# Deformable Neural Radiance Fields using RGB and Event Cameras Supplemental Material

Anonymous ICCV submission

Paper ID 9284

In this document, we provide additional details of our method and implementations. We further provide more qualitative and some quantitative results. Please also refer to the supplementary video for additional qualitative visualizations.

## A. Video Interpolation using Events for NeRF

The usage of event cameras in computer vision has been motivated by their unique advantages, namely, high dynamic range, low latency, ultra-low power consumption, and the absence of motion blur. [7] take advantage of synthesis-based and flow-based approaches which are robust to motion blur. [2] propose using the dynamic filtering layer to address the spatially variant threshold and represent the residuals between a blurry image and sharp frames as integrals of events [4]. [9] introduces a unified framework for event-based video deblurring and interpolation using a double integral (LDI) network and a fusion network. [3] propose a novel three-phase CNN architecture that fuses intensity images and event stream first at event camera resolution and then scales up to RGB resolution, followed by colorization. [8] deploys adversarial learning to reconstruct HR intensity images from LR event streams. Unlike the mentioned works, we do not interpolate the video frames and then train the deformation neural radiance Field upon them. Instead, we process the asynchronous events in their raw form which brings us computational efficiency while exploiting the temporal precision of events.

## B. Details of Synthetic Data

In this section, we provide additional details of the campfire and fluids dataset, created by us. The campfire and fluids datasets are made from Blender models. To simulate realistic fire and fluid in high framerate we set 960fps in Blender. The resolution of fire is $128^3$ while for fluids is $320^3$. We decrease the reaction speed for the campfire to 0.8 and increase the maximum temperature to 3.1 to control the flames rising speed. Turbulence is also introduced to bring uncertainty to the flaming process. For fluids, we increase the particle sampling per cell to better simulate the motion of water. More details such as customized shading can be found in the Blender file of our dataset, which we will make available.

## C. Details of PoseNet

Recall that in Section 3, we encode the pose residual term through screw axis representation: $\mathsf{S} = (r(t); v(t)) \in \mathbb{R}^6$. We train 2 MLPs which output $r(t) \in \mathbb{R}^3, v(t) \in \mathbb{R}^3$ after encoding time using sinusoidal positional encoding similar to the input coordinate $\gamma(t) : \mathbb{R}^1 \rightarrow \mathbb{R}^{1+\theta m}$. This allows the network to learn high-frequency representation from time input. We choose $m = 1$ for PoseNet and also for encoding time in the deformation network. Next, we calculate the translation and rotation using Rodrigues' rotation formula $\mathsf{R} = \mathsf{I} + \sin\theta\mathsf{K} + (1 - \cos\theta)\mathsf{K}^2$ where $\mathsf{R}$ is the rotation matrix, $\theta$ is the rotation angle which can be calculated by $\theta = \|r(t)\|$, and $\mathsf{K}$ is a skew-symmetric matrix also known as the cross product matrix of the unit vector $\hat{r}(t) = \frac{r(t)}{\|r(t)\|}$. We can also compute the translation vector as:

$$\mathbf{v} = \frac{(1 - \cos(\theta))(\hat{\mathbf{r}} \times \mathbf{v}) + \sin(\theta)\hat{\mathbf{r}} \times (\hat{\mathbf{r}} \times \mathbf{v}) + \cos(\theta)\mathbf{v}}{\theta^2},$$

As mentioned in Section 3.1, the PoseNet learns the pose residuals. Here, we compare the results for learning to pose residual and absolute pose, which are presented in Table 1. It is expected that learning residual yields better results as the initialized pose will be closer to the ground truth pose.

| Lego | MSE | PSNR | SSIM | LPIPS |
|---|---|---|---|---|
| Pose residual | 0.32 | 35.04 | 0.99 | 0.03 |
| Pose absolute | 2.18 | 26.09 | 0.96 | 0.151 |

Table 1. **Absolute pose vs. residual pose.** In the upper row the PoseNet is trained to learn the residual from the interpolated poses using RGB frame poses, while the bottom row is trained to learn the pose directly.

Recall that in Figure 5 of the main paper we provide the position error obtained by initial pose interpolation. Here

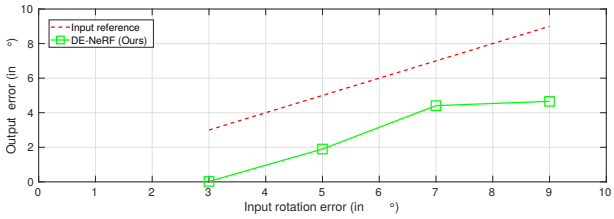


Figure 1. PoseNet robustness against injected rotation noise.

in Figure 1 we report the rotation error of our method after injecting different magnitudes of rotation noise. We found that our method is robust to small rotation noise and can effectively reduce large rotation noise.

| Number of Views | 10 | | 25 | | 50 | |
|---|---|---|---|---|---|---|
| Metinaccurately | LPIPS | PSNR | LPIPS | PSNR | LPIPS | |
| Train from scratch | 30.06 | 0.0623 | 33.95 | 0.045 | 35.45 | 0.036 |
| Finetune | 32.13 | 0.046 | 32.55 | 0.037 | 35.04 | 0.034 |

Table 2. **Training Protocol.** Training from scratch provide better results with more accurate pose initialization but for inaccurate pose initialization training PoseNet as finetuning after a first round of training performs better.

## D. More Details on Implementation & Training

We first dive into more details about the implementation of our DE-NeRF and then provide the training protocols and hyperparameters.We initialize the radiance field MLP with Xavier-initialization and similar to [5] for both deformation and PoseNet we initialize the output layer with uniform distribution from $\mathcal{U}(-10^{-4}, 10^{-4})$ so that the transformation will initially remain close to identity. The optimizer for training both the deformation radiance Field and the PoseNet are Adam [1]: $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate follows an exponentially decayed schedule from 2e-3 to 1e-4. For PoseNet the learning rate decayed from 1e-3 to 5e-5 for stable training.

Here we compare two different training protocols in **Non-rigid Lego** with a different number of frames used as shown in Table 2. Train from scratch implies that DE-NeRF learns the radiance field as well as the trajectory residual at the same time from the beginning. In contrast, the Finetune approach learns the deformation radiance field only based on the interpolated trajectory and then enables the PoseNet later for joint training. We found that when the pose is well-initialized, using 25 and 50 frames, better results are obtained when enabling PoseNet in the beginning. Conversely, when the initialized pose inaccurately for 10 frames cases, we found enabling the PoseNet after a pre-training improves the results. It can be attributed to the benefits of solving tasks one by one and the learned radiance field provides additional constraints for PoseNet training.

| $\lambda$ | MSE | PSNR | SSIM | LPIPS |
|---|---|---|---|---|
| 100 | 0.54 | 33.59 | 0.993 | 0.0429 |
| 10 | 0.30 | 35.44 | 0.997 | 0.0363 |
| 1 | 0.32 | 35.02 | 0.995 | 0.0533 |
| 0.1 | 1.02 | 30.24 | 0.981 | 0.0929 |
| 0.01 | 6.13 | 22.20 | 0.873 | 0.3124 |

Table 3. **Influence of $\lambda$.** PSNR values for different hyperparameter $\lambda$ of the RGB photometric loss on Lego dataset.

| Lego | MSE | PSNR | SSIM | LPIPS |
|---|---|---|---|---|
| Translation | 0.32 | 35.04 | 0.99 | 0.03 |
| SE(3) | 0.33 | 34.57 | 0.99 | 0.052 |
| Umbrella | MSE | PSNR | SSIM | LPIPS |
| Translation | 0.45 | 33.44 | 0.95 | 0.341 |
| SE(3) | 0.37 | 33.99 | 0.95 | 0.342 |

Table 4. **Different warps.** Two cases highlighting the pros and cons of SE(3) deformation field compared to translation only.

### D.1. Sensitivity to Hyperparameter $\lambda$

In Table 3 we analyze how the hyperparameters $\lambda_r gb$ impact the performance. It can be noticed that with small $\lambda_r gb$ the method may suffer due to color confusion caused by noisy brightness estimation. Similarly, the PSNR drops when $\lambda_r gb$ is too large as the event information cannot be fully exploited. Nevertheless, our experiments show that the proposed method is not too sensitive to the choice of the hyperparameter $\lambda$.

## E. Translation vs. SE(3) Deformation Field

As suggested in [5], we also experimented with SE(3) deformation field, separately from the translation-only deformation field reported in the paper. We found that SE(3) field occasionally performs better than that of translation only. These examples are Fluid and Umbrella. Therefore, the qualitative results in the supplementary material for the latter example are shown with SE(3) deformation field. These two cases can be directly compared in Table 4 (in the paper) and Table 5 (here) for the mentioned examples. Some quantitative results for translation only and SE(3) deformation field are reported in Table 4. It can be seen that the SE(3) deformation is not always better. Therefore, we reported translation-only deformation field results in the main paper, for simplicity.

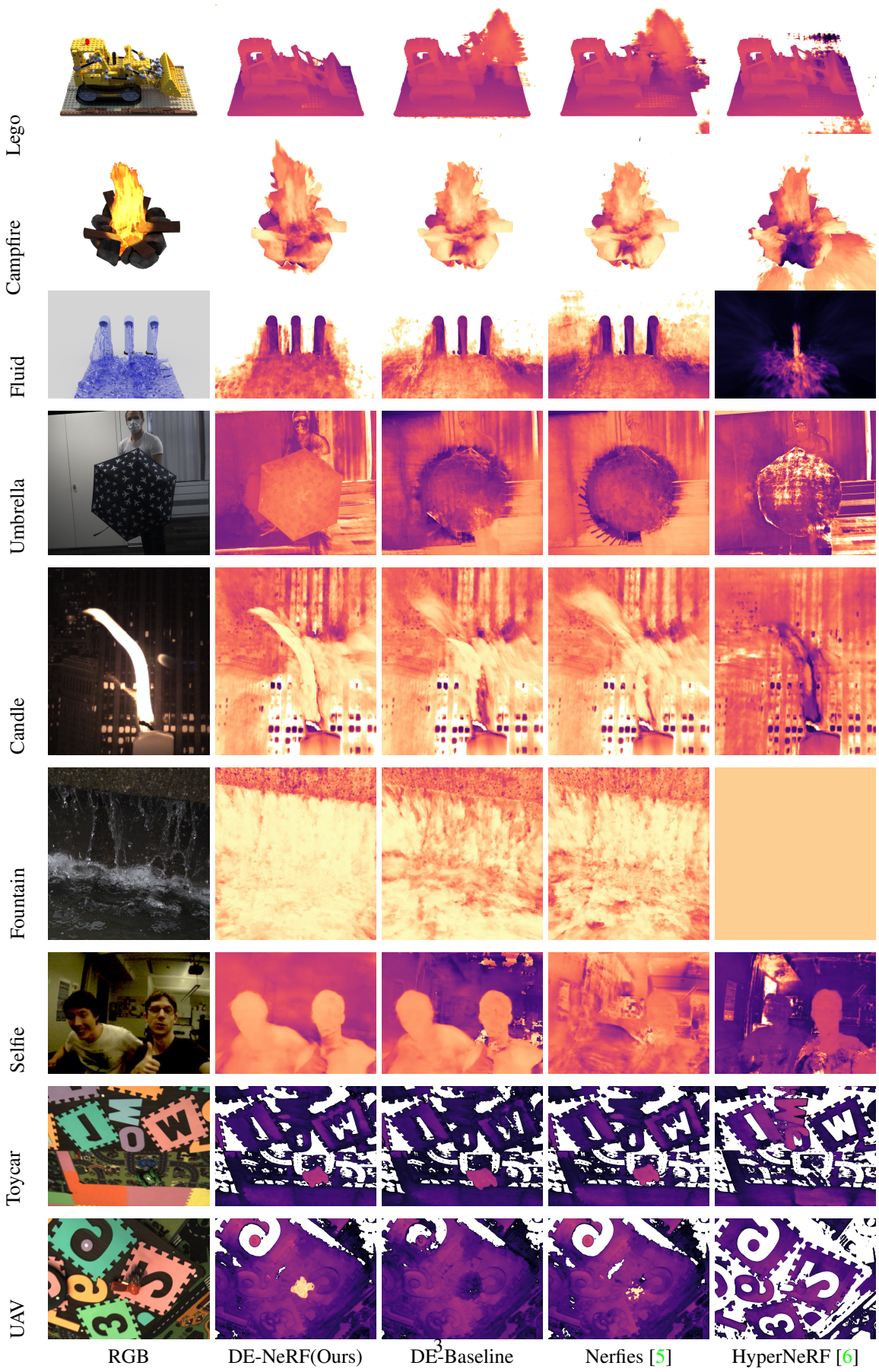


Table 5. Qualitative depth comparisons of our method against other methods on synthetic and real-world datasets.

## F. More Qualitative Results

We present more qualitative results for our method and comparison with other methods Figure 5. Please, also refer to our supplementary video.

## References

[1] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[2] Songnan Lin, Jiawei Zhang, Jinshan Pan, Zhe Jiang, Dongqing Zou, Yongtian Wang, Jing Chen, and Jimmy Ren. Learning event-driven video deblurring and interpolation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*, pages 695–710. Springer, 2020. 1

[3] Genady Paikin, Yotam Ater, Roy Shaul, and Evgeny Soloveichik. Efi-net: Video frame interpolation from fusion of events and frames. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1291–1301, 2021. 1

[4] Liyuan Pan, Cedric Scheerlinck, Xin Yu, Richard Hartley, Miaomiao Liu, and Yuchao Dai. Bringing a blurry frame alive at high frame-rate with an event camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6820–6829, 2019. 1

[5] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 2, 3

[6] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021. 3

[7] Stepan Tulyakov, Daniel Gehrig, Stamatios Georgoulis, Julius Erbach, Mathias Gehrig, Yuanyou Li, and Davide Scaramuzza. Time lens: Event-based video frame interpolation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16155–16164, 2021. 1

[8] Lin Wang, Tae-Kyun Kim, and Kuk-Jin Yoon. Eventsr: From asynchronous events to image reconstruction, restoration, and super-resolution via end-to-end adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8315–8325, 2020. 1

[9] Xiang Zhang and Lei Yu. Unifying motion deblurring and frame interpolation with events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17765–17774, 2022. 1