Dynamic NeRFs for Soccer Scenes

Sacha Lewin
University of Liège
Liège, Belgium
sacha.lewin@student.uliege.be

Maxime Vandegar
EVS Broadcast Equipment
Liège, Belgium
m.vandegar@evs.com

Thomas Hoyoux
EVS Broadcast Equipment
Liège, Belgium
t.hoyoux@evs.com

Olivier Barnich
EVS Broadcast Equipment
Liège, Belgium
o.barnich@evs.com

Gilles Louppe
University of Liège
Liège, Belgium
g.louppe@uliege.be

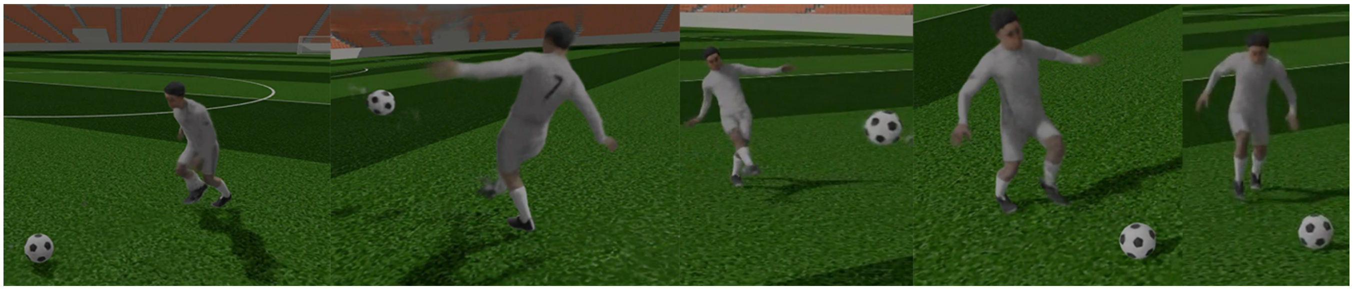

Figure 1: Novel view synthesis in a synthetic dynamic environment, given 30 known views and camera poses.

ABSTRACT

The long-standing problem of novel view synthesis has many applications, notably in sports broadcasting. Photorealistic novel view synthesis of soccer actions, in particular, is of enormous interest to the broadcast industry. Yet only a few industrial solutions have been proposed, and even fewer that achieve near-broadcast quality of the synthetic replays. Except for their setup of multiple static cameras around the playfield, the best proprietary systems disclose close to no information about their inner workings. Leveraging multiple static cameras for such a task indeed presents a challenge rarely tackled in the literature, for a lack of public datasets: the reconstruction of a large-scale, mostly static environment, with small, fast-moving elements. Recently, the emergence of neural radiance fields has induced stunning progress in many novel view synthesis applications, leveraging deep learning principles to produce photorealistic results in the most challenging settings. In this work, we investigate the feasibility of basing a solution to the task on *dynamic NeRFs*, i.e., neural models purposed to reconstruct general dynamic content. We compose synthetic soccer environments and conduct multiple experiments using them, identifying key components that help reconstruct soccer scenes with dynamic NeRFs. We show that, although this approach cannot fully meet the quality requirements for the target application, it suggests promising avenues toward a cost-efficient, automatic solution. We also make our work dataset and code publicly available, with the goal to encourage further efforts from the research community on the task of novel view synthesis for dynamic soccer scenes. For code, data, and video results, please see https://soccernerfs.isach.be.

CCS CONCEPTS

• **Computing methodologies → Computer vision representations**.

KEYWORDS

3D reconstruction, scene representation, dynamic, neural radiance fields, sports, soccer

ACM Reference Format:
Sacha Lewin, Maxime Vandegar, Thomas Hoyoux, Olivier Barnich, and Gilles Louppe. 2023. Dynamic NeRFs for Soccer Scenes. In *Proceedings of the 6th International Workshop on Multimedia Content Analysis in Sports (MMSports '23), October 29, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3606038.3616158

1 INTRODUCTION

Synthesizing novel views of a scene from a sparse sample of images is a long-standing problem in computer vision [7, 18, 31]. A notable field of application is sports broadcasting, in which action replays have a major role in story-telling and performance analysis. As one of the most popular sports, soccer receives a lot of broadcast coverage from top to low-tier competitions all over the world, with much care given to making the viewer experience ever more pleasant and engaging. Augmenting the broadcast production of soccer events with novel-view video synthesis of action replays is

MMSports '23, October 29, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0269-3/23/10…$15.00
https://doi.org/10.1145/3606038.3616158

therefore very attractive to industrial actors, and a real opportunity for the computer vision research community.

Despite the industry interest in novel view synthesis of soccer replays, only a few proprietary systems exist on the market. Indeed, such interest cannot outweigh the need for the highest image quality in broadcast productions; the industry, therefore, imposes very high standards in terms of the photorealism of the synthesized views. One noteworthy system [9] is able to deliver synthetic replays that are stunningly photorealistic, but for a few visual artifacts. Their setup is composed of dozens of very high-resolution static cameras, installed all around the soccer field high up above the bleachers. Their image data are processed by private, proprietary software running on very powerful hardware. These image data remain private as well, and equivalent public datasets are simply nonexistent. The only insight offered by this system to the research community is the validity of using a static multi-camera setup for the task.

Even with no image data available, one can reason about the challenges that arise from using an array of distant static cameras as a basis for the reconstruction of a soccer environment. Outdoor sports like soccer are composed of a large static environment, the stadium, and small dynamic elements, the players and the ball. Traditional computer vision methods would most likely have to rely on very high-resolution images, as in [9], to reconstruct an underlying 3D model of the scene able to faithfully render the movements of the small dynamic elements. Having to deal with massive amounts of image data for reconstructing a single, short soccer action is however not a desirable property for a solution.

Building on the modern deep learning-based paradigm to computer vision problems, neural radiance fields [24] (NeRFs) have recently become the state of the art for high-quality novel view synthesis, and have been widely improved and extended to produce excellent results in very challenging settings. A notable line of work is *dynamic* NeRFs, i.e. neural models purposed to reconstruct spatiotemporal content, as opposed to only spatial, static content. This, therefore, begs the question: *Are dynamic NeRFs suitable for reconstructing soccer scenes?* To find potential answers to this question, we propose this exploratory work, in which we make three important assumptions.

First, we only consider camera setups similar to the one used by the aforementioned proprietary system [9], deeming it optimal for the task at hand. Specifically, we use an array of 20 to 30 static cameras, positioned all around the soccer stadium and pointing toward the soccer field. This assumption goes well with the working conditions usually recommended to achieve good performance with NeRFs. Moreover, most NeRFs assume input views to be calibrated by third-party Structure from Motion (SfM) tools, which are known to bring robust results with such camera setups in mostly static environments, such as a soccer stadium.

Second, we limit our study to synthetic soccer datasets, yet we believe its results also apply to real data. As already mentioned, soccer image datasets with the considered camera setups are virtually nonexistent for the public, to the best of our knowledge. We therefore composed synthetic datasets, using public computer graphics engines and models. Because we control the cameras in our 3D virtual environments, this assumption also allows us to leave camera calibration aspects out of the scope of our work. We are confident that our findings remain valid when working on real use cases, given the availability of robust SfM tools, and the reputation of very good photorealism of NeRFs with real image data.

Third, we only consider *general* dynamic NeRFs, i.e., dynamic NeRFs with *no domain knowledge*, to identify early limitations of the neural-based reconstruction paradigm in the context of our task. Another important reason is that domain-specific priors are often difficult and expensive to produce. For instance, an accurate skeletal reconstruction of the players would be predictably very useful for soccer replay synthesis, but is a hard task in itself, especially with the considered camera setups. Our goal is to avoid resorting to such priors, which are likely to be complex and costly. This assumption also has the advantage to make our study potentially insightful for the use of dynamic NeRFs for other sports than soccer, given similar camera setups.

We select recent state-of-the-art general dynamic NeRF models and compare them in three synthetic soccer environments of increasing complexity. Our aim is to progressively transition from ideal conditions for the considered models, to conditions that are similar to the optimal camera setup used in [9].

Our contributions could be summarized as follows:

(1) We provide a study of the performance of general dynamic NeRFs on the task of soccer replay synthesis in increasingly complex environments. Models are studied as they were introduced in the literature, then augmented with general, non-domain specific components that we identify. We close the study with a higher-level discussion about limitations and future work.

(2) As we wish to foster research efforts toward solving this challenging task, we publicly release our code, including the improving components and experimental settings, and our complete work dataset, including images, depth maps, Blender [8] scripts, and camera calibrations for all synthetic environments. These are all ready-to-use in Nerfstudio [35], a rich and popular open-source framework for using and developing NeRF models.

The remainder of this paper is organized as follows. Section 2 provides preliminaries about NeRFs and introduces their extension to dynamic environments. Section 3 details our experimental setup: methods, evaluation, and environments. Section 4 showcases and discusses results. Finally, Section 5 provides a higher-level discussion about the feasibility of using these methods, along with some paths for improvement.

2 NEURAL SCENE REPRESENTATION

Neural Radiance Fields [24]. The original neural radiance field (NeRF) model implicitly encodes a scene in the weights of a multilayer perceptron (MLP). The model learns to associate density and color information to any point in space, which allows for rendering images using classical volume rendering [15, 23]. This process is end-to-end differentiable, which allows for training using only captured views and their associated camera poses. For improving training and rendering time, more recent methods [6, 26, 34] use a hybrid approach, leveraging both implicit and explicit representations, such as voxel grids. Those methods store learnable features, which are then decoded with an MLP.

Dynamic NeRFs. Various techniques have been proposed to extend NeRFs to dynamic reconstruction. Methods such as [11, 27, 30] learn a separate field, known as a deformation network, that maps each point to its corresponding position in a canonical scene. Other methods input time to the radiance field. While direct conditioning on time provides poor results [30], indirect conditioning [1, 12, 13, 19, 33, 37] obtains state-of-the-art results on various popular benchmarks. Some models leverage domain knowledge, such as Human NeRFs. They often work by learning the motion of a skinned multi-person linear model (SMPL [22]) along with its appearance [28, 29, 42]. A more recent method supports human-object interactions [21]. These specific models still require complex and controlled setups. Two non domain-specific models, K-planes and NeRFPlayer, are of particular interest to us, based on their state-of-the-art performance on diverse benchmarks, and the approach they take to the reconstruction of dynamic content.

K-Planes [13]. This model builds upon methods [6, 32] that factorize the 4D space into 6 planes, corresponding to each pair of coordinates. This approach, and concurrent work [5, 36], offer greatly-improved efficiency with high-quality results. The planes store feature vectors uniformly in space and time, at increasing scales, similar to the multiresolution hash encoding used in [26]. The feature vectors associated with a given point in space and time are then decoded by a shallow MLP into a density and an RGB color. K-Planes reaches state-of-the-art performance on various datasets.

NeRFPlayer [33]. This method introduces two main contributions: (i) a dynamic version of traditional explicit feature storage, such as the hash encoding from [26], by using a sliding window over a larger fixed-size feature vector, and (ii) a scene decomposition into different areas depending on their nature: static, deformed, or new. Each area is modeled with a different approach, which is mostly beneficial to monocular setups. On common dynamic multi-view datasets [19], NeRFPlayer reaches high-quality results, similar to K-Planes.

3 IMPLEMENTATION

The selected methods are K-Planes [13] and NeRFPlayer [33], outlined in Section 2. These versions are implemented in Nerfstudio [35], an open-source framework that we use for all our experiments. For fair comparisons, shared settings between the models are identical, such as proposal sampling and scene contraction [2]. Model-specific hyperparameters follow the original implementations, except for the model size. We increase the hash map size of NeRFPlayer to 2^{20} with a temporal dimension of 64 and use Nerfacto [35] as the backbone. We also drop the decomposition from NeRFPlayer, which mainly benefits monocular setups and results in unnecessarily large models. We add two additional scales to K-Planes, resulting in multiscale resolutions from 2^6 to 2^{11}. When enabled, ray importance sampling based on global medians (ISG) is employed [19]. Training follows typical Nerfstudio settings: models are trained for 30,000 iterations using Adam [17] with a learning rate of 10^{-2}, which takes about 1 to 2 hours on an NVIDIA RTX 3090 GPU for each scene. Unlike typical methods which train using downsampled images for faster training, we observe improvements when using full-resolution 1080p images in our environments, without large increases in training time.

We make both our code and datasets publicly available. The former includes slightly modified versions of K-Planes and NeRFPlayer, more convenient data management for dynamic environments, training settings, and other components mentioned in Section 4, such as ray importance sampling and dedicated metrics. The latter include training images, calibrated poses, depth maps, Blender files, and data parsers to readily conduct experiments within Nerfstudio.

3.1 Evaluation

Three metrics are typically used for assessing the visual quality of novel view synthesis: (i) PSNR, which computes differences at the pixel level, (ii) SSIM [38], which takes structural changes into account, and (iii) LPIPS [41], based on features in deep convolutional networks which better correlate with human judgment. Quantitative evaluation is known to be a difficult task in novel view synthesis applications and, sometimes, to hardly reflect visual quality accurately. Environments like ours make it even more challenging. Indeed, the dynamic content of interest is the players and the ball, which occupy a small region of the images. As the metrics are computed over the whole image, they are barely affected by the reconstruction quality of small elements of interest. Furthermore, we consider dynamic scenes, and computing per-frame metrics conveys no information about the temporal consistency of the results.

While the first issue can be tackled in synthetic environments by including additional views close to the content of interest, it is often not possible in real conditions. To address this, we propose alternative versions of these three metrics which are computed in restricted bounding boxes around the dynamic content. The boxes can be automatically generated by simply using an object detection model such as RetinaNet [20]. We refer to them as *focused metrics*.

To illustrate them, we compare *default* and *focused* metrics computed between one ground-truth evaluation image and novel views generated by four different versions of K-Planes, (a) to (d), where we vary the depth and width of the MLP decoder, which causes differences in prediction quality. The predicted images and associated PSNRs are depicted in Fig. 2. The other metrics, i.e., SSIM and LPIPS, are reported in Tab. 1. The predictions highlight the necessity for alternative evaluation methods. In (a, b, c), the bleachers are poorly reconstructed, which strongly affects all metrics, as only (d) obtains a good score. However, the player is only missing in (a), which is better reflected by the focused metrics. Also, artifacts are present around the player in (b), which underlines the need for not restricting the bounding box right around the player.

Despite these improvements, the new metrics still convey no temporal information. Furthermore, they fail if the players or the ball are not detected. For those reasons, qualitative evaluation is always preferred. With a focus on assessing the reconstruction quality of the player, we render novel viewpoints along camera paths closer to the player than the distant views used for training. In our synthetic scenes, we include additional close-up views for quantitative evaluation, which are useful when the focused metrics fail, such as in the *Players* environment, in Section 4.3. Otherwise, one camera is excluded from the training set and used for evaluation only.

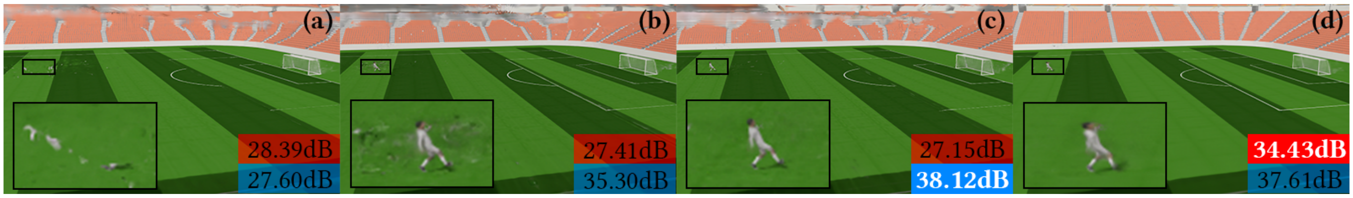

Figure 2: Illustration of *focused* metrics. Each image is a prediction from the same evaluation camera pose using different model settings. The black box represents the window in which the focused metrics are computed. When using the default metrics (shown in red), only the fourth model achieves a high score, primarily due to its well-reconstructed bleachers. With the focused metrics (shown in blue), only the first model receives a low score as it fails to accurately reconstruct the dynamic content of interest.

Table 1: Comparing default and focused metrics for the novel views shown in Fig. 2. The default metrics are best on scenes where static elements are better reconstructed, while the dynamic-focused metrics better reflect the quality of the region of interest. Best results in bold, second-best underlined.

	Default			Focused		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
(a)	<u>28.39</u>	0.724	0.240	27.60	0.735	0.254
(b)	27.41	<u>0.745</u>	<u>0.218</u>	35.30	0.818	0.079
(c)	27.15	0.737	0.231	**38.12**	<u>0.882</u>	<u>0.039</u>
(d)	**34.43**	**0.805**	**0.149**	<u>37.61</u>	**0.924**	**0.018**

3.2 Environments

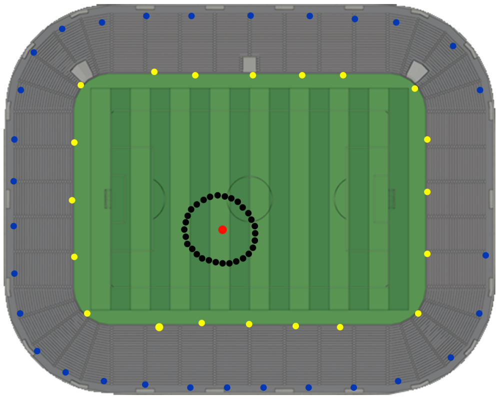

Figure 3: Illustration depicting the camera placements for each setup. The player position is highlighted in red for the *Single Player* setups. The 30 close-up cameras are represented in black, the 20 broadcast-style cameras are shown in yellow, and the 30 stadium-wide angles are denoted in blue. Examples of associated training views can be observed in Fig. 4, 5.

To the best of our knowledge, most sports datasets are limited to a few synchronized cameras, and there are no public datasets that include dozens of synchronized and calibrated views. For example, the KTH Multiview Football Dataset II [16] only contains three synchronized cameras, that often focus on a single player.

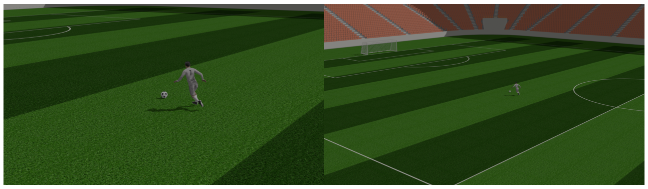

Figure 4: Camera setups for the *"Single Player"* scene. Left: close-up cameras are placed around the players, similar to common datasets for novel view synthesis. Right: cameras are placed around the field and oriented toward the player, typical of broadcast conditions.

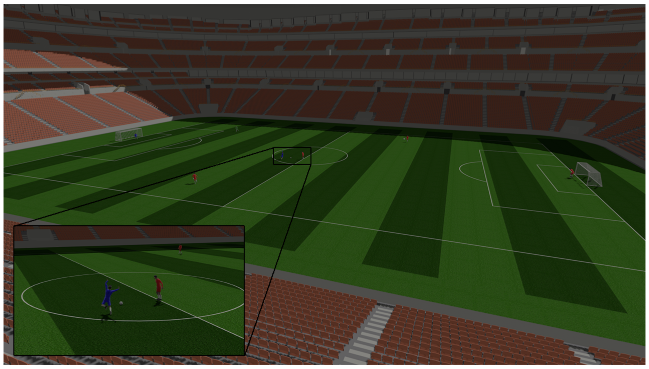

Figure 5: Example training view for the *"Players"* scene, along with a close-up view dedicated for evaluation (bottom left). The players occupy a very small part of the images.

In this exploratory work, we build synthetic environments to circumvent the lack of real data. The scenes are of increasing complexity, starting from relatively close-up views, commonly used with NeRFs, then using more distant cameras giving a field of view similar to what is used in broadcast coverage, and finally considering even more distant cameras placed in the bleachers covering the whole field with more players. The different setups are illustrated in Fig. 3. This allows us to progressively tackle the challenges that occur with soccer environments, mainly the reconstruction of small dynamic content. All cameras are static and the environments are built using Blender [8] with player models from Adobe Mixamo [14], and a stadium model available under a free CC0 license [25].

Single Player: Close-up Views. This synthetic environment features a single player placed at the center of the field, shooting a ball. This first camera setup is composed of 30 close-up views around the player and resembles typical conditions of benchmarks like DyNeRF [19]. An example training image is depicted in Fig. 4 (left).

Single Player: Broadcast-style Views. Within the same environment, we consider a second camera configuration that features 20 views placed around the field, whose field of view is close to broadcast conditions. The player represents only a tiny portion of the images. An example training image is depicted in Fig. 4 (right).

Players: Stadium-wide Views. This more complex environment features several players and balls interacting all over the field, captured by 30 wide-angle cameras placed high up in the bleachers and are thus much more distant from the field. Six additional cameras, used exclusively for evaluation, are placed near the players for more meaningful results. In this setup, training views cover the whole field at all times but cover very few details about the players and balls due to their large distance. An example training view is depicted in Fig. 5.

4 EXPERIMENTS

In this section, we assess the performance of K-Planes [13] and NeRFPlayer [33] in increasingly complex environments, each described in Section 3.

4.1 Single Player: Close-up Views

As a first attempt, we run the original models from the initial papers in similar conditions to traditional datasets [4, 19]. The player occupies a large region of the training images, is captured by 30 cameras, and performs smooth motion. In these settings, the models are able to reconstruct the stadium flawlessly. The player's motion is reconstructed, but the texture is blurry, even when using larger models. The ball is not reconstructed when moving fast in the air and disappears.

We can circumvent these issues by employing an improved pixel sampling strategy. Traditionally, training rays are traced by uniformly sampling pixels although dynamic content, especially if small, should be sampled more often. In [19], several improved strategies are described, known as *Ray Importance Sampling* (IS). This new sampling strategy, which prioritizes sampling dynamic content pixels, is particularly necessary for setups like ours, considering the scale of dynamic objects, even in this first more ideal environment. This general modification, which can be applied to both models, yields substantial improvements in quality and training time. Renderings are performed around the player with both models, with and without ray importance sampling, and are depicted in Fig. 6. Associated metrics, computed using a dedicated evaluation camera, are reported in Tab. 2. Visual details are recovered much quicker, and final results are drastically more detailed when using importance sampling. Overall, results are similar between NeRFPlayer and K-Planes when using similar model sizes, as depicted in Fig. 6. NeRFPlayer tends to recover slightly more details on the player but produces more artifacts around it. While it is not able to reconstruct the ball when it is in the air, K-Planes manages to reconstruct it, although ghosting effects appear. When not using importance sampling, the ball is never reconstructed. Here, the use of focused metrics barely affects our interpretation of the results, due to the player's scale in the images, which causes the bounding boxes to cover a large part of the view. While the focused PSNR improves when using importance sampling, the other metrics sometimes degrade, which does not support qualitative results from Fig. 6. This may be explained by the fact that IS helps to partially reconstruct the ball, which introduces artifacts.

4.2 Single Player: Broadcast-style Views

While the models perform well with close-up cameras, such views are usually not available in practical applications. Here, we experiment with the same scene but observed by more distant training views, which are positioned like broadcast cameras, all around the field.

Example renderings, using ray importance sampling, are depicted in Fig. 7. The player is still reconstructed accurately, although less detailed, compared to the closer camera setup. In these new conditions, importance sampling is even more necessary, as the player is barely reconstructed without it. However, even with IS, the ball is not reconstructed when in motion. Instead, artifacts appear everywhere in the direction of cameras.

4.3 Players: Stadium-wide Views

This final synthetic environment moves further away from the center of the scene and features 30 wide-angle cameras, located high in the bleachers, that cover the whole field. Many players are present on the soccer pitch, interacting with each other and with balls. This setting is particularly challenging, due to the very small visibility of players in the training images.

Results are depicted in Fig. 8 and 9. Even in this very challenging configuration, the players are reconstructed and we can distinguish their motion. However, even with larger models, 1080p training images, and ray importance sampling, the results remain blurry. The ball is barely reconstructed when moving slowly, and not at all when moving fast (e.g., when being shot). Such camera setups, therefore, seem to be limited for detailed results, at least when using no domain knowledge.

5 DISCUSSION

In this exploratory work, we compared recent state-of-the-art dynamic NeRF models, i.e., K-Planes [13] and NeRFPlayer [33], in increasingly complex soccer environments, to assess their readiness for broadcast-quality novel-view video synthesis of soccer replays. In the ideal NeRF setup, where close-up cameras capture detailed views of the target moving objects, the models reached great reconstruction quality. However, when using distant views in a camera setup similar to the best-result proprietary system [9], the results offered by general dynamic NeRFs drastically degrade. In such distant camera setups, we showed that incorporating additional components to the original models, like ray importance sampling [19], becomes an absolute necessity.

We tried to avoid working with very high-resolution images, as opposed to [9], limiting our input image data to 1080p. Indeed,

Table 2: Quantitative results for both models with and without ray importance sampling (see Fig. 6). Due to using closer cameras, the focused metrics have a limited impact on the results.

		Default			Focused		
		PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Without Importance Sampling	K-Planes	**32.84**	**0.786**	**0.167**	34.41	**0.816**	**0.126**
	NeRFPlayer	31.54	0.754	0.211	34.55	0.807	0.181
With Importance Sampling	K-Planes	32.53	0.751	0.199	**35.06**	0.788	0.156
	NeRFPlayer	31.26	0.721	0.225	34.78	0.781	0.191

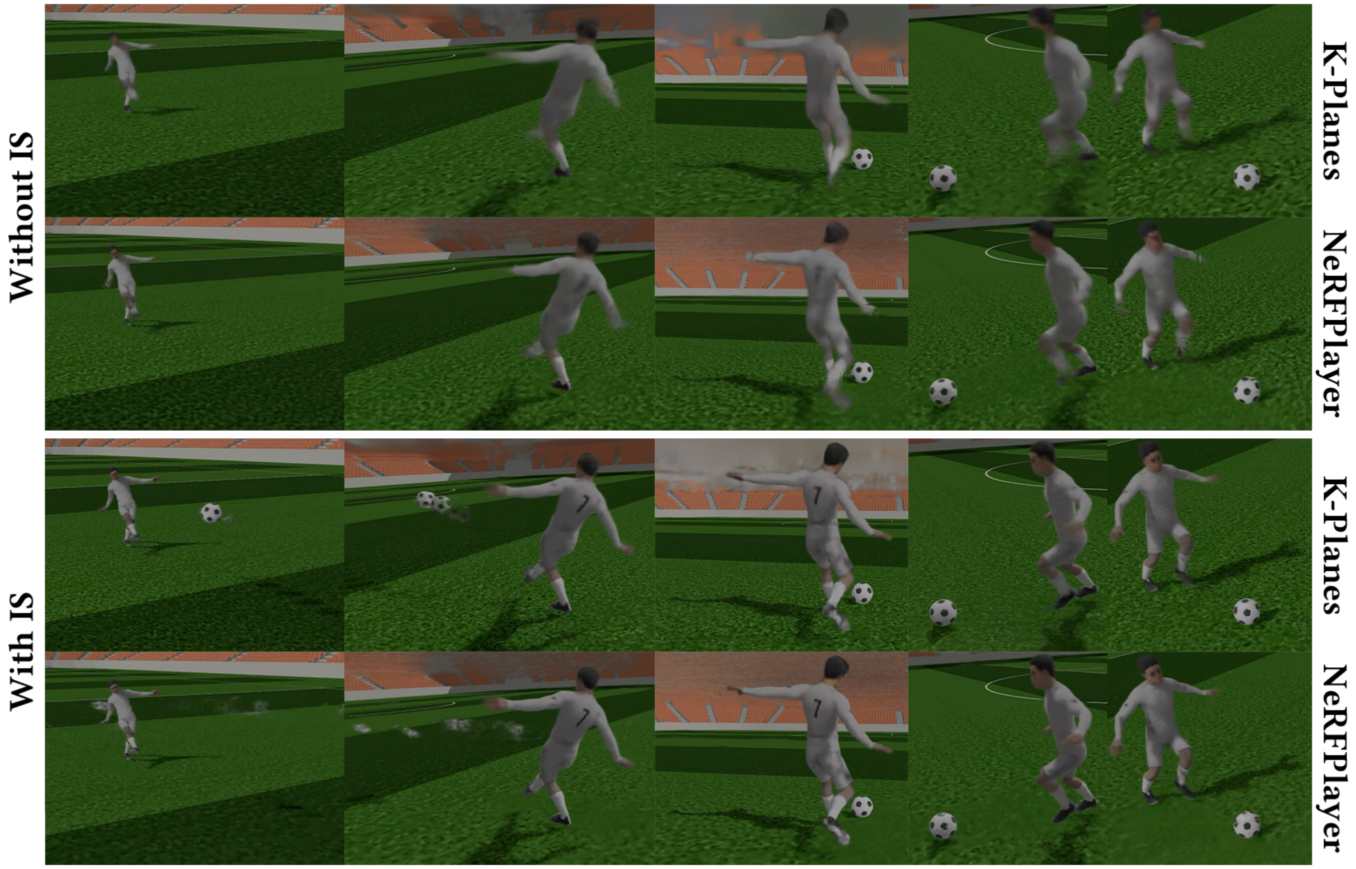

Figure 6: Comparative results between K-Planes and NeRFPlayer on the *Single Player* environment with *close-up cameras*, with and without ray importance sampling (*IS*). Overall, both models obtain similar results. NeRFPlayer tends to recover slightly more details on the player, possibly due to the factorization of K-Planes, but induces more artifacts and does not manage to reconstruct the ball when in the air. Importance sampling drastically improves results for both models and allows K-Planes to recover the ball when in the air.

even though increasing the image resolution is an obvious path of improvement toward capturing fine details, the massive amounts of data thus generated are close to being prohibitive computationally, and we wanted to explore more economical solutions. For similar reasons, we avoided resorting to domain-specific priors in this study, as such priors can be arduous and costly to produce, e.g., an accurate skeletal reconstruction of the players. Assuming such restrictions, and despite our improving components, we must conclude that general dynamic NeRF models may fall short of meeting the high-quality requirements of the broadcast industry for novel view synthesis of soccer replays.

Although it was not the focus of our work, another inconvenience of using dynamic NeRFs in broadcast applications might be their time performance. Indeed, the models we selected require one hour to train on thirty 4-second clips and 5 minutes to render a

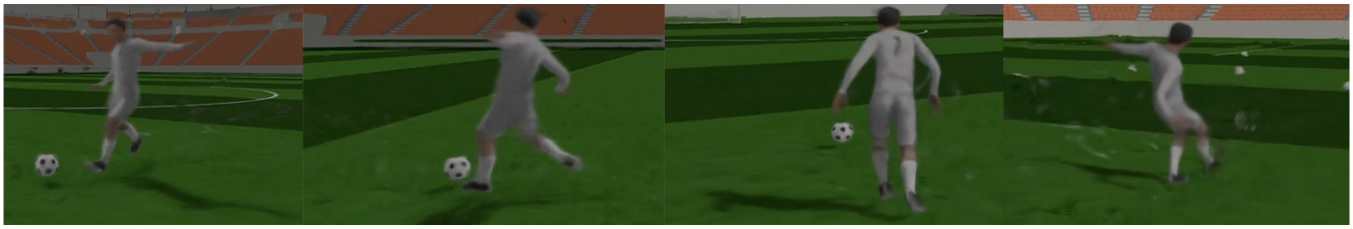

Figure 7: Novel views synthesis in the *Single Player* environment using a camera setup similar to broadcast conditions. Results are obtained from K-Planes with Ray Importance Sampling. The player is well reconstructed, but its texture is blurrier when compared to using closer cameras. Despite the utilization of Importance Sampling, the model fails to accurately reconstruct the ball in motion (on the right).

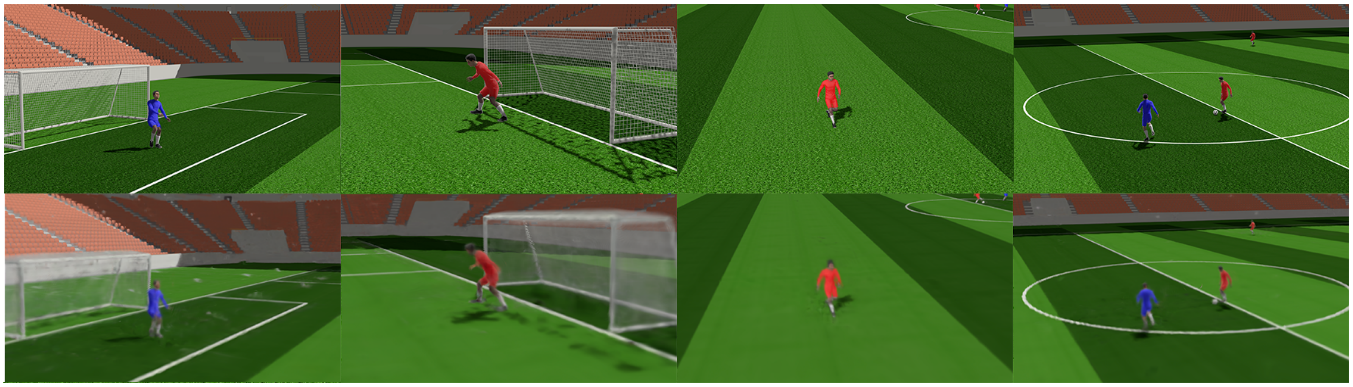

Figure 8: Ground truths (top) and predictions (bottom) for the *Players* environment from close-up views dedicated to evaluation. In these difficult conditions, the players are still reconstructed, although quite blurry.

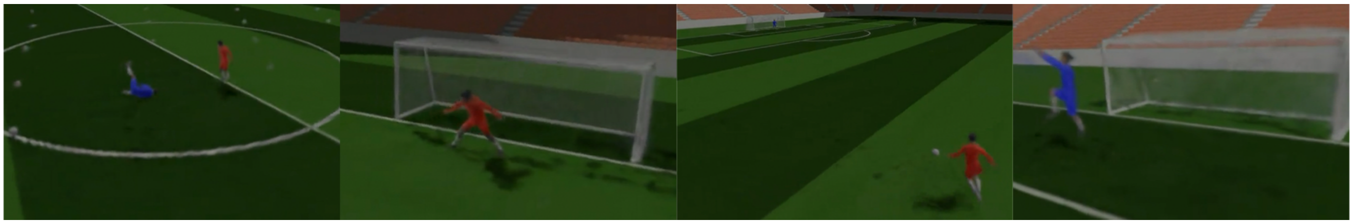

Figure 9: Additional novel views synthesis results in the *Players* environment. In these challenging conditions, the motion and position of the players are correctly reconstructed, albeit significantly blurry. The ball is not reconstructed and causes artifacts visible on the whole field (leftmost image).

10-second video (about 1FPS for 1080p rendering). However, we believe that training times could very certainly be lowered, notably by pre-training a model for the empty stadium. Nonetheless, even the most recent models [37] require more than 15 minutes of training, which while being unsuitable for live replay, might fit post-match applications.

Still, we believe that dynamic NeRFs could play an important role as the core part of a fully satisfying solution. Following the same line of work as what was done in our study, a first path of improvement would be to try incorporating other general components into dynamic NeRFs. The visibility loss from Nerfbusters [39], the improved proposal sampling from Zip-NeRF [3], and the restorer from NeRFLiX [43] are promising components that would certainly be beneficial to a detailed reconstruction of soccer scenes in distant camera setups. Nevertheless, using such general improving components may still not be enough for the task.

Although using absolutely no domain knowledge is appealing, it may be necessary to use some domain-specific components to reach broadcast-quality results as well as a better time performance during training, more in line with broadcast time constraints. Yet, one should be cautious of the complexity and costs associated with bringing specific models within a solution. For instance, while showing impressive results in controlled working conditions, NeRFs that focus on human reconstruction [21, 40, 42] are not directly usable with a distant camera setup such as ours, and would require considerable adaptation to reconstruct humans in more diverse, less-constrained configurations, such as multiple humans at arbitrary positions.

As manifest as increasing the input image resolution, another path of improvement is to obtain and leverage more zoomed-in input views, together with the distant views given by our chosen camera setup inspired by [9]. Our study indeed showed that broadcast-style views may capture enough details to render novel views with near-acceptable quality for the target application. Such cameras could not be static, though, and it is unrealistic to suggest manning dozens of additional broadcast-style cameras with operators tasked to follow the action. This naturally leads to consider using the image data coming from the actual broadcast cameras, which are used to cover the soccer event. Including broadcast moving cameras within the reconstruction task would introduce new difficulties, such as motion blur, less accurate camera calibration, view sparsity for the zoomed-in region of interest, and the inadequacy of importance sampling as it relies on static cameras. The benefits could however outweigh the difficulties. First, robust SfM tools could still be used with satisfaction in a mixed setup of distant static cameras and broadcast moving cameras, to retrieve the calibration of the moving cameras at all times. Second, using such a mixed setup could allow using less static cameras than the dense 20-30 camera array considered in this study. A case could even be made that broadcast cameras become the main source of information in an economical solution, using all available NeRF extensions that deal with sparse camera setups, such as depth supervision [10] based on what SfM tools output for the scene structure, along with the camera calibrations.

An indirect, but very important path of improvement is the design of better evaluation metrics for dynamic NeRFs. Evaluating these models in less-frequently considered dynamic environments, such as soccer, poses significant challenges. In our study, we proposed a simple yet better method for computing evaluation metrics. However, much more could be made, particularly in detecting general moving content, incorporating temporal information, and finding ways to accurately reflect the challenging reconstruction quality of the ball. Proper evaluation of these models is *crucial* because, without accurate assessment, it is difficult to determine the readiness of a method for real-world applications.

Finally, also an indirect path of improvement: acknowledging and remedying the lack of public multi-view soccer datasets. Even a single image dataset of a dozen synchronized cameras capturing a few soccer actions would be of tremendous interest to the community. The synthetic environments we built are a modest proxy of such a dataset, that we publicly release along with all the code used for the experiments, both ready to use in Nerfstudio, an open-source framework for NeRF research. We strongly encourage building richer datasets, both by extending our scenes and by recording real data using enough synchronized and calibrated cameras.

ACKNOWLEDGMENTS

We sincerely thank EVS Broadcast Equipment for providing the necessary compute for the various conducted experiments. We also thank the Nerfstudio community for helpful insights.

REFERENCES

[1] Benjamin Attal, Jia-Bin Huang, Christian Richardt, Michael Zollhoefer, Johannes Kopf, Matthew O'Toole, and Changil Kim. 2023. HyperReel: High-fidelity 6-DoF video with ray-conditioned sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16610–16620.

[2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. 2022. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5470–5479.

[3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. 2023. Zip-NeRF: Anti-aliased grid-based neural radiance fields. *arXiv preprint arXiv:2304.06706* (2023).

[4] Michael Broxton, John Flynn, Ryan Overbeck, Daniel Erickson, Peter Hedman, Matthew Duvall, Jason Dourgarian, Jay Busch, Matt Whalen, and Paul Debevec. 2020. Immersive light field video with a layered mesh representation. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 86–1.

[5] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. 2022. Efficient geometry-aware 3D generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16123–16133.

[6] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. 2022. Tensorf: Tensorial radiance fields. In *European Conference on Computer Vision*. Springer, 333–350.

[7] Shenchang Eric Chen and Lance Williams. 1993. View interpolation for image synthesis. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. 279–288.

[8] Blender Online Community. 2018. *Blender - a 3D Modelling and Rendering Package*. Stichting Blender Foundation, Amsterdam. http://www.blender.org

[9] Intel Corportation. [n. d.]. *Intel ©True View*. https://www.intel.com/content/www/us/en/sports/technology/true-view.html

[10] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. 2022. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12882–12891.

[11] Yilun Du, Yinan Zhang, Hong-Xing Yu, Joshua B Tenenbaum, and Jiajun Wu. 2021. Neural radiance flow for 4d view synthesis and video processing. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society, 14304–14314.

[12] Jiemin Fang, Taoran Yi, Xinggang Wang, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Matthias Nießner, and Qi Tian. 2022. Fast dynamic radiance fields with time-aware neural voxels. In *SIGGRAPH Asia 2022 Conference Papers*. 1–9.

[13] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. 2023. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12479–12488.

[14] Adobe Inc. 2015. Mixamo. Retrieved April 16, 2023 from https://www.mixamo.com

[15] James T Kajiya and Brian P Von Herzen. 1984. Ray tracing volume densities. *ACM SIGGRAPH computer graphics* 18, 3 (1984), 165–174.

[16] Vahid Kazemi, Magnus Burenius, Hossein Azizpour, and Josephine Sullivan. 2013. Multi-view body part recognition with random forests. In *2013 24th British Machine Vision Conference, BMVC 2013; Bristol; United Kingdom; 9 September 2013 through 13 September 2013*. British Machine Vision Association.

[17] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014).

[18] Marc Levoy and Pat Hanrahan. 1996. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. 31–42.

[19] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. 2022. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5521–5531.

[20] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*. 2980–2988.

[21] Jia-Wei Liu, Yan-Pei Cao, Tianyuan Yang, Eric Zhongcong Xu, Jussi Keppo, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. 2023. HOSNeRF: Dynamic Human-Object-Scene Neural Radiance Fields from a Single Video. *arXiv preprint arXiv:2304.12281* (2023).

[22] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. 2015. SMPL: A skinned multi-person linear model. *ACM transactions on graphics (TOG)* 34, 6 (2015), 1–16.

[23] Nelson L Max. 1986. Light diffusion through clouds and haze. *Computer Vision, Graphics, and Image Processing* 33, 3 (1986), 280–292.

[24] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* 65, 1 (2021), 99–106.

[25] MrChimp2313. 2012. Stadium Blender Model. https://www.blendswap.com/blend/7488

[26] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)* 41, 4 (2022), 1–15.

[27] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. 2021. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5865–5874.

[28] Sida Peng, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Xiaowei Zhou, and Hujun Bao. 2021. Animatable neural radiance fields for modeling dynamic human bodies. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14314–14323.

[29] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. 2021. Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9054–9063.

[30] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. 2021. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10318–10327.

[31] Steven M Seitz and Charles R Dyer. 1999. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision* 35 (1999), 151–173.

[32] Ruizhi Shao, Zerong Zheng, Hanzhang Tu, Boning Liu, Hongwen Zhang, and Yebin Liu. 2023. Tensor4d: Efficient neural 4d decomposition for high-fidelity dynamic reconstruction and rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16632–16642.

[33] Liangchen Song, Anpei Chen, Zhong Li, Zhang Chen, Lele Chen, Junsong Yuan, Yi Xu, and Andreas Geiger. 2023. Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields. *IEEE Transactions on Visualization and Computer Graphics* 29, 5 (2023), 2732–2742.

[34] Cheng Sun, Min Sun, and Hwann-Tzong Chen. 2022. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5459–5469.

[35] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David McAllister, and Angjoo Kanazawa. 2023. Nerfstudio: A Modular Framework for Neural Radiance Field Development. In *ACM SIGGRAPH 2023 Conference Proceedings (SIGGRAPH '23)*.

[36] Krishna Wadhwani and Tamaki Kojima. 2022. SqueezeNeRF: Further factorized FastNeRF for memory-efficient inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2717–2725.

[37] Feng Wang, Sinan Tan, Xinghang Li, Zeyue Tian, and Huaping Liu. 2022. Mixed neural voxels for fast multi-view video synthesis. *arXiv preprint arXiv:2212.00190* (2022).

[38] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.

[39] Frederik Warburg, Ethan Weber, Matthew Tancik, Aleksander Holynski, and Angjoo Kanazawa. 2023. Nerfbusters: Removing Ghostly Artifacts from Casually Captured NeRFs. *arXiv preprint arXiv:2304.10532* (2023).

[40] Chung-Yi Weng, Brian Curless, Pratul P Srinivasan, Jonathan T Barron, and Ira Kemelmacher-Shlizerman. 2022. Humannerf: Free-viewpoint rendering of moving people from monocular video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern Recognition*. 16210–16220.

[41] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 586–595.

[42] Fuqiang Zhao, Wei Yang, Jiakai Zhang, Pei Lin, Yingliang Zhang, Jingyi Yu, and Lan Xu. 2022. Humannerf: Efficiently generated human radiance field from sparse inputs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7743–7753.

[43] Kun Zhou, Wenbo Li, Yi Wang, Tao Hu, Nianjuan Jiang, Xiaoguang Han, and Jiangbo Lu. 2023. NeRFLiX: High-Quality Neural View Synthesis by Learning a Degradation-Driven Inter-viewpoint MiXer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12363–12374.