movement conditioning on the player “Valanciunas”.Position: C