Fast Online Video Super-Resolution with Deformable Attention Pyramid

Dario Fuoli[1] Martin Danelljan[1] Radu Timofte[1,2] Luc Van Gool[1,3]
[1]Computer Vision Lab, ETH Zürich, Switzerland [2]CAIDAS, University of Würzburg, Germany [3]KU Leuven, Belgium
{dario.fuoli, martin.danelljan, vangool}@vision.ee.ethz.ch, radu.timofte@uni-wuerzburg.de

Abstract

Video super-resolution (VSR) has many applications that pose strict causal, real-time, and latency constraints, including video streaming and TV. We address the VSR problem under these settings, which poses additional important challenges since information from future frames is unavailable. Importantly, designing efficient, yet effective frame alignment and fusion modules remain central problems. In this work, we propose a recurrent VSR architecture based on a deformable attention pyramid (DAP). Our DAP aligns and integrates information from the recurrent state into the current frame prediction. To circumvent the computational cost of traditional attention-based methods, we only attend to a limited number of spatial locations, which are dynamically predicted by the DAP. Comprehensive experiments and analysis of the proposed key innovations show the effectiveness of our approach. We significantly reduce processing time and computational complexity in comparison to state-of-the-art methods, while maintaining a high performance. We surpass state-of-the-art method EDVR-M on two standard benchmarks with a speed-up of over $3\times$.

1. Introduction

Video super-resolution (VSR) is the problem of restoring spatial high-frequency components from low-resolution video frames. In contrast to single image super-resolution, where methods are bound to rely on image priors, VSR offers the opportunity to utilize additional observations from adjacent frames and long-range temporal correlations to reconstruct a single frame. For this reason, effective frame alignment and fusion of salient features along the temporal axis constitute the main challenges in VSR.

Many practical applications, including TV and video streaming, depend on the ability to run algorithms online and in real-time, where minimal latency and high-speed processing are essential. However, hard time constraints in online video processing pose major challenges for learned VSR, as high performance strongly correlates with computational complexity in deep neural networks (DNN). Contrary to many other computer vision problems, it is thus important to carefully optimize the network's performance while minimizing architectural complexity. In addition to fast inference, a tailored solution to the problem of online VSR is required. Contrary to many prior works [9, 2, 3, 14, 28, 18, 12], we therefore address the problem of designing a strictly causal VSR approach. This imposes an additional challenge, since causality prohibits the access to information from future frames.

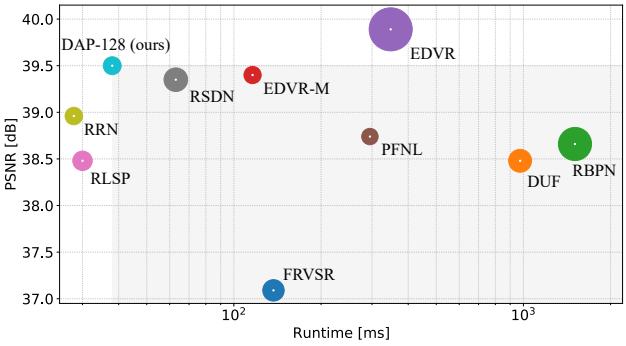

Figure 1: Runtime vs. performance on UDM10 [33] (runtime in log-scale). Disk areas correspond to number of parameters for each method. Our method DAP-128 achieves highly competitive performance with high speed (38ms per frame) and minimal complexity. Light gray highlights the Pareto dominant region of DAP-128.

Designing effective yet efficient alignment and fusion methods for VSR brings considerable challenges. Existing methods use inefficient alignment strategies, *e.g.* expensive alignment in feature space [28], exhaustive attention computation [33, 18] or ineffective implicit convolution-based alignment [15], without specific care about runtime. Most rely on information from neighboring frames only and neglect the potential of computation reuse between consecutive frames. As a consequence of missing efficient alignment/fusion mechanisms, fast methods generally avoid such modules entirely [6, 13, 11]. In this work, we employ a recurrent VSR architecture due to its online nature and efficiency, and address the aforementioned open issues of efficient alignment and fusion.

We propose a VSR approach by taking inspiration from

recent advances in attention [29] and transformers [27, 5]. As a case in point, attention- and transformer-based solutions have been successfully employed to computer vision tasks. Attention provides an advantage over convolutions as it allows effective matching and fusion of global information in early layers. Additionally, the operation implicitly serves for alignment to handle the displacements between frames in VSR. While the mechanism facilitates high performance, its quadratic complexity along with exhaustive correlation computations often render it unsuitable to time critical applications, especially in high-dimensional domains like video. In order to leverage its potential, the attention mechanism requires major adaptations to fulfil the requirements for fast video processing.

We tackle the aforementioned challenges by dynamically predicting pairs to be utilized for attention, thereby averting the high computational complexity in classic attention algorithms. In particular, we employ a deformable attention pyramid (DAP) for efficient information fusion at dynamically computed locations in the hidden state of our recurrent unit. DAP simultaneously addresses the misalignment and fusion through flexible offset prediction and discriminative aggregation with attention. The deformable attention mechanism allows robust fusion of spatially shifted features and counteracts error accumulation in the hidden state by means of dynamic selection of informative features.

For fast offset prediction we utilize a light-weight convolutional network. However, shallow convolutional networks suffer from small receptive fields due to their locality bias. This drawback limits the ability to handle large spatial displacements caused by movement between frames. We efficiently expand the receptive field by using a pyramid type network comprising a multi-level encoder followed by iterative attention-based offset refinement. According to the computed offsets, our fusion module effectively aggregates information from the hidden state. After the alignment/fusion stage, the combined processing of hidden state features and upsampling is performed by residual convolutional blocks, which ultimately output the high-resolution frame and the next hidden state.

Our experiments show great benefits of our proposed modules to the problem of VSR. An extensive ablation study clearly highlights the effectiveness of our contributions. We significantly reduce processing time and computational complexity in comparison to state-of-the-art methods, while achieving high performance. We attain higher PSNR than state-of-the-art method EDVR-M $+0.06$dB with a speed-up of over $3\times$ on the standard benchmark REDS.

2. Related Work

The ability to leverage complementary information in the temporal dimension for improved interpolation quality, represents a major difference between VSR and single-image super-resolution, where restoration algorithms are constrained to rely on priors only. An overview of recent state-of-the-art VSR methods is provided by [20, 19, 7, 25]. Two distinct mechanisms have been proposed in the literature to leverage this extra information in VSR; (1) sliding windows [28, 33, 18, 12, 30] and (2) recurrent processing [31, 22, 8, 6, 11, 13, 2, 3]. Sliding windows extract information from a fixed set of adjacent frames, while recurrent approaches accumulate information over time in a hidden state for exploitation at the current time step.

Window-based Earlier methods [16, 26, 1, 24] compute optical flow (OF) to warp adjacent frames for motion compensation with respect to the center frame. DUF [15] investigates VSR without explicit motion compensation by applying 3D-convolutions on a set of adjacent frames in combination with dynamic upscaling filters. Recent window-based designs often achieve higher performance in trade-off with runtime. Such a strategy has the benefit to use extensive parallel processing during training, which facilitates exploration of larger models. PFNL [33] adopts non-local residual blocks [29] as an alternative to motion estimation in order to progressively fuse information of adjacent frames. Contrary to other window-based methods, which fuse frames individually, RBPN [8] introduces a module to iteratively aggregate information from neighboring frames inside a fixed temporal window with recurrent back-projection. EDVR [28] proposes separate modules for alignment and fusion. Frames are aligned in feature space with cascaded deformable convolutions and fused by application of temporal and spatial attention maps. MuCAN [18] utilizes a hierarchical correspondence aggregation strategy to detect inter-frame correspondences by selecting a fixed set of the most similar patches after an exhaustive search on a local neighborhood. Aggregation from these selected patches is performed by a convolutional block. To adress the issue of misalignment, TGA [12] splits neighboring frames within a window into groups, according to their temporal distances from the center frame. Fusion is accomplished by application of attention maps.

Recurrent The temporal receptive field of approach (1) is limited in consequence of its fixed window size and usually depends on the availability of future frames, which introduces latency at inference. Approach (2) has a potentially unlimited temporal receptive field and generally accumulates information more efficiently with reuse of computation through a hidden state. Recurrent networks for super-resolution can be further divided into unidirectional [22, 6, 11, 13] and bidirectional methods [9, 2, 3].

Unidirectional: As one of the first, FRVSR [22] accounts for motion between consecutive frames in a recurrent fashion. The previous high-resolution estimate is warped towards the current frame with OF. Later, RLSP [6] introduced efficient propagation of implicit information in a hid-

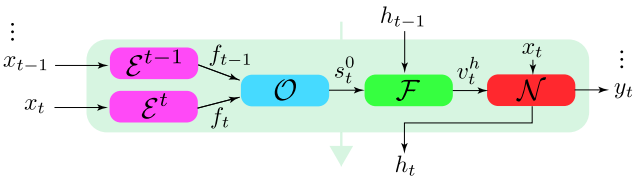

Figure 2: Schematic overview of our proposed method showing the interactions of our main modules in the recurrent cell.

den state with a fully convolutional recurrent network to VSR. The information in the hidden state is accumulated without explicit motion compensation. RSDN [11] further improves that concept by dividing the content into structure and detail components. Additionally, it accounts for error accumulation in the hidden state by detecting large displacements between frames.

Bidirectional: In order to leverage long-distance temporal correlations, an aggregation strategy considering information from all frames in a video is favorable and can be efficiently leveraged with forward and backward passes. While this approach is best suited for high performance, it violates causality – a necessary property for online inference. BasicVSR [2] achieves high performance with light recurrent cells by employing two passes over an entire video. Its successor BasicVSR++ [3] further improves performance with extensive bidirectional propagation strategies. Unfortunately, these approaches can not be evaluated online as they violate causality. Therefore, we design a unidirectional recurrent network for fast online inference and instead maximize information accumulation from the hidden state with our proposed efficient dynamic module.

Dynamic attention/transformer mechanisms are also explored in unrelated domains like object detection in images [34], video object segmentation [23] and to increase network capacity [4]. [34] proposes a transformer network that attends to dynamically predicted locations in order to detect relevant object features within a single image. Contrary, our proposed DAP leverages attention to efficiently match locations between consecutive frames. Additionally, our DAP leverages the discriminative feature of attention for more robust aggregation/fusion from the hidden state, where information is dynamically merged from multiple locations according to their relevance. [23] employs a top-k memory matching scheme to reduce computational overhead of its attention-based module for video object segmentation. DCN [4] uses dynamic ensemble learning over convolution kernels to increase network capacity. Our DAP is fundamentally different, instead predicting and attending to multiple spatial offset locations to explicitly match and fuse information across large spatial distances.

3. Method

3.1. Overview

According to Nyquist-Shannon's sampling theorem, the frequency band of a discrete signal is band-limited at a specific frequency f_N in the spectrum, called the Nyquist frequency. A VSR algorithm's task, is to recover the high-frequency content above said frequency from a low-resolution video $x \in \mathbb{R}^{T\times H\times W\times C}$, which is lost after subsampling its high-resolution counterpart $y \in \mathbb{R}^{T\times rH\times rW\times C}$ with scaling factor r. To fulfil the requirements for online VSR, an effective and efficient algorithm is necessary.

We propose a recurrent algorithm to address the two most important aspects of online VSR with strong emphasis on fast runtimes. Namely, the handling of misalignment between frames in combination with the update/extraction of information in the hidden state h_t. Such a setup allows efficient temporal aggregation of salient features from past frames $x_{0:t-1}$.

Recent advances in attention- and transformer architectures led to large performance gains in a wide range of computer vision tasks. However, due to pixel-dense processing requirements in VSR, a naive implementation of attention- or transformer type components is highly inefficient and prohibits the application of such operations due to their quadratic computational complexity. To alleviate this issue, we design an attention mechanism to dynamically predict a subset of relevant key/value pairs in the hidden state, omitting an exhaustive and expensive search over all possible pairs.

In particular, a recurrent cell propagates a pixel-dense hidden state h_t. Hidden state fusion and misalignment are simultaneously handled by our proposed deformable attention mechanism DAP. DAP uses a pyramid type network for dynamic offset prediction. First, our encoder network $\mathcal{E}$, individually divides frames x_t, x_{t-1} into multi-level feature maps f_t, f_{t-1} representing fine-to-coarse views on the input, effectively enlarging the receptive field and enriching representational power.

$$f_t, f_{t-1} = \mathcal{E}^t(x_t), \mathcal{E}^{t-1}(x_{t-1}) \tag{1}$$

Our deformable attention module $\mathcal{O}$ iteratively refines the calculated offsets s_t from coarse to fine.

$$s_t^0 = \mathcal{O}(f_t, f_{t-1}) \tag{2}$$

Our fusion module $\mathcal{F}$ then aggregates the hidden state features according to the final offsets.

$$v_t^h = \mathcal{F}(h_{t-1}, s_t^0) \tag{3}$$

After the fusion/alignment stage, our main processing unit $\mathcal{N}$, consisting of repeated residual information multi-distillation blocks [10], estimates the high-resolution frame

y_t and the next hidden state h_t.

$$[y_t, h_t] = \mathcal{N}(x_t, v_t^h), \quad t = 0, ..., T \tag{4}$$

A high-level overview of our method is shown in Fig. 2, which depicts the recurrent cell and the interactions of its main modules. Next, we will explain our proposed modules in more detail.

3.2. Deformable Attention Pyramid (DAP)

In order to fuse the accumulated past information from the hidden state h_{t-1} in relation to the current time step t, we employ a deformable attention pyramid. Our module operates on pixel-dense representations to adhere to the low-level processing requirements for VSR. We design our DAP to aggregate spatially displaced information in a robust and highly efficient manner. To achieve these properties we employ a pyramid type processing module working on encoded multi-level features computed from input frames x_{t-1} and x_t to efficiently enlarge the receptive field. Further, to avoid exhaustive correlation computations we restrict our attention module to a small set of key/value pairs at dynamically predicted offset locations in x_{t-1} for cross-attention with the current frame x_t.

First, offsets from x_{t-1} to x_t are computed. The offsets serve two purposes; (1) the handling of misalignment between frames and (2) a drastic minimization of attention weight computations. According to these offsets, the information is fused by cross-attention for exploitation at time step t. The final full-resolution offsets allow robust pixel-dense fusion through cross-attention between current frame x_t and hidden state h_{t-1}.

Multi-level Encoder It is essential in VSR to capture offsets across large distances since there can be fast motion in the camera or objects. As a solution to this problem, we use a multi-level encoder to obtain features in multiple resolutions. The coarser level features serve to capture larger motion due to a larger spatial view on the frames. We encode features for the last and current input frames x_{t-1} and x_t at levels $l = 0, ..., L$. To further enrich a frame's representation at each level, we encode it into higher d-dimensional features f_t^l. The smaller resolution representations at higher levels are obtained by repeated convolutional blocks $\mathcal{C}^l$, consisting of 4 convolutions, with intermediate bilinear downsampling steps between blocks ($\times 2$) which we denote with the operator $D_\downarrow(\cdot)$, see Eq. 5. Even with small 3×3 kernels, such a strategy increases the receptive field exponentially through the repeated downsampling operations. We employ individual processing chains for both input frames x_{t-1} and x_t, we set $L = 3$.

$$\begin{aligned} f_t^l &= D_\downarrow\left(\mathcal{C}^l\left(f_t^{l-1}\right)\right), \quad f_t^0 = \mathcal{C}^0\left(x_t\right), \quad l = 0, ..., L \\ f_{t-1}^l &= D_\downarrow\left(\mathcal{C}^l(f_{t-1}^{l-1})\right), \quad f_{t-1}^0 = \mathcal{C}^0\left(x_{t-1}\right), \quad l = 0, ..., L \end{aligned} \tag{5}$$

Deformable Attention In order to significantly reduce the complexity of our attention module, we confine the search for salient features to dynamically selected locations in the feature maps, instead of an exhaustive correlation computation over a large neighborhood or even the whole frame. The quadratic component prevalent in attention mechanisms is overcome by applying pure cross-attention towards the current frame x_t, resulting in a linear dependency in the number of key/value pairs. In particular, we process the encoded features f_t^l, f_{t-1}^l with our optimized, light-weight, deformable attention operation mechanism at each pyramid level l. We largely reduce the computational effort by merely computing pixel-dense correlations between embedded features representing the queries Q_t^l of current frame x_t, and key/value pairs K_{t-1}^l/V_{t-1}^l sampled at k dynamically predicted spatial locations $s^l \in \mathbb{R}^{H/2^l \times W/2^l \times 2k}$ in f_{t-1}. Queries and key/value pairs are linearly embedded with parameters W_Q^l, W_K^l and W_V^l. We apply scaled dot-product attention with softmax to aggregate the values from $V_{t-1}^{l^T}$. We account for resolution mismatch between feature maps at level l and sampling locations s^{l+1} from the previous level $l+1$ with bilinear upsampling ($\times 2$), denoted by $U_\uparrow(\cdot)$. The corresponding equations are presented in Eq. 6, an illustration is provided in Fig. 3. Please note the reverse order for pyramid level index l, since the processing is performed from coarse to fine.

$$\begin{aligned} l &= L, ..., 0 \\ Q_t^l &= W_Q^l f_t^l \\ K_{t-1}^l &= W_K^l \mathcal{S}(f_{t-1}^l, U_\uparrow(s_t^{l+1})) \\ V_{t-1}^l &= W_V^l \mathcal{S}(f_{t-1}^l, U_\uparrow(s_t^{l+1})) \\ v_t^l\left(Q_t^l, K_{t-1}^l, V_{t-1}^l\right) &= \text{softmax}\left(\frac{Q_t^l K_{t-1}^{l^T}}{\sqrt{d}}\right) V_{t-1}^l \end{aligned} \tag{6}$$

Iterative Refinement We propose an efficient iterative coarse-to-fine scheme to address blending of multi-level offset representations in different resolutions, attention-aggregated values v_t^l, and features f_t^l from the current frame x_t. In each pyramid level, the dense offsets $s_t^l \in \mathbb{R}^{H/2^l \times W/2^l \times 2k}$ are iteratively refined by adding residual offsets to the previous level's offsets s_t^{l+1} with a light-weight convolutional block $\mathcal{C}_\mathcal{S}^l$. Our offset prediction network $\mathcal{C}_\mathcal{S}^l$ uses large 7×7 kernels to ensure dense computation with a large receptive field, in contrast to smaller 3×3 kernels employed in all our other modules.

$$\begin{aligned} &l = L, ..., 0 \\ &s_t^l = \mathcal{C}_\mathcal{S}^l\left(f_t^l, v_t^l, U_\uparrow(s_t^{l+1})\right) + U_\uparrow(s_t^{l+1}), \quad s_t^L = \mathcal{C}_\mathcal{S}^L\left(f_t^L\right) \end{aligned} \tag{7}$$

Hidden State Fusion Ultimately, the top level offsets s_t^0 serve to fuse salient hidden state features for exploitation at

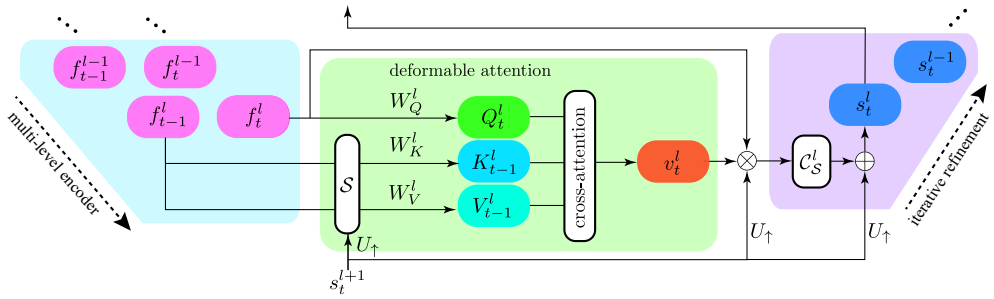

Figure 3: Deformable attention pyramid (DAP). First we calculate multi-level features from x_t, and x_{t-1} using a U-Net [21] type encoder. In each pyramid level l, k sampling locations $s_t^l \in \mathbb{R}^{H/2^l \times W/2^l \times 2k}$ are calculated per pixel to serve as key/value locations in the upper level's deformable attention block. Features in the upper level are fused from $t-1$ towards t according to s_t^l with cross-attention, before calculating the residual offsets with convolutional block $\mathcal{C}_S^l$ with respect to the lower layer $l+1$. Offsets s_t^l are refined iteratively until the obtained locations s_t^0 at level $l=0$ are finally employed to perform cross-attention fusion in the hidden state h_{t-1}.
$\otimes$: concatenation in channel dimension, $\oplus$: element-wise addition.

time step t. For that matter, another deformable attention block v_t^h takes care of fusion in full resolution by leveraging the computed offsets s_t^0. Since it is critical to minimize the computational effort for fast VSR, our DAP module is processing frames in a lower d-dimensional space ($d=8$), because the frames' channels are fixed in size setting a limit on available information. Conversely, a larger channel size in our main processing pipeline – the size of the hidden state – increases the upper limit of storable information, which facilitates higher performance. Thus, the hidden state's deformable attention block v_t^h performs query/key matching in d-dimensional embeddings, while the values are embedded and aggregated in their native high-dimensional space. We denote our network by DAP-n, n represents the feature size in the main processing block.

$$v_t^h\left(Q_t^0, K_{t-1}^h, V_{t-1}^h\right) = \text{softmax}\left(\frac{Q_t^0 {K_{t-1}^h}^T}{\sqrt{d}}\right) V_{t-1}^h \tag{8}$$

A significant improvement in runtime is achieved by grouped sampling inside tensors at all DAP stages, which are omitted in the notation for clarity. The number of groups is set according to the number of sampled key/value pairs $k=4$.

4. Experiments

We conduct comprehensive experiments in our ablation study to highlight our key innovations and compare our best performing configuration [1] to state-of-the-art methods on 3 diverse standard benchmarks REDS [19], UDM10 [33] and Vimeo-90K [32] with 2 different subsampling kernels (Bicubic and Gaussian). We use the proposed split for REDS according to [28] along with the provided training pairs from [19]. Our results for Vimeo-90K and UDM10 test sets are obtained by application of a Gaussian low-pass filter followed by resampling of every 4-th pixel along each spatial dimension. Following the literature we set the Gaussian filter's standard deviation and kernel size to $\sigma=1.6$ and 13 respectively, Vimeo-90K serves as training set for both test sets. During training we uniformly crop sequences with spatial size 256×256 (high resolution) and adopt random flips, rotations and temporal inversion to augment the data. The initial learning rate is set to 10^{-4} and is reduced after reaching a plateau in two steps to 0.5×10^{-4}, then 10^{-5}. To stabilize training we use gradient clipping. More specific details will be explained in the respective subsections. Similar to recent proposals in the literature our networks are trained with a smooth version of L1 loss [2], which showed benefits over L2 loss for super-resolution as it is less sensitive to outliers. Our model can be trained end-to-end without relying on pretrained modules or external data. We use Adam optimizer [17] and set the scaling factor to $r=4$ in all our experiments.

4.1. Ablation

To highlight our key contributions, we perform a comprehensive ablation study, where a comparison between different configurations show the benefits of our proposed modules, see Tab. 1. We use REDS for training and evaluation. During training we collect batches composed of $b=32$ samples with crop size $T\times H\times W\times C=5\times64\times64\times3$. We provide both validation and test set results to emphasize the robustness of our ablation study, but restrict the detailed discussion to REDSval.

Modules Configuration 1 is trained without any motion handling and fusion, we only employ the main module $\mathcal{N}$

[1] For code and other material refer to `https://github.com/dariofuoli/DAP`.

[2] PyTorch's torch.nn.SmoothL1Loss() with $\beta=10^{-2}$

Configuration	Offsets	Pyramid	Attention	Features	REDS4val [19] (Y)	REDS4 [19] (RGB)
1				64	28.77/0.7906	28.59/0.8155
2			✓	64	28.95/0.7926	28.69/0.8184
3	✓			64	29.82/0.8194	29.50/0.8461
4	✓	✓		64	30.07/0.8264	29.66/0.8507
5	✓	✓	✓	64	30.36/0.8341	29.97/0.8571
6	✓	✓	✓	128	30.77/0.8440	30.49/0.8676

Table 1: Ablation study on REDS (PSNR/SSIM). All models are trained in the same settings on sequences of 5 frames. Red denotes best, blue denotes second best.

to propagate a hidden state. The large performance drop compared to all configurations with offsets validates the importance of handling motion and misalignment in VSR. It clearly shows the downsides of naive convolution-based networks for video processing and promotes the necessity for other mechanisms. Similar conclusions can be drawn from adding our attention-based fusion module $\mathcal{F}$ without providing offsets. Still, a slight gain of 0.18dB can be achieved by our fusion module even without offsets. The addition of offsets to configuration 1 significantly improves performance by 1.05dB in configuration 3. Furthermore, a larger receptive field is attained by application of a pyramid refinement mechanism with simple convolutional fusion instead of our proposed deformable attention. Thus, configuration 4 improves PSNR by 0.25dB. Our complete setup with all our proposed modules in combination further boosts performance by 0.29dB (configuration 5). Moreover, increasing the feature dimension from 64 to 128 adds another 0.41dB. Hence, our proposed DAP network realizes large gains for VSR. As an aside, in addition to inferior performance of configuration 4, which relies on offset prediction and convolution without our proposed attention mechanism, we observed instabilities during training, which leads us to the conclusion that attention stabilizes training of DNN's in combination with offset prediction.

Configuration	$\overrightarrow{\text{DAP-64}}$	$\overleftarrow{\text{DAP-64}}$	$\overrightarrow{\text{DAP-128}}$	$\overleftarrow{\text{DAP-128}}$
REDS [19]	29.97/0.8571	**30.16/0.8635**	30.49/0.8676	**30.72/0.8751**

Table 2: Forward/Reverse ($\rightarrow$/$\leftarrow$) evaluation on REDS4 test set. We evaluate the same model in both directions.

Reverse Evaluation A core feature of state-of-the-art bidirectional methods is their ability to fuse all information over an entire video offline. Further, this strategy includes aggregation in reverse time order, which may have additional benefits in certain cases. Window-based methods usually aggregate information from future frames with similar potential advantages. We analyze the effects of relative motion patterns induced by time reversal also in our online setting, since such motion can appear even in non-reversed video, *e.g*. objects moving away from the camera or a camera that is zooming out. Therefore, we investigate the difference between forward and backward evaluation, *i.e*. we evaluate the sequences on the REDS test set in both temporal directions in Tab. 2. Surprisingly, reverse time order aggregation in these videos increases performance significantly, *i.e*. by $+0.19$dB and $+0.23$dB for DAP-64 and DAP-128 respectively. After inspection, we attribute this gain to forward camera motion being more prevalent in these videos. If objects are moving towards the camera, or vice versa, in reverse time order they first appear in higher resolution, simplifying super-resolution for these objects. Thus, having the opportunity to process video in reverse or having access to future frames, may substantially improve performance for VSR depending on the content, resulting in additional advantages for non-causal methods compared to online algorithms.

4.2. Comparison with State of the Art

We compare the performance of our method to state-of-the-art methods on 3 different datasets with diverse properties, see Tab. 3. Since we address the *causal* VSR problem, we do not compare to bidirectional methods. Such methods cannot be evaluated in a single pass, which inhibits their application for online processing. Additionally, they have an incomparable advantage as a consequence of access to all frames from a video sequence. However, we still report the results of prominent bidirectional methods [2, 3] for reference in Tab. 3.

REDS is a challenging high-resolution (720×1280) dataset, because large displacements and a non-stabilized camera complicate temporal aggregation. UDM10 (720×1272) on the other hand contains more steady camera motion and continuous movement. Vimeo-90K contains short sequences of only 7 frames for training and testing in small resolution (256×448). The dataset was released with window-based evaluation in mind, *i.e*. only the center frame is expected to be restored, which impedes a fair comparison to our recurrent method. To improve comparability we reflect the sequences at the end for 3 frames to compute the metrics on the last frame representing the center frame. Yet, our method still has a disadvantage due to the pressure of estimating each frame up to the end of the sequence, contrary to aggregation from adjacent frames only.

REDS For comparison with state-of-the-art methods on REDS we extend our training sequences to facilitate learning of longer temporal dependencies. We uniformly crop

Method	Unid.	Onl.	R-T.	Run [ms]	fps [1/s]	FLOPs [G]	MACs [G]	REDS4[19] PSNR/SSIM	UDM10[33] PSNR/SSIM	Vimeo-90K[32] PSNR/SSIM
Bicubic	✓	✓	✓	-	-	-	-	26.14/0.7292	28.47/0.8253	31.30/0.8687
TOFlow [31]	✓	✗	✗	-	-	-	-	27.98/0.7990	36.26/0.9438	34.62/0.9212
FRVSR [22]	✓	✓	✗	*137	*7.3	-	-	-	37.09/0.9522	35.64/0.9319
DUF [15]	✓	✗	✗	*974	*1.0	-	-	28.63/0.8251	38.48/0.9605	36.87/0.9447
RBPN [8]	✓	✓	✗	*1507	*0.7	-	-	30.09/0.8590	38.66/0.9596	37.20/0.9458
PFNL [33]	✓	✗	✗	*295	*3.4	-	-	29.63/0.8502	38.74/0.9627	-
MuCAN [18]	✓	✗	✗	2'208	0.5	15'853.2	7'922.8	30.88/0.8750	-	-
EDVR-M [28]	✓	✗	✗	116	8.6	925.7	462.3	30.53/0.8699	39.40/0.9663	37.33/0.9484
EDVR [28]	✓	✗	✗	348	2.9	4'037.3	2'017.3	31.09/0.8800	39.89/0.9686	37.81/0.9523
TGA [12]	✓	✗	✗	427	2.3	-	-	-	-	37.59/0.9516
RSDN [11]	✓	✓	✗	63	15.9	713.2	356.3	-	39.35/0.9653	37.23/0.9471
RRN [13]	✓	✓	✓	28	35.7	387.5	193.6	-	38.96/0.9644	-
RLSP [6]	✓	✗	✓	30	33.3	503.7	251.8	-	38.48/0.9606	36.49/0.9403
DAP-128 (ours)	✓	✓	✓	38	26.3	330.0	164.8	30.59/0.8703	39.50/0.9664	37.29/0.9476
BasicVSR [2]	✗	✗	✗	82	12.2	754.3	376.7	31.42/0.8909	39.96/0.9694	37.53/0.9498
IconVSR [2]	✗	✗	✗	100	10.0	904.9	451.9	31.67/0.8948	40.03/0.9694	37.84/0.9524
BasicVSR++ [3]	✗	✗	✗	110	9.1	837.1	418.1	32.39/0.9069	40.72/0.9722	38.21/0.9550

Table 3: Comparison with state of the art. We compare runtime, frames per second (fps), FLOPs, MACs and PSNR/SSIM metrics on 3 standard benchmarks. Additionally, we denote if a method is unidirectional, *i.e.* if it can generate output in a single pass (Unid.), can be evaluated strictly online (Onl.), *i.e.* no future frames are needed, and if it can produce video (720p) in real-time (R-T.). All PSNR/SSIM results and runtime measurements marked with * are reported from the respective papers. All other methods are profiled (Run/fps/FLOPs/MACs) in the same settings on a NVIDIA RTX2080Ti by us. Red denotes best, blue denotes second best.

sequences of length $T = 15$ with a reduced batch size of $b = 8$, as a result of higher memory demand. To avoid expensive training from randomly initialized parameters, our model is initialized with the pre-trained weights of configuration 6 from the ablation study and refined for $T = 15$. Training on a larger sequence length T further boosts our performance on REDS by 0.1dB and 0.0027 in PSNR and SSIM respectively. With the exception of large and slow models EDVR and MuCAN we significantly surpass all other models in performance with high speed, we even supersede EDVR-M with a reduction in runtime of over $\times 3$ and largely reduced computational demand. DAP-128 impressively handles the complex motion in REDS with only 38ms per frame. Our method is capable of producing over 24 fps needed for real-time evaluation with the lowest computational complexity among all methods (330.0 GFLOPs/164.8 GMACs).

Vimeo-90K/UDM10 As already mentioned, Vimeo-90K has limits due to its intended evaluation protocol. Nevertheless, DAP-128's performance on Vimeo-90K is comparable with recurrent state-of-the-art method RSDN and window-based EDVR-M, despite EDVR-M's advantage on Vimeo-90K being a window-based method. Higher performance is expected for window-based EDVR and TGA as a consequence of larger capacity and aforementioned advantages in evaluation on Vimeo-90K. On the contrary, UDM10 defines a standard evaluation strategy, more suitable for a realistic and fair comparison. We achieve the second best performance in PSNR with high speed, EDVR is over 9-times slower with a huge computational complexity overhead. Due to our highly efficient aggregation strategy in our proposed DAP, we significantly surpass recurrent method RSDN both in terms of performance, runtime and computational demand. We largely improve performance over RRN with a gain of $+0.54$dB, lower computational complexity and only slightly increased runtime.

Runtime and Computational Complexity We set out to design an algorithm to overcome the challenges of online VSR. On top of limitations in temporal information aggregation (only past information available), the fulfilment of hard time constraints and low computational complexity is crucial for this task. With our proposed DAP and overall efficient network design, we achieve the best performance in relation to computational effort and are able to move the Pareto front. DAP-128 is a high-speed method with the lowest computational demand among all other methods in Tab. 3 (330.0 GFLOPs/164.8 GMACs), it is able to reach real-time evaluation speed with over 26 frames per second and high performance. Our network design surpasses the fundamental algorithmic design used by EDVR and other window-based methods, since our method achieves better performance overall with much reduced computational complexity and therefore faster runtimes, *e.g.* EDVR-M vs. DAP-128. Note, that EDVR's and other window-based

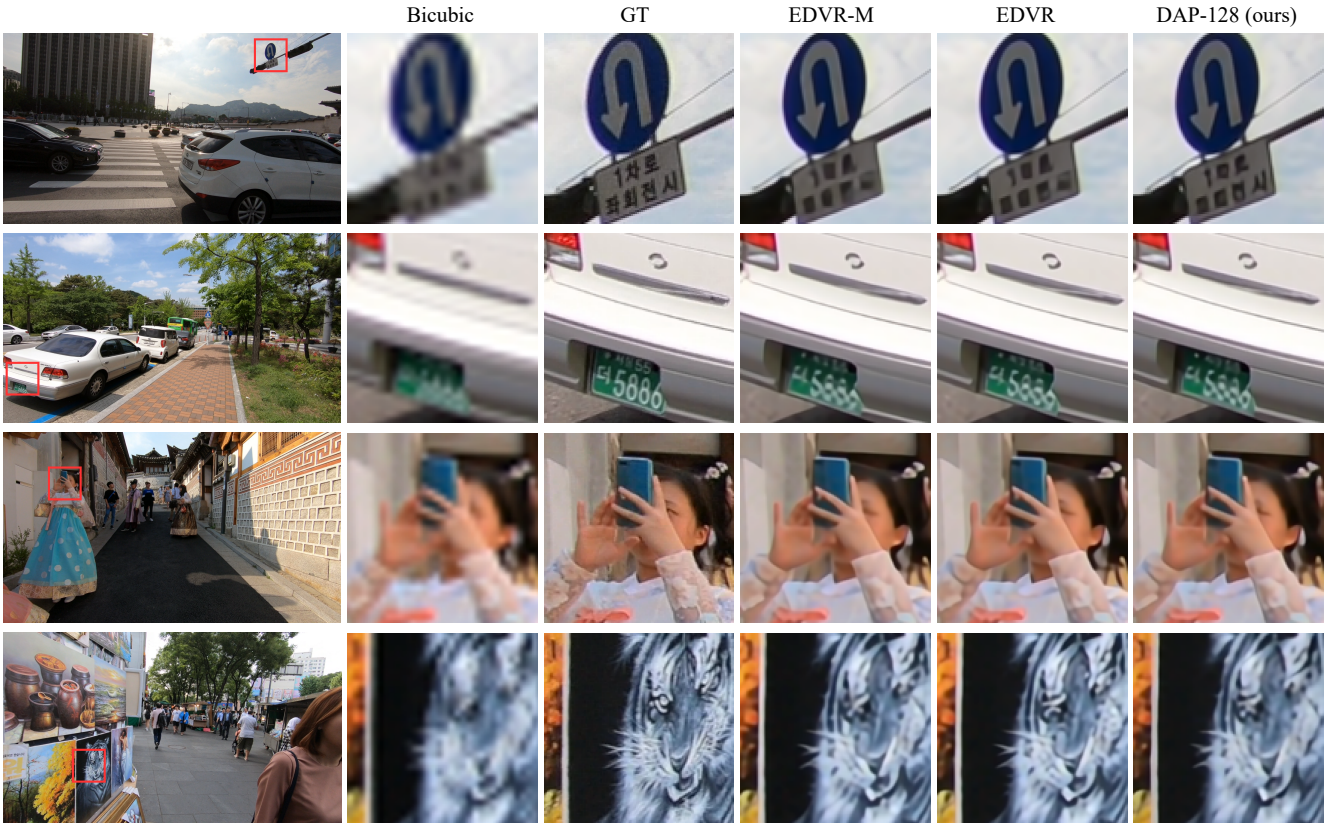

Figure 4: Visual examples on REDS. Our method achieves competitive performance compared to EDVR-M and EDVR with significantly reduced computational effort.

method's runtime measurements do not account for online evaluation due to the opportunity to process windows in parallel offline. Thus, online evaluation most likely leads to higher runtimes and latency in practice. We also surpass the design of state-of-the-art recurrent networks like RSDN with a significant speed-up and over 2-times reduced computational demand. For an illustration of performance vs. runtime and complexity (number of parameters) please refer to Fig. 1.

Visual Examples We provide visual examples for qualitative evaluation in Fig. 4. We compare our method against Bicubic interpolation as a baseline, state-of-the-art method EDVR, its lighter version EDVR-M and the ground truth (GT) on all 4 REDS test sequences. Our method produces high quality frames in accordance to the PSNR/SSIM evaluation in Tab. 3. However, there are individual strengths and weaknesses among all methods. The signals in the first row are restored with higher quality than both EDVR-M and even its heavier version EDVR. On the other hand, the tiger in row 4 is restored in more detail by EDVR. As explained in Sec. 4.1, access to future frames can be highly advantageous. EDVR and most other window-based methods access future frames in their window. As a consequence of forward camera motion, the tiger appears in higher resolution in future frames in this particular scene.

5. Conclusion

We address the two main challenges in online VSR; efficient temporal aggregation and misalignment. Despite the inherent relationship between computational complexity and network capacity, our light-weight designs enable high performance with fast runtimes in the online setting, which is achieved by our effective attention-based module for combined fusion/alignment of information from the hidden state only. In contrast to other attention-based solutions for VSR, our proposed DAP avoids exhaustive operations by dynamically attending to the salient locations in the hidden state, thereby significantly reducing the high computational burden associated with attention and transformers. Our attention mechanism enables efficient pixel-dense processing, a crucial feature for super-resolution. Comprehensive experiments and ablation studies reinforce our contributions and provide analysis of our method. We surpass state-of-the art method EDVR-M on two standard benchmarks with a speed-up of over $3\times$ and the lowest computational complexity among all compared methods.

Acknowledgements. This work was partly supported by a Huawei Technologies Co. Ltd project, the ETH Zürich Fund (OK) and the Alexander von Humboldt Foundation.

References

[1] Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[2] Kelvin CK Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Basicvsr: The search for essential components in video super-resolution and beyond. *arXiv preprint arXiv:2012.02181*, 2020.

[3] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. *arXiv preprint arXiv:2104.13371*, 2021.

[4] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[6] Dario Fuoli, Shuhang Gu, and Radu Timofte. Efficient video super-resolution through recurrent latent space propagation. In *ICCV Workshops*, 2019.

[7] Dario Fuoli, Shuhang Gu, Radu Timofte, et al. Aim 2019 challenge on video extreme super-resolution: Methods and results. In *ICCV Workshops*, 2019.

[8] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Recurrent back-projection network for video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[9] Yan Huang, Wei Wang, and Liang Wang. Bidirectional recurrent convolutional networks for multi-frame super-resolution. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[10] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the 27th ACM International Conference on Multimedia (ACM MM)*, pages 2024–2032, 2019.

[11] Takashi Isobe, Xu Jia, Shuhang Gu, Songjiang Li, Shengjin Wang, and Qi Tian. Video super-resolution with recurrent structure-detail network. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 645–660, Cham, 2020. Springer International Publishing.

[12] Takashi Isobe, Songjiang Li, Xu Jia, Shanxin Yuan, Gregory Slabaugh, Chunjing Xu, Ya-Li Li, Shengjin Wang, and Qi Tian. Video super-resolution with temporal group attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[13] Takashi Isobe, Fang Zhu, and Shengjin Wang. Revisiting temporal modeling for video super-resolution. In *BMVC*, 2020.

[14] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[15] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[16] Armin Kappeler, Seunghwan Yoo, Qiqin Dai, and Aggelos K Katsaggelos. Video super-resolution with convolutional neural networks. In *IEEE Transactions on Computational Imaging*, 2016.

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[18] Wenbo Li, Xin Tao, Taian Guo, Lu Qi, Jiangbo Lu, and Jiaya Jia. Mucan: Multi-correspondence aggregation network for video super-resolution. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 335–351, Cham, 2020. Springer International Publishing.

[19] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[20] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[21] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[22] Mehdi S. M. Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-Recurrent Video Super-Resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[23] Hongje Seong, Seoung Wug Oh, Joon-Young Lee, Seongwon Lee, Suhyeon Lee, and Euntai Kim. Hierarchical memory matching network for video object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[24] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR*, 2016.

[25] Sanghyun Son, Suyoung Lee, Seungjun Nah, Radu Timofte, and Kyoung Mu Lee. Ntire 2021 challenge on video

super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 166–181, June 2021.

[26] Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. Detail-revealing deep video super-resolution. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[28] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[29] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[30] Zeyu Xiao, Xueyang Fu, Jie Huang, Zhen Cheng, and Zhiwei Xiong. Space-time distillation for video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2113–2122, June 2021.

[31] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision (IJCV)*, 127(8):1106–1125, 2019.

[32] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision (IJCV)*, 127(8):1106–1125, 2019.

[33] Peng Yi, Zhongyuan Wang, Kui Jiang, Junjun Jiang, and Jiayi Ma. Progressive fusion video super-resolution network via exploiting non-local spatio-temporal correlations. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3106–3115, 2019.

[34] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.