Optical Flow Regularization of Implicit Neural Representations for Video Frame Interpolation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recent works have shown the ability of Implicit Neural Representations (INR) to carry meaningful representations of signal derivatives. In this work, we leverage this property to perform video frame interpolation by explicitly constraining the derivatives of the INR to satisfy the optical flow constraint equation. We achieve state of the art video frame interpolation on limited motion ranges using only a target video and its optical flow, without learning the interpolation operator from additional training data. We further show that constraining the INR derivatives not only allows to interpolate intermediate frames but also improves the ability of narrow networks to fit observed frames, which suggests potential applications to INR optimization and video compression.

1 Introduction

Many core concepts across the fields of signal processing are defined in terms of continuous functions and their derivatives: surfaces are continuous manifolds in space, motion is a rate of change in space through time, etc. In contrast, the modern digital infrastructure is inherently discrete: digital sensors capture discrete observations of the world sampled in time and space; digital computers store and process discrete representations of signals. In order to model continuous notions on discrete signal representations, classical signal processing approaches have resorted to a variety of heuristics and assumptions, often taking the form of constant first or second derivatives of the signal between consecutive observations. The lack of generality of any such handcrafted heuristics, combined with the ever improving quantitative results of Machine Learning (ML) approaches, have led to the near ubiquitous use of ML approaches in recent signal processing research. These approaches leverage large collections of data to infer statistical properties of signals instead of hand-crafted heuristics.

In computer vision, Video Frame Interpolation (VFI) is one task representative of such development. VFI models aim to interpolate intermediate frames between the consecutive frames of a video. To do so, most successful approaches rely on the optical flow as an approximation of the motion field to guide the interpolation of pixel intensities from the grid of two consecutive frames onto the pixel grid of intermediate frames. Classical approaches formulate assumptions such as constant speed or acceleration of the motion field between consecutive frames [CITE]. The value of each pixel in the inferred intermediate frame is computed by first shifting the pixel intesities of the observed frames following the optical flow directions, and then interpolating the shifted pixel intensities onto the intermediate frame's pixel grid. These approaches suffer from the following two limitations:

- Optical flow constraint used to infer the optical flow holds for limited situations.
- Linear interpolation of pixel intensities along the optical flow directions does not hold in practice.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

These limitations share a common root cause: discretization. Indeed, both the optical flow constraint and the constant motion field assumption only truly hold at the infinitesimal scale, for much smaller time deltas than typical FPS used in practice.

ML approaches [CITE] have instead proposed to learn the frame interpolation operator from large video collections, without explicitly formulating any assumption on the optical flow. While these approaches have achieved great success in terms of benchmark performance, they are prone to generalization errors when applied to unseen videos. Indeed differences between the training set distribution (i.e. VFI benchmark videos) and the target video distribution hinders the performance of ML approaches: differences in the range of motion, exposure time and frame-per-second have been shown to limit the generalization of state-of-the-art models to video frame interpolation in the wild [CITE].

In the mean time, research on implicit representations seek better discrete representations of continuous signals. In recent years Implicit Neural Representations (INR), i.e. representing signals as Neural Networks (NN) have been shown to offer several competitive advantages over explicit representations, with notable early successes for 3D shape representations [CITE]. Of particular interest to us is the work of SIREN [CITE], in which it has been shown that representing signals using Multi Layer Perceptrons (MLP) with sine activation functions carry meaningful representations of the signal derivatives. Inspired by this work, we question wether such approach may be used to guide the interpolation process of VFI by controlling the exact derivatives of the signal rather than finite differences, thus avoiding the discretization pitfalls of traditional approaches. We do so by constraining the derivatives of SIREN representations to satisfy the optical flow constraint, i.e., to be orthogonal to the video's optical flow (which we compute using existing state-of-the-art OF models). We find that this approach outperforms most existing machine learning-based approaches on small motion range benchmarks, without relying on machine learning for the interpolation operator: we simply regularize the implicit representation to satisfy the definition of the optical flow. In this sense, our approaches is most similar to classical VIF approaches, except that instead of wrapping the OF on discrete explicit frame representations, we apply the optical flow constraint on the exact gradient of the the INR. Our method is thus not subject to any mismatch between training and test data. Furthermore, our approach can sample any number of frame in-between the observed frames due to the continuous nature of the representation. In addition to its application to VFI, we also show that constraining the gradient of the model also improves the ability of narrow MLPs to fit the signal, suggesting potential applications in INR optimization and video compression.

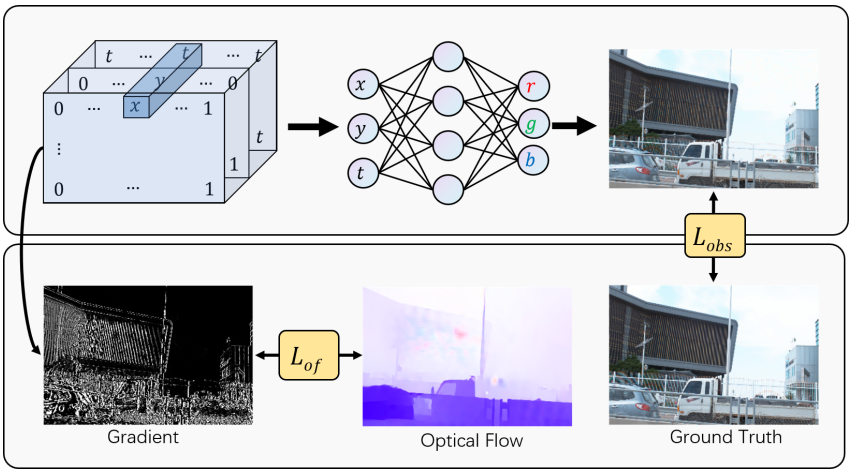


Figure 1: Illustration of our approach

To summarize, the contributions of this work are:

- We propose a regularization method for SIREN which achieve state-of-the-art video frame interpolation on small motion ranges.
- In contrast to other state-of-the art approaches, our approach does not rely on training on a large external training set. It only relies on the target video and its estimated optical flow.

- We show that our regularization approach not only helps generalizing to intermediate frame generalization but also helps narrow models fit the observed frames.

On the other hand, our approach (in its current form) presents important limitations:

- It relies on an input optical flow, which is computed using existing ML-based model and thus suffers the limitations of ML approaches.
- Optimization of the INR is very time-consuming, which hinders our ability to work on full resolution videos for time constraints.
- Our method currently only works on limited motion range. It does not match state-of-the art ML models on large motion ranges.

While we acknowledge the importance of the above limitations, we believe these to not be fundamental limitations of our approach but rather important future INR research directions. We discuss these limitations at length and present possible axis to tackle them in Section XXX. The remainder of this paper is organized as follows: We briefly present some related work in Section XXX, the detail of our method in Section XXX, and design several experiments to highlight the advantages of our approach in Section XXX.

2 Related Work

Classical video interpolation. The classical approach relies on optical flow to characterize the motion relationship between two frames. The quality of interpolation depends on the accuracy of the optical flow. Classical optical flow is usually sparse and can only work at one pixel or sub-pixel level of motion [1]. Sparse feature extraction [2] is used to enhance the optical flow correctly. In order to synthesize dense optical flow and work on large displacements, [3] uses sparse convolution and max-pooling. But instead of learning the model, the parameters are set manually.

Deep learning video interpolation. A number of deep learning models have been developed for video interpolation tasks. Almost all models can be categorized as: optical flow based, and kernel based.

Optical Flow-Based. Optical flow-based approaches are the most popular in video frame interpolation. The standard technique of video frame interpolation aims at explicitly estimating motion in the form of optical flow, warping two input frames to an intermediate frame, and synthesizing the occlusion region. The frames are constrained by the assumption of linear motion and constant luminance between them. However, video interpolation of video frames is heavily dependent on the accuracy of optical flow.

The Super-SloMo [4] proposed by Jiang et,al. is a non-negligible work in the task of optical flow-based video frame interpolation. Super-SloMo extends the U-Net architecture proposed by Liu et al [5]. The bilateral optical flow is calculated for the input two frames and approximates the key frame with the intermediate optical flow of the two frames. Then the frames of the input are warped according to the obtained intermediate optical flow.

RRIN [6] mentioned that the estimation of intermediate frames in Super-SloMo works poorly near the boundaries because the optical flow is not locally smooth in these regions. RRIN proposes to improve the accuracy of optical flow by residual learning. BMBC [7] adds two additional approximate vectors to Super-SloMo to make the bilateral motion estimation more accurate.

Kernel-Based. To avoid explicit motion estimation and warping stages, the kernel-based approach performs a convolution operation on the input frames and the output of the convolution is used as the result of interpolating the frames. Niklaus et al. [8] proposed a fully convolutional deep neural network using a spatially adaptive convolutional kernel to perform the prediction of intermediate frames for two frames with consecutive inputs. Niklaus et al. [9] improved their method by using a separable convolution with spatially adaptive one-dimensional convolutional kernel pairs estimated for each pixel, in reducing the parameters of the model. The results of kernel-based methods for frame interpolation can be limited by the size of the kernel.

Lee et al. proposed Adacof [10], which can use any pixel at any position for convolution operation, so that the convolution kernel is no longer limited to the local range. And many methods residing

in optical flow are defined as a special case of Adacof. However, most kernel-based methods can only generate one intermediate frame, and if one wants to generate multiple intermediate frames, one needs to do it recursively. EDSC [11] is the first kernel-based method proposed to generate multiple intermediate frames, but the results are not as good as the optical flow method.

Implicit Neural Representation. (INR)

INR use a neural network to represent an object approximately, which is essentially a way to parameterize the signal. Since [12], [13] was developed, INR has performed well in the areas of 3D vision tasks, images, and video. The image and video tasks most relevant to this paper are around the direction of image/video compression.

COIN [14] first proposed the use of INR to compress images, mapping pixel coordinates to RGB values. COIN++ [15] cooperated with the meta-learning approach for image compression work based on COIN. In the field of video compression, NeRV [16] proposed by Chen et al. successfully encodes the video into a neural network, i.e., the content of the video is saved using a neural network. Only the frame index of the model needs to be provided, and the corresponding RGB picture is output. In other words, this makes it possible to output infinite frames of video using a neural network. Although NeRV briefly attempts the task of performing video frame interpolation, this is not NeRV's main work. The NRFF [17] proposed by Rho et al., which uses optical flow and residuals information for video compression, does not directly fit all frames.

Most related to our approach is the concurrent work by Shangguan et al. [18], which also uses INR for video interpolation tasks. Their approach, CURE, uses machine learning. It requires visual features of the video and does not fully map the pixel coordinates and frame positions of the video to RGB images.

3 Method

We consider a ground-truth video as a continuous signal v mapping continuous spatial (x, y) and temporal (t) coordinates to RGB values:

$$\begin{aligned} v &: (x, y, t) \rightarrow (R, G, B) \\ v &: \mathbb{R}^3 \rightarrow \mathbb{R}^3 \end{aligned} \tag{1}$$

Our goal is to find a continuous function f_θ, parameterized by a finite parameter set $\theta \in \Theta$, with minimum distance d to the ground-truth signal:

$$\begin{aligned} f_\theta &: (x, y, t) \rightarrow (R, G, B) \\ s.t.\ \theta &= min_\Theta \iiint d(f_\theta(x, y, t), v(x, y, t)) dx dy dt \end{aligned} \tag{2}$$

where the distance function d may either be the Peak Signal to Noise Ratio (PSNR) or the Structural Similarity Index Measure (SSIM). To do so, we only have access to regularly sampled observation of the signal v (i.e. the explicit representation of the video), which we denote as:

$$\begin{aligned} \mathcal{V} &\in \mathbb{R}^{T \times H \times W \times 3} \\ s.t.\ \mathcal{V}_{xyt} &= v(x, y, t) \end{aligned} \tag{3}$$

where T represents the number of frames in the video, and $H \times W$ the spatial resolution. We use SIREN as parameterized function class f_θ. The most straightforward way to approximate Equation 2 is to optimize the model parameters so as to fit the video frames, using the following loss function we refer to as the observation loss:

$$\mathcal{L}_{obs} = \frac{1}{HWT} \sum_{x=1}^{W} \sum_{y=1}^{H} \sum_{t=1}^{T} ||f_\theta(x, y, t) - \mathcal{V}_{xyt}||^2 \tag{4}$$

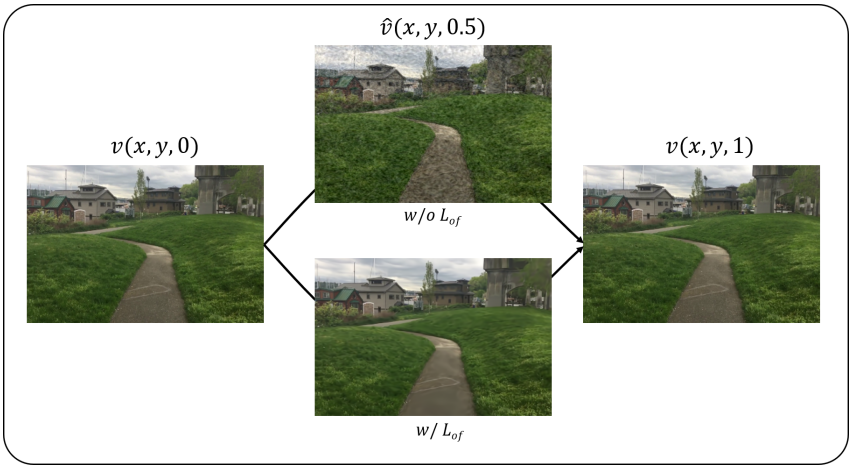


Figure 2: Illustration of INR frame interpolation with and without optical flow regularization. Without regularization (middle top), intermediate frames show unnatural high-frequency variations. Regularizing the INR to satisfy the optical flow constraint equation result in nicely interpolated frames (middle bottom).

However, we found that optimizing the INR to only minimize this observation loss leads to overfitting the observation with high temporal frequencies: the intra-frame signal, which we aim to correctly recover, shows important deviations from the observed frames, as illustrated in Figure XXX. This observation has lead us to consider fitting not only the signal itself, but to also constrain its derivatives. In particular, we regularize the model so as to respect the optical flow constraint.

The optical flow constraint equation states that for an infinitesimal lapse of time δt, the brightness of physical points perceived by a camera at arbitrary coordinates (x, y, t) should remain constant. In other words, given the displacement $(\delta x, \delta y)$ of a physical point in the image coordinate system, the image brightness v should remain constant:

$$v(x, y, t) = v(x + \delta x, y + \delta y, t + \delta t) \tag{5}$$

We introduce the vector notation $\mathbf{x} = (x, y, t)$ for readability. Expressing movement as a ratio of displacement in time and abbreviating, we can write the optical flow F and the above constraint as:

$$\begin{aligned} F(\mathbf{x}) &= (\frac{\delta x}{\delta t}, \frac{\delta y}{\delta t}, 1) \\ v(\mathbf{x}) &= v(\mathbf{x} + F(\mathbf{x})) \end{aligned} \tag{6}$$

First order Taylor expansion allows us to rewrite expand equation XXX into the following

$$\begin{aligned} v(\mathbf{x}) = v(\mathbf{x}) + \frac{\delta v}{\delta \mathbf{x}} \cdot F(\mathbf{x}) \\ \frac{\delta v}{\delta \mathbf{x}} \cdot F(\mathbf{x}) = 0 \end{aligned} \tag{7}$$

which holds exactly in the limit of infinitesimal δt. We leverage this optical flow constraint equation to regularize the INR. Denoting the derivatives of the SIREN as:

$$D(f, \theta, x, y, t) = \Big(\frac{\delta f_\theta(x, y, t)}{\delta x}, \frac{\delta f_\theta(x, y, t)}{\delta y}, \frac{\delta f_\theta(x, y, t)}{\delta t} \Big) \tag{8}$$

we can now define the optical flow regularization loss

$$\mathcal{L}_{of} = \frac{1}{HWT} \sum_{x \in W} \sum_{y \in H} \sum_{t \in T} |D(f, \theta, x, y, t) \cdot F(x, y, t)| \tag{9}$$

This loss constrains the derivatives of the signal to be orthogonal to the optical flow and can be understood as keeping constant brightness along the optical flow trajectories. The total loss we use to optimize the INR is a weighted sum of these two terms:

$$\mathcal{L} = \lambda \mathcal{L}_{obs} + (1 - \lambda)\mathcal{L}_{of} \tag{10}$$

where λ is a hyper-parameter taking values between 0 and 1 whose impact we investigate in the following section. The exactitude of the optical flow constraint at the infinitesimal scale plays in our favor: As we regularize the true derivative of the signal representation, we do not assume constant derivatives of the signal on any interval. We believe this is the main factor behind our positive results. On the other hand, the optical flow we used was estimated from discrete consecutive frames, and thus does not represent the true infinitesimal motion field but an estimation of finite differences. We discuss this limitation in Section XXX.

4 Experiments

Following previous works, we use the Adobe[CITE], X4K[CITE] and ND Scene[CITE] dataset as benchmarks to compare to the state-of-the-art. We run all additional experiments on the 720p240fps1.mov video of Adobe dataset illustrated in Figure 2. Due to the time-consuming operation of optimizing SIREN representations, we optimize and evaluate all models on a 240×360 resolution. For each video in the Adobe dataset, we selected only the first 40 frames for our experiments. For the Adobe dataset, we used the same dataset splitting approach as Super-SloMo, where eight videos were used as the testing set.

We show the effect of different hyperparameters on the model in Figure 4. We use the greedy algorithm to find the best combination of hyperparameters. We will end up using a SIREN model with 6 depth and 720 width. We use an omega of 25 and a lambda of 0.12. We optimize the models using the Adam optimizer using a cosine learning rate with maximum learning rate of 3.6e-5 during 15k epochs.

We start by showing the impact of controlling the fit of high frequencies without the optical flow loss in section **??**. We show that limiting the frequency fitted improves interpolation at the cost of degrading the fit of observed frames, but it does not allow to reach the same accuracy as optical flow regularization, showing that OF regularization does more than only regularizing the fitted frequency range. In Section 4.2, we quantitatively compare our results to state of the art models on standard datasets. We show that our approach achieves state-of-the-art results on low-range motion datasets, but underperforms existing methods for high-range motion videos. We present an ablation study in Section 4.3, providing insights and appropriate settings for the different model hyper-parameters, and a qualitative analysis of our results in Section 4.4.

Finally, we report an additional counter-intuitive result in Section 4.5. We show that our proposed optical flow regularization loss can help the lightweight SIREN model to fit the video better. This indicates that our method is potentially helpful for video compression.

4.1 Optical flow constraint and high frequencies

Figure 2 illustrates the fact that applying the optical flow constraint smoothes out the high-frequency variations from the intermediate frames of vanilla SIREN representations. We start by questioning wether the OF constraint does more than simply removing the high frequency variations of the representation. To do so, we compare the results of vanilla SIREN representations geared towards different frequencies and compare their interpolated frames PSNR to those of OF-constrained representations. We constrain the SIREN frequency by varying their ω parameter, and report our comparison in Figure 3.

Constraining the high frequency with ω down to 5 degrades the fitting of observed frames, but improves the quality of interpolated frames. This suggests that ω behaves similarly to a regularization parameter controlling a regime of overfitting versus underfitting of the observed frames high frequencies. Figure 3 also shows that vanilla SIREN models remain well under the OF-constrained SIREN, confirming than the OF constraint provides more than high frequency regularization.

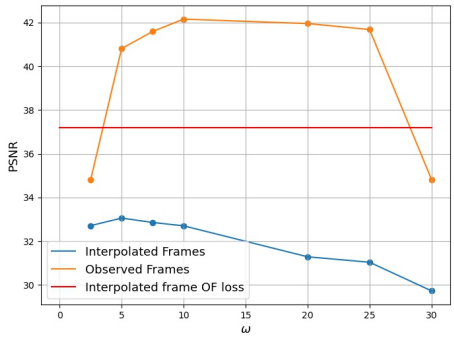


Figure 3: Evolution of the PSNR of observed and interpolated frames with ω without OF loss. Limiting the high frequency fit alone does not reach the same interpolation accuracy as the OF loss.

Table 1: Quantitative comparison to state-of-the-art VFI on Standard benchmarks. Results are formatted as PSNR / SSMI.

(a) Limited motion range

	Adobe-240FPS	X4K
Super-SloMo []	27.77 / 0.886	27.38 / 0.852
RRIN [XXX]	32.37 / 0.962	30.70 / 0.927
BMBC [XXX]	27.83 / 0.917	27.42 / 0.858
AdaCof [XXX]	35.50 / 0.968	34.61 / 0.921
ABME [XXX]	35.28 / 0.966	34.30 / 0.919
FILM [XXX]	35.97 / 0.971	**35.14** / 0.939
Ours	**36.52** / **0.977**	35.06/ **0.944**

(b) Large motion range

	ND Scene
V-NF [X]	23.30 / 0.726
NSFF [X]	28.03 / 0.925
CURE [X]	**36.91** / **0.984**
Ours	29.22 / 0.921

4.2 State of the art models

Table XXX quantitatively compares the results of our method to state-of the art VFI models on different datasets. Despite its simplicity, and without any training data, our approach outperforms existing models on limited motion ranges (Table XXX). However, as illustrated in Figure XXX, it falls short of state-of-the-art methods on the more complex ND Scene benchmark due to larger motion ranges. We provide further comparison in the qualitative analysis of Section XXX and section XXX discusses possible ways forward to bridging the gap performance on large motion datasets.

4.3 Ablation study

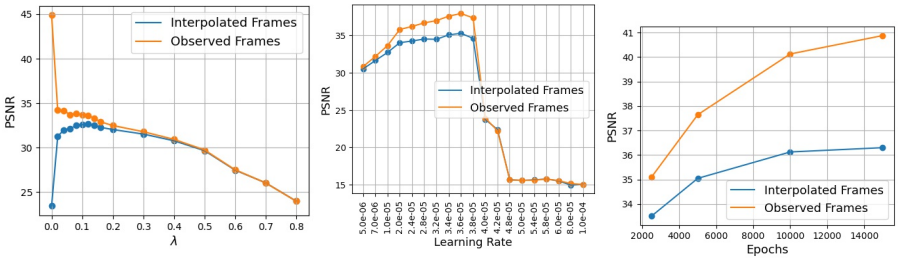
We describe in this section the method of searching for the best hyperparameter combination using the greedy algorithm. Figure 4 illustrates the results of the hyperparameter searching. We will start with a SIREN model of depth 9 and width 512. The baseline experiment is set up with a learning rate of 1e-5, 5000 epochs, and 30 ω . The order of our hyperparameter search is the lambda controlling the loss balance λ, the learning rate, epochs, omega, and the depth and width of SIREN. For each new hyperparameter searching, the best combination of previous hyperparameters is used.

4.4 Qualitative analysis

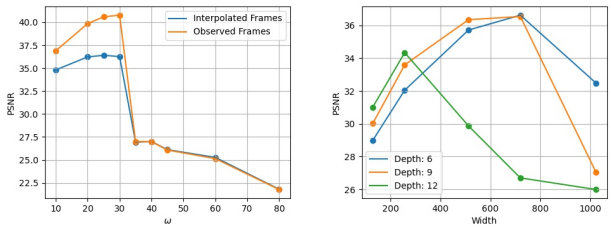
Do qualitative analysis

4.5 Video fitting

Fit video.

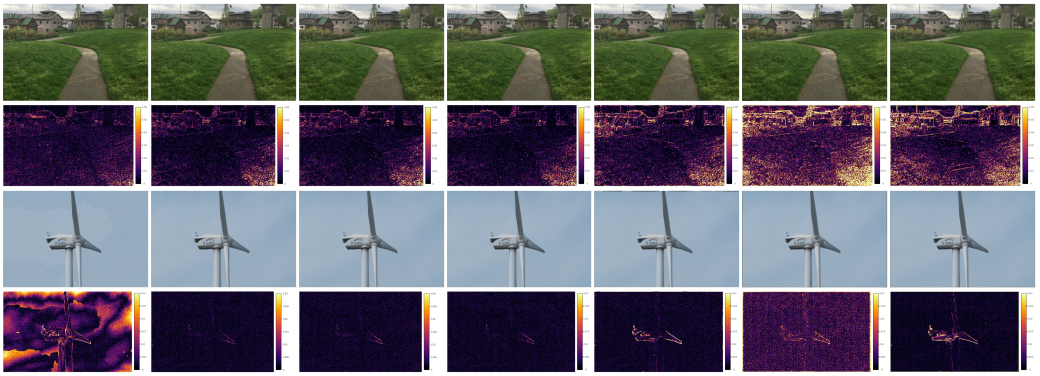


(a) Loss balance λ (b) Learning Rate (c) Epochs



(d) ω (e) Width and Depth

Figure 4: Hyperparameter Searching and Ablation Experiments.



(a) Ours (b) ABME (c) FILM (d) AdaCof (e) BMBC (f) RRIN (g) SloMo

Figure 5: Small Motion Video Qualitative Analysis. The interpolated frame results and the residual heat map of the two videos are shown. Our proposed method can fit high frequency details well (e.g., grass), but fits low frequency information poorly (e.g., sky). The BMBC and RRIN lose the pixels at the edges.

5 Limitations

While we believe our results to be very encouraging, the proposed approach is not yet practical. Here, we discuss what we believe to be the three main limitations of, and possible solutions to, our approach

Slow optimization process. Fitting XXX frames of a video at XXX resolution currently takes XXX hours on a XXX GPU using Pytorch. This computation time is a huge draw back as it limits our ability to process full resolution video as well as to explore different hyper parameters and variants of the methods. We expect new methods speeding up the convergence of video INR to be very beneficial to this line of research. Given recent successes of INR approaches to high impact applications (i.e., video compression [CITE]), We hopefully expect to see advances in INR optimization research.

Reliance on trained optical flow model. SIREN models allow us to apply the optical flow on the exact derivatives of the signal, bypassing the heuristics of classical approach without relying on machine learning. The optical flow we use, however, is given by a ML model trained on discrete representations, which raises two problems: it is subject to the same generalization errors as ML

(a) $v(x, y, 0)$ (b) $\hat{v}(x, y, 0.5)$ (c) $v(x, y, 0.5)$ (d) $v(x, y, 1)$ (e) Residual Error

Figure 6: Large Motion Video Qualitative Analysis

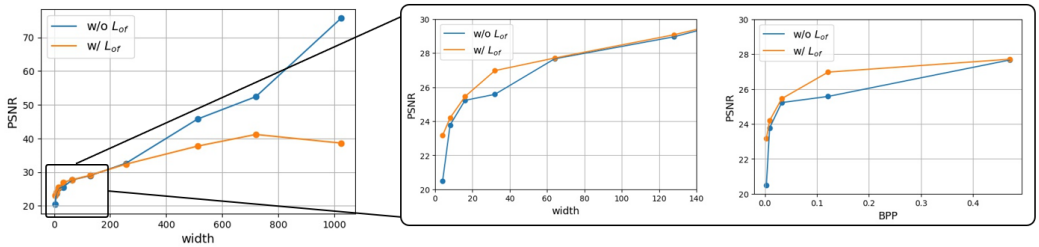


Figure 7: Our proposed optical flow regularization loss can help the lightweight SIREN model to fit the video better.

approaches, and is subject to finite difference errors, i.e. noisy illumination variations and occlusions. Future work will aim to bypass our reliance on ML-based OF using alternative constraints on the exact derivatives.

Inability to interpolate high motion range videos. In its current form, we only apply the optical flow constraint on the observed frames of the video. This has proven sufficient to reach state-of-the art on low motion ranges but is not sufficient for large motions. A promising axis of improvement would be to apply further regularization on the interpolates frames (i.e for non-inter input times like $t = 0.5$)). Possible regularization may include interpolated optical flows, or texture constraints as has been proposed in related works [COTE], which may prevent the ghosting effects illustrated in Figure XXX.

6 Conclusion

In this paper, we have shown that regularizing INR using the optical flow constraint equation enabled VFI without relying on ML to perform the interpolation step. We have shown that this approach is sufficient to reach state-of-the-art interpolation on low motion range videos, without resorting to learning-based interpolation. This method is not yet practical for high resolution large motion range videos due to the three limitations highlighted in the previous section. Nevertheless, these limitations can be tackled and we hope the insights presented in this paper can serve prove useful in the future, either on their own or combined with related approaches.

References

[1] Lei Chen, Hua Yang, Takeshi Takaki, and Idaku Ishii. Real-time optical flow estimation using multiple frame-straddling intervals. *Journal of Robotics and Mechatronics*, 24(4):686–698, 2012.

[2] Ce Liu, Jenny Yuen, Antonio Torralba, Josef Sivic, and William T Freeman. Sift flow: Dense correspondence across different scenes. In *European conference on computer vision*, pages 28–42. Springer, 2008.

[3] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. Deepflow: Large displacement optical flow with deep matching. In *Proceedings of the IEEE international conference on computer vision*, pages 1385–1392, 2013.

[4] Huaizu Jiang, Deqing Sun, Varun Jampani, Ming-Hsuan Yang, Erik Learned-Miller, and Jan Kautz. Super slomo: High quality estimation of multiple intermediate frames for video interpolation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9000–9008, 2018.

[5] Ziwei Liu, Raymond A Yeh, Xiaoou Tang, Yiming Liu, and Aseem Agarwala. Video frame synthesis using deep voxel flow. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4463–4471, 2017.

[6] Haopeng Li, Yuan Yuan, and Qi Wang. Video frame interpolation via residue refinement. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2613–2617. IEEE, 2020.

[7] Junheum Park, Keunsoo Ko, Chul Lee, and Chang-Su Kim. Bmbc: Bilateral motion estimation with bilateral cost volume for video interpolation. In *European Conference on Computer Vision*, pages 109–125. Springer, 2020.

[8] Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive convolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 670–679, 2017.

[9] Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive separable convolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 261–270, 2017.

[10] Hyeongmin Lee, Taeoh Kim, Tae-young Chung, Daehyun Pak, Yuseok Ban, and Sangyoun Lee. Adacof: Adaptive collaboration of flows for video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5316–5325, 2020.

[11] Xianhang Cheng and Zhenzhong Chen. Multiple video frame interpolation via enhanced deformable separable convolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[12] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020.

[13] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems*, 33:7462–7473, 2020.

[14] Emilien Dupont, Adam Goliński, Milad Alizadeh, Yee Whye Teh, and Arnaud Doucet. Coin: Compression with implicit neural representations. *arXiv preprint arXiv:2103.03123*, 2021.

[15] Emilien Dupont, Hrushikesh Loya, Milad Alizadeh, Adam Goliński, Yee Whye Teh, and Arnaud Doucet. Coin++: Data agnostic neural compression. *arXiv preprint arXiv:2201.12904*, 2022.

[16] Hao Chen, Bo He, Hanyu Wang, Yixuan Ren, Ser Nam Lim, and Abhinav Shrivastava. Nerv: Neural representations for videos. *Advances in Neural Information Processing Systems*, 34, 2021.

[17] Daniel Rho, Junwoo Cho, Jong Hwan Ko, and Eunbyung Park. Neural residual flow fields for efficient video representations. *arXiv preprint arXiv:2201.04329*, 2022.

[18] Wentao Shangguan, Yu Sun, Weijie Gan, and Ulugbek S Kamilov. Learning cross-video neural representations for high-quality frame interpolation. *arXiv preprint arXiv:2203.00137*, 2022.