Signal, Image and Video Processing (2023) 17:4415–4424
https://doi.org/10.1007/s11760-023-02675-z

ORIGINAL PAPER

Multi-frame spatio-temporal super-resolution

Zahra Gharibi[1] · Sam Faramarzi[2]

Received: 25 March 2023 / Revised: 18 May 2023 / Accepted: 15 June 2023 / Published online: 31 July 2023

Abstract

Increasing the resolution of digital images and videos using digital super-resolution (SR) techniques has been of great interest in industry and academia over the past three decades. Most SR methods target improving only the spatial resolution of images and videos, whereas improving the temporal resolution could be more critical for some videos. Motion blur is a temporal artifact by nature, so removing it using spatial SR techniques would be highly challenging and often unsuccessful. This paper proposes a multi-frame motion-based video super-resolution method to increase both spatial and temporal resolutions of a single input video. Our optimization problem is based on a maximum a posteriori estimator that estimates each high-resolution (HR) frame by fusing multiple low-resolution frames. The form of the image prior used in the optimization framework is based on the assumption that natural HR frames are piecewise smooth. We introduce a new method to enhance the sharpness of edges in the video frames during the optimization process. We also involve a temporal constraint that improves temporal consistency in the estimated video. Moreover, we propose a new scheme for motion estimation that better suits video frame rate upsampling. Our results are compared with state-of-the-art SR methods, including ML-based ones, which confirm the effectiveness of the proposed method.

Keywords Video restoration · Space–time super-resolution · Global Optimization · Probability modeling · Maximum a posteriori (MAP) estimator

1 Introduction

Digital super-resolution (DSR or briefly SR) is a set of image restoration techniques aiming to increase spatial and/or temporal resolutions in images and videos, mainly using a controlled optimization procedure. For an imaging system, spatial resolution refers to the finest detail visually distinguishable in captured images. In contrast, temporal resolution defines the highest frequency of dynamic events perceivable in a video sequence. SR has been a very active research area in both academia and industry over the past three decades. It has found practical applications in many real-life problems, including the display industry, medical imaging, satellite and aerial photography, astronomy, surveillance, and remote sensing.

There are two categories of SR techniques in the literature: multi-frame SR (MFSR) and single-frame SR (SFSR), where frame means a still image or a video frame. MFSR mainly refers to the traditional way of doing SR, which reconstructs a high-resolution (HR) frame by fusing multiple low-resolution (LR) frames [1–3]. Each LR frame must have information not present in other LR frames. This condition is fulfilled by the existence of subpixel motion (globally or locally) between the LR frames, which commonly happens in most captured frame sequences due to the movement of objects in the scene or the camera. Most MFSR techniques in the literature assume that the motion is global, and the blur function is known a priori. However, a few ones allow for local motion in their model and estimate both motion and blur along with the HR frames [4, 5].

Learning-based SR (LBSR) techniques reconstruct an HR frame from a single LR frame. This SR category of techniques assumes that the relationship between the LR and HR frames can be learned from a training set that contains several LR frames and their corresponding HR

✉ Zahra Gharibi
zgharibi@csusm.edu

Sam Faramarzi
samfar2454@gmail.com

[1] Department of Operations, and Supply Chain Management, Califonia State University San Marcos, San Marcos, CA, USA

[2] San Marcos, CA, USA

frames. In a traditional SFSR approach called patch-based or dictionary-based SR [6–8], an LR input frame is segmented into small patches. Then each patch is compared against the LR patches in the training set to find its best match. Finally, an input LR patch is replaced with the corresponding HR patch of its best match. SFSR is also developed for videos where an LR video is segmented into spatio-temporal patches [9]. Machine-learning (ML) and Deep Learning (DL)-based SFSR methods [10–20], including Convolutional Neural Networks (CNN) and Generative Adversarial Nets (GAN) based techniques, have been of much interest in recent years. Learning-based MFSR methods are also introduced for videos by leveraging the temporal correlation between video frames for more accurate reconstruction [9, 10, 21–24]. However, in this work, we refer to the traditional (none-ML) motion-based MFSR techniques simply as MFSR to differentiate them from the LBSR techniques.

The majority of SR publications aim to improve only the spatial resolution in images/videos. Nevertheless, space—time (or spatio-temporal) super-resolution (STSR) for improving both spatial and temporal resolutions is considered in very few publications. One approach for STSR is to use multiple LR video sequences having spatial (sub-pixel) as well as temporal (sub-frame) misalignments [25–27]. It means that the corresponding pixels in the input videos' frames are in different spatial locations due to the scene's movements, and the videos are captured in slightly different timestamps. Here, for simplicity in modeling the motion, the capturing cameras are kept close to each other compared to their distances from the scene. This constraint enables the motion between the videos to be globally modeled as a 2D homography transformation in space and a 1D affine transformation in time [28].

STSR from a single video is also proposed in several works [9]. proposes a patch-based approach assuming that in a natural video, space–time patches recur many times inside the same video at different spatio-temporal scales. This method is effective on videos having a repeated act like a rotating turbine [29]. proposes a 3D steering kernel regression method to fuse the frames without an explicit motion estimation. However, they employ a suboptimal imaging model to first estimate the upsampled output frames without deburring and then apply deblurring to each output frame individually. A few DL-based STSR methods are also proposed [10–12], but they mainly target frame interpolation to increase the video frame rate. A one-stage space–time video SR method is introduced in [11] to increase the spatial resolution and temporal frame rate using a frame feature temporal interpolation and a deformable ConvLSTM recurrent model.

We propose in this paper an STSR method from a single video using an MFSR approach. It takes an LR video as input and reconstructs an HR video with a larger frame size and/or a higher number of frames. The proposed technique improves an input video's spatial and temporal resolutions by combining each video frame with its adjacent frames. For this purpose, we extend the sequential motion estimation approach introduced in [5] to support temporal upsampling. The optimization framework is based on a maximum a posteriori (MAP) statistical framework that applies the desired level of smoothness while restoring sharp edges in the estimated HR frames. We introduce a sharpening process embedded in the optimization framework to intensify the recovered edges. Furthermore, we improve the temporal consistency by adding a temporal constraint between the current and previous reconstructed frames. We compare our proposed STSR method's performance with a few SR methods, including deep learning-based ones.

It should be noted that the proposed method can remove/reduce spatial blur, temporal blur, spatial aliasing, and noise in video sequences. It can also increase the video frame rate and perform view interpolation. However, it does not address the removal of temporal aliasing resulting from very fast dynamic events. Removing the temporal aliasing is mainly done by capturing multiple videos with temporal misalignments [25, 30].

The rest of this paper is organized as follows: Sect. 2 demonstrates our problem formulation, with subsections that discuss our assumed STSR imaging model, the proposed optimization framework, the extended motion estimation method, the initial estimate of the HR frame, and our strategy of color processing. The experimental results are presented in Sect. 3, and finally, Sect. 4 concludes the work of this paper.

2 Problem formulation

2.1 Imaging model

Although the forward imaging model represented in this section is similar to those used in other MFSR methods, it is extended to include both spatial and temporal resolution improvements. Here, the input in the image domain is a four-dimensional (4D) LR video $g(x_l, y_l, c, t_l)$ of size $W \times H \times C \times T$, where $x_l \in [0, W-1]$ and $y_l \in [0, H-1]$ are spatial pixel coordinates, $c \in [0, C-1]$ is the color channel, and $t_l \in [0, T-1]$ is the frame number. Here, W is the frame width, H is the frame height, C is the number of color channels (1 for gray-scale and 3 for color videos), and T is the number of frames. The output would be a 4D HR video $f(x_h, y_h, c, t_h)$ of size $rW \times rH \times C \times sT$, where r and s are the scaling (upsampling) factors for space and time domains. So the frame dimensions and the frame rate of the input video would increase by factors of r and s, respectively.

For simplicity, we represent our formulation in the vector–matrix notation where the input and output videos are vectors in lexicographical order, of sizes $WHCT \times 1$

and$r^2sWHCT \times 1$, and shown in bold lower-case letters as $\boldsymbol{g}$ and$\boldsymbol{f}$, respectively. The i th frame of the output video $\boldsymbol{f}_i$ (of size$r^2WHC \times 1$) is estimated from the input frames$\{\boldsymbol{g}_{j-a}, \ldots, \boldsymbol{g}_j, \ldots, \boldsymbol{g}_{j+b}\}$, where $j = \lfloor i/s \rfloor$.[1] In other words, our proposed framework combines a+b+1 adjacent frames of the input LR video around the center frame $\boldsymbol{g}_j$ (of size$WHC \times 1$) to reconstruct the output frame$\boldsymbol{f}_i$, where a and b are the number of adjacent LR frames in the backward and forward directions, respectively.[2] We use the following linear imaging model to relate the ith HR frame to the k th LR frame:

$$\boldsymbol{g}_k = \boldsymbol{D}\boldsymbol{B}_k\boldsymbol{M}_{j,k}\boldsymbol{f}_i + \boldsymbol{n}_k, \quad j = i/s, \quad k \in [j-a, \quad j+b] \tag{1}$$

where $\boldsymbol{M}_{j,k}$ is the motion matrix that models warping (registration) from $\boldsymbol{f}_i$ to $\boldsymbol{g}_k$, $\boldsymbol{B}_k$ is the spatio-temporal blur matrix, $\boldsymbol{D}$ is the spatio-temporal downsampling matrix, and $\boldsymbol{n}_k$ is the noise vector. According to this model, an HR frame $\boldsymbol{f}_i$ is warped, blurred, and downsampled in both space and time, then added up with noise to form an LR frame $\boldsymbol{G}_k$. The motion matrix represents the movement of the scene's objects and the camera between two frames. The 3D blur kernel is the overall effect of the camera's and objects' movements, defocus, depth of field, optical and sensor blurs, and exposure time. Downsampling is the outcome of capturing the scene in discrete spatial positions (pixels) and temporal timestamps, dictating the camera's frame resolution and framerate.

2.2 Proposed STSR framework

We use a maximum a posteriori (MAP) framework to estimate an HR frame given a few neighboring LR frames:

$$\boldsymbol{f}_i = \arg\max_{X_i} \prod_k \Pr(\boldsymbol{f}_i | \boldsymbol{g}_k), \quad j = i/s, \quad k \in [j-a, \quad j+b] \tag{2}$$

Using the Bayes rule, this can be alternatively written as:

$$\boldsymbol{f}_i = \text{arg}\underset{\boldsymbol{X}_i}{max} \prod_k \frac{\Pr(\boldsymbol{g}_k | \boldsymbol{f}_i)\Pr(\boldsymbol{f}_i)}{\Pr(\boldsymbol{g}_k)} \tag{3}$$

where $\Pr(\boldsymbol{g}_k | \boldsymbol{f}_i)$ is the likelihood (a.k.a. data fidelity or data fusion term), $\Pr(\boldsymbol{f}_i)$ is the prior on the HR frame (a.k.a. regularization term), and $\Pr(\boldsymbol{g}_k)$ is the evidence of the LR frame. The denominator in (3) can be ignored because it is not a function of $\boldsymbol{f}_i$. Moreover, since the nominator's densities have exponential forms, it would be simpler to minimize the minus log of the functional in (3) equivalently. This yields:

$$\boldsymbol{f}_i = \text{arg}\underset{\boldsymbol{X}_i}{min} \left\{ \sum_k -\log\left[\Pr(\boldsymbol{g}_k | \boldsymbol{f}_i)\right] - \log\left[\Pr(\boldsymbol{f}_i)\right] \right\} \tag{4}$$

Assuming the noise to be white Gaussian, $-\log\left[\Pr(\boldsymbol{g}_k | \boldsymbol{f}_i)\right]$ in (4) would be proportional to the energy of noise, i.e. $\|\boldsymbol{D}\boldsymbol{B}_k\boldsymbol{M}_{jk}\boldsymbol{f}_i - \boldsymbol{g}_k\|_2^2$ which is the sum of squared differences (SSD) between the simulated and observed LR frames. The operator $\|\cdot\|_2^2$ denotes the square of norm-2, which is defined for a vector $\boldsymbol{A}$ with elements a_i as $\|\boldsymbol{A}\|_2^2 = \boldsymbol{A}^T\boldsymbol{A} = \sum a_i^2$ where $\boldsymbol{A}^T$ is the transpose of $\boldsymbol{A}$.

Natural HR images are not globally smooth but mostly piecewise-smooth, as they consist of smooth regions surrounded by sharp edges. An appropriate form for the regularization term based on such observation would penalize high-energy variations much less than norm-2 in the reconstructed frame while still suppresses smaller variations (noise) effectively, so it allows for sharp edges to appear in the estimated frame. One example is $\|\boldsymbol{H}\boldsymbol{f}_{\boldsymbol{i}}\|_1$ where $\|\cdot\|_1$ denotes norm-1 (defined as $\|\boldsymbol{A}\|_1 = \sum|a_i|$). In our framework, we chose the following form for the regularization term: $\|\nabla\boldsymbol{f}_i\|_1 = \|\boldsymbol{H}_h\boldsymbol{f}_i\|_1 + \|\boldsymbol{H}_v\boldsymbol{f}_i\|_1$ where ∇ is the gradient operator, and $\boldsymbol{H}_h$ and $\boldsymbol{H}_v$ are first-order derivatives (FODs) in the horizontal and vertical directions, respectively. Therefore, the following optimization framework is stemmed from (4):

$$\boldsymbol{f}_i = \arg\min_{\boldsymbol{f}_i} \left\{ \sum_{\substack{k=j-a \\ j=i/s}}^{j+b} \|\boldsymbol{D}\boldsymbol{B}_k\boldsymbol{M}_{j,k}\boldsymbol{f}_i - \boldsymbol{g}_k\|_2^2 + \lambda\|\boldsymbol{H}_h\boldsymbol{f}_i\|_1 + \lambda\|\boldsymbol{H}_v\boldsymbol{f}_i\|_1 \right\} \tag{5}$$

To improve the optimization framework, we apply a few modifications to the functional in (5).

Remark 1 The vulnerability of the functional in (5) to the motion estimation error can be reduced by applying the adaptive weighting operator $\boldsymbol{O}_k$ defined in (6) to the norm-2 function in the fidelity term, i.e. $\|\boldsymbol{O}_k(\boldsymbol{D}\boldsymbol{B}_k\boldsymbol{M}_{j,k}\boldsymbol{f}_i - \boldsymbol{g}_k)\|_2^2$. The operator $\boldsymbol{O}_k$ is a diagonal matrix that assigns smaller weights to the outlier pixels.

$$\boldsymbol{O}_k = \text{diag}\left(\exp\left\{-\frac{\|\boldsymbol{D}\boldsymbol{B}_k\boldsymbol{M}_{j,k}\boldsymbol{f}_i - \boldsymbol{Y}_k\|_1}{\sigma}\right\}\right) \tag{6}$$

What matrix $\boldsymbol{O}_k$ does is assigning a lower weight to the pixels in the kth LR frame that have a higher deviation from the central LR frame to lessen the contribution of those pixels

[1] $\lfloor\cdot\rfloor$ is the floor operator.

[2] A few first and last frames of the video may have less number of adjacent frames. Also, for real-time applications, b should be set to zero.

in estimating $\boldsymbol{f}_i$. The scalar parameter σ in (6) determines the decay speed of the exponential function.

Remark 2 Modifying the norm-2 function of the fidelity term in (5) with $\|\boldsymbol{DB}_k\boldsymbol{M}_{j,k}(\boldsymbol{I}-\beta\boldsymbol{S})\boldsymbol{f}_i-\boldsymbol{g}_k\|_2^2$ will boost the sharpness of the estimated frame $\boldsymbol{f}_i$, where $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{S}$ is a high-pass filter operator, and β is a scalar that controls the sharpness amount.

According to the unsharp masking technique [31], an edge-sharpened frame $\widehat{\boldsymbol{f}}_i$ can be obtained from a frame $\boldsymbol{f}_i$ by summing up $\boldsymbol{f}_i$ with its high-passed filtered form, i.e. $\widehat{\boldsymbol{f}}_i=\boldsymbol{f}_i+\beta\boldsymbol{S}\boldsymbol{f}_i$. Consequently, by replacing $\boldsymbol{f}_i$ in the likelihood term with $\boldsymbol{f}_i-\beta\boldsymbol{S}\boldsymbol{f}_i=(\boldsymbol{I}-\beta\boldsymbol{S})\boldsymbol{f}_i$ a sharper image is obtained. We do not need to apply the above modification to $\boldsymbol{f}_i$ in the regularization term since a high-pass filtering operation already exists in this term. Our experiments show that the optimization problem modified using this technique converges as fast as the original one.

Remark 3 The temporal consistency of the estimated video is improved by adding the term $\|\boldsymbol{f}_i-\boldsymbol{M}_{i-1,i}\boldsymbol{f}_{i-1}\|_2^2$ to the functional in (5), which minimizes the error between each estimated frame $\boldsymbol{f}_i$ and its motion-compensated previous estimated frame $\boldsymbol{f}_{i-1}$.

The modified optimization framework using the above propositions is obtained as:

$$\boldsymbol{f}_i=\arg\min_{X_i}\left\{\sum_{\substack{k=j-a\\ j=i/s}}^{j+b}\left\|\boldsymbol{O}_k\big(\boldsymbol{DB}_k\boldsymbol{M}_{j,k}(\boldsymbol{I}-\beta\boldsymbol{S})\boldsymbol{f}_i-\boldsymbol{g}_k\big)\right\|_2^2\right. \tag{7}$$
$$\left.+\lambda\|\boldsymbol{H}_h\boldsymbol{f}_i\|_1\lambda\|\boldsymbol{H}_v\boldsymbol{f}_i\|_1+\gamma\|\boldsymbol{f}_i-\ \boldsymbol{M}_{i-1,i}\boldsymbol{f}_{i-1}\|_2^2\right\}$$

The optimization problem in (11) is convex but non-quadratic and can be solved using the iteratively reweighted least-squares (IRLS) method [32]. IRLS can solve (11) in an iterative manner in which each step comprises solving a weighted least-square problem. If $\boldsymbol{f}_i^{(n)}$ is the k th HR frame to be estimated at the n th iteration of IRLS, then $\|\boldsymbol{H}\boldsymbol{f}_i^{(n)}\|_1$[$\boldsymbol{H}$ stands for $\boldsymbol{H}_h$ or $\boldsymbol{H}_v$ in (7)] can be replaced by $\left(\boldsymbol{H}\boldsymbol{f}_i^{(n)}\right)^T\boldsymbol{W}_i^{(n-1)}\left(\boldsymbol{H}\boldsymbol{f}_i^{(n)}\right)$ where $\boldsymbol{W}_i^{(n-1)}=\text{diag}\left(\left|\boldsymbol{H}\boldsymbol{f}_i^{(n-1)}\right|\right)^{-1}$. To prevent division by zero, zero elements of $\boldsymbol{H}\boldsymbol{f}_i^{(n-1)}$ are replaced with a small number ϵ (e.g. 0.01).

Remark 4 Using IRLS, the functional in (7) results in the following linear equation where $\boldsymbol{A}_{i,k}^{(n-1)}=\boldsymbol{O}_k^{(n-1)}\boldsymbol{DB}_k\boldsymbol{M}_{jk}(\boldsymbol{I}-\beta\boldsymbol{S})$.

$$\left(\sum_k\boldsymbol{A}_{i,k}^{(n-1)T}\boldsymbol{A}_{i,k}^{(n-1)}+\lambda\boldsymbol{H}_h^T\boldsymbol{W}_h^{(n-1)}{}_i\boldsymbol{H}_h+\lambda\boldsymbol{H}_v^T\boldsymbol{W}_v^{(n-1)}{}_i\boldsymbol{H}_v+\gamma\boldsymbol{I}\right)$$
$$\boldsymbol{f}_i^{(n)}=\sum_k\boldsymbol{A}_{i,k}^{(n-1)T}\boldsymbol{g}_k+\gamma\boldsymbol{M}_{i-1,i}\boldsymbol{f}_{i-1} \tag{8}$$

The equation in (8) can be easily proved by replacing the norm-1 terms in (7) with their equivalent IRLS forms for the nth iteration, taking the derivative of (7) with respect to $\boldsymbol{f}_i^{(n)}$, and setting the derivative to zero. IRLS iterates between solving the least square problem in (8) using an iterative method such as Conjugate Gradient [33] and estimating $\boldsymbol{A}_{i,k}$ and $\boldsymbol{W}_i$ matrices based on the value of $\boldsymbol{f}_i^{(n-1)}$. The advantage of using Conjugate Gradient to solve (8) at the nth iteration of IRLS is that the matrix in the left-hand side of (8) does not require explicit calculation since it can be decomposed into a set of filtering and weighting operations.

2.3 Motion estimation

For spatial-only SR (no temporal upsampling), motion estimation can be performed using either a central or a sequential scheme [5]. In the central scheme, motion is directly estimated between the current frame and its adjacent frames (Fig. 1a). However, in the sequential scheme, the motion is first estimated between each adjacent frame and its previous frame (Fig. 1b); then, the central motion is obtained from the sequentially estimated motion. While the former approach provides better accuracy, the latter has significantly less computational complexity since only one motion is estimated from/to each frame when estimating multiple HR frames using SR. However, for spatio-temporal SR (STSR), the central scheme cannot be used since we do not have proper initial estimates for the frames missing in the input LR video (due to different temporal resolutions) before motion is estimated.

We expand the model employed in [5] to estimate motion for our proposed STSR method using a sequential scheme. In Fig. 2a, the solid circles are related to the frame positions available in the input LR video, and the empty circles correspond to the frame positions missing in the input LR video due to a lower temporal resolution. For every frame position (solid or empty), we aim to estimate motion from that position to all its neighboring positions using the following procedure:

1. Upsample each LR frame $\boldsymbol{g}_j$ individually via interpolation (e.g. using Bilinear or Bicubic methods) to obtain upsampled frame $\boldsymbol{g}_j$.
2. Estimate motion sequentially from each upsampled LR frame to its previous upsampled LR frame, i.e. $\boldsymbol{M}_{j,j-1}$ (Fig. 2b). Hence $\boldsymbol{z}_{j-1}=\boldsymbol{M}_{j,j-1}\boldsymbol{z}_j$.
3. Obtain motion between adjacent HR frame positions (Fig. 2c) using motion between adjacent LR frame positions as $\boldsymbol{M}_{i,i-1}=\boldsymbol{M}_{i,i-s}/s$. This conversion is obtained

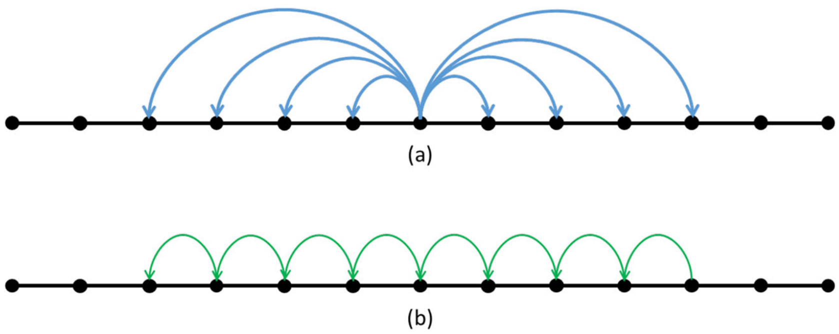


Fig. 1 Two schemes for estimating motion between the frames in SR. **a** Central scheme. **b** Sequential scheme

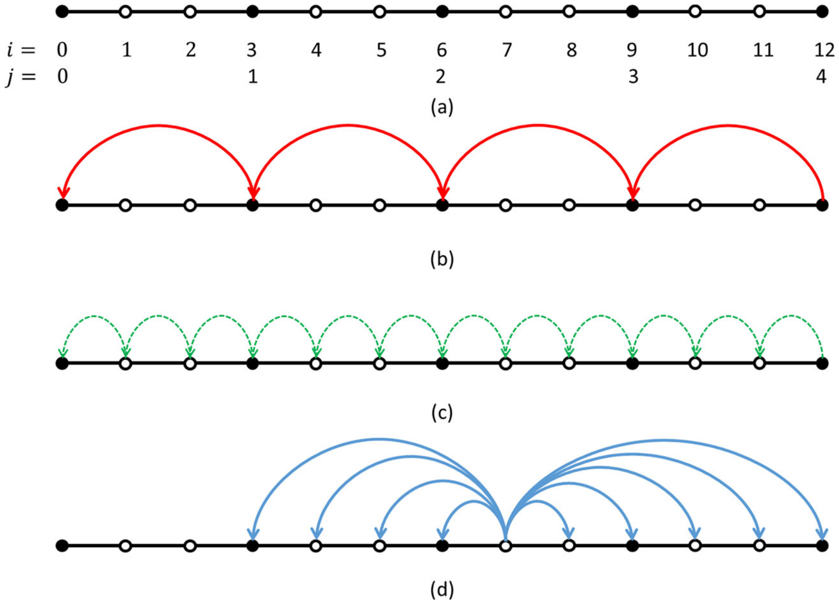


Fig. 2 Steps to estimate motion for the current frame ($i = 7$)

by assuming that the motion between two consecutive solid circles is distributed linearly at the intermediate empty circles.

4. Obtain motion from the current frame position to all its adjacent frame positions using (9) (sequential to central motion conversion). For instance, in Fig. 2d, the current HR frame position is $i = 7$, and the number of forward and backward adjacent frames are chosen such that we get two neighboring LR frame positions (black dots) in each direction, which yields $a = 4$ and $b = 5$.

$$\boldsymbol{M}_{i,k} = \begin{cases} \sum_{j=k+1}^{i} \boldsymbol{M}_{j,j-1} & a \leq k < i \\ 0 & k = i \\ -\sum_{j=i+1}^{k} \boldsymbol{M}_{j,j-1} & i < k \leq b \end{cases} \tag{9}$$

2.4 Initial estimate and color handling

A reasonable initial estimate for the HR frames, i.e. $\boldsymbol{f}_i^{(0)}$, is essential because it helps the SR algorithm reach the final solution in fewer iterations. Also, due to the SR problem's ill-posedness, multiple solutions may exist that minimize the optimization functional, so different initial estimates may result in different solutions due to the framework's local minima. We use a multi-frame non-uniform interpolation method followed by a single-frame deburring step to obtain an initial estimate for our MFSR problem. Rather than using the imaging model in (1), we use a suboptimal model by swapping the motion and blur operators and assuming similar blur kernel and noise characteristics for all frames, which yields:

$$\boldsymbol{g}_k = \boldsymbol{D}\boldsymbol{M}_{j,k}\boldsymbol{B}\boldsymbol{f}_i^{(0)} + \boldsymbol{n_k} = \boldsymbol{D}\boldsymbol{M}_{j,k}z_i + \boldsymbol{n_k}, \quad j = \lfloor i/s \rfloor, \quad k \in [j-a, \quad j+b] \tag{10}$$

where $z_i = \boldsymbol{B}\boldsymbol{f}_i^{(0)}$. An intuitive way to estimate z_i would be:

$$z_i = \sum_k {\boldsymbol{M}_{i,k}}^{-1} \boldsymbol{D}^{-1} \boldsymbol{g}_k = \sum_k \boldsymbol{M}_{k,i} \boldsymbol{D}^T \boldsymbol{g}_k \tag{11}$$

According to (11) the LR frames are projected onto the HR grid through upsampling and warping. Since motion vectors have arbitrary values, the projected points may not be uniformly distributed over the HR grid. Therefore, a non-uniform interpolation process is required to estimate the HR grid's pixel values from the projected points. Once z_i is obtained, $\boldsymbol{f}_i^{(0)}$ can be estimated by removing blurring and noise. We use a few iterations of the proposed STSR framework in Sect. (2.2) with $\boldsymbol{D}$ set to the identity matrix $\boldsymbol{I}$ (i.e. no upsampling).

If the input video is in the RGB color space, SR must be applied to all three red, green, and blue color channels since they need to have the same resolution. However, since the human visual system (HVS) is less sensitive to color than luminance (gray level), a more efficient way would be to decorrelate luminance from color and apply SR only to the luminance channel. This process is done in video coding using the YCbCr color space where Y expresses luminance and Cb and Cr convey color information [5]. Using this approach, we apply SR only to the Y channel and upsample Cb and Cr channels via Bicubic interpolation.

3 Experimental results

Unlike most state-of-the-art (SOTA) SR methods, our proposed method does not include a training step. As the first set of experiments to test our method, we use the Vid4 [4] and SPMCs-30 [34] benchmark datasets. Vid4 which is used by most publications contains four video sequences (City, Calendar, Foliage, and Walk) of slightly different sizes close to 720 × 576, each of which has at least 34 frames. SPMCs-30 contains 30 video sequences of dynamic scenes, each has 31 frames of size 960 × 540. Our proposed method is compared with SOTA SFSR and MFSR methods, including VSRnet [13], VESCPN [14], DBPN [15], RDN [16], RCAN [17], TOFlow [12], and TDAN [18] for 4X spatial upsampling, similar to [18]. Table 1 shows the quantitative comparison using PSNR (in dB) and SSIM [35] quality metrics. The results on SPMCs-30 are not reported for VSRnet [13] and VESCPN [14] since their source codes or reconstructed frames are not publicly available. The visual comparisons of different methods on Vid4 and SPMCs-30 datasets are shown in Figs. 3 and 4, respectively. Our proposed method demonstrates similar or better results than those SOTA methods.

Figure 5 provides quantitative comparisons between our proposed method and Bicubic. Figure 5a shows the PSNR variations of our proposed SR method versus Bicubic concerning variation in the standard deviation (σ) of Gaussian blur. The maximum PSNR values for both SR and Bicubic are obtained for σ in the range of [0.6, 1]. In this range, SR shows an average PSNR difference of 7.4dB compared to the Bicubic interpolation, which is an impressive improvement. A blur function with small support ($\sigma \in [0.6, 1]$) is effective in suppressing noise. However, as σ increases, the LR images become very blurry and SR becomes less effective as reflected in its PSNR values getting closer to Bicubic.

Figure 5b represents the variations in PSNR values of our SR method and Bicubic for different downsampling ratios. The increase in the downsampling ratio results in lower-resolution LR images, making it harder for SR to recover the missing details. Despite a considerable drop in PSNR for the downsampling of 4, our SR method has still provided 4.4dB more improvement than Bicubic. Figure 5c also demonstrates the PSNR variation for different noise power or SNR values. For higher values of σ of noise, we increase the regularization parameter λ in (8) to increase the smoothness of the reconstructed frame. This figure shows that SR has a higher PSNR difference with Bicubic for higher SNR values.

Figure 6 shows another example of improving the spatial resolution of a traffic light footage using the proposed method compared to Bicubic. Due to the high distance of the scene from the camera, the target object is noisy and has a low resolution. Therefore, it is hard to read the plate number using Bicubic upsampling. However, our proposed method has significantly improved the image quality.

The next experiment shown in Fig. 7 demonstrates the performance of our proposed SR method in removing the temporal blur. Motion blur is a temporal artifact in nature, as it appears due to the fast movements of objects in the scene or the camera itself during the exposure time [25, 36]. When the scene is roughly static and planar, the perceived spatial motion blur would be space-invariant (similar for all regions). However, when the scene is highly dynamic during the exposure time or the camera is filming a scene with far and near-field objects while moving fast, the perceived motion blur would be space-variant. Removing a space-variant motion blur from a single image is highly challenging. It requires segmenting the scene into objects and background, applying different deblurring to different parts of the scene, and finally putting the deblurred objects back together in a coherent way. On the other hand, our multi-frame SR method is inherently capable of removing motion blur through applying temporal deblurring, as described below.

Figure 7a shows one frame of the Old Town Cross video, and Fig. 7b demonstrates the LR frame generated by applying a temporal rectangular blur of length 5 (so the exposure time is expanded over five frames). Since the scene is not planar and the camera is not moving parallel to the scene, a more severe motion blur happens on the right side than on the left side of the frame. Figure 7c, d result from the motion

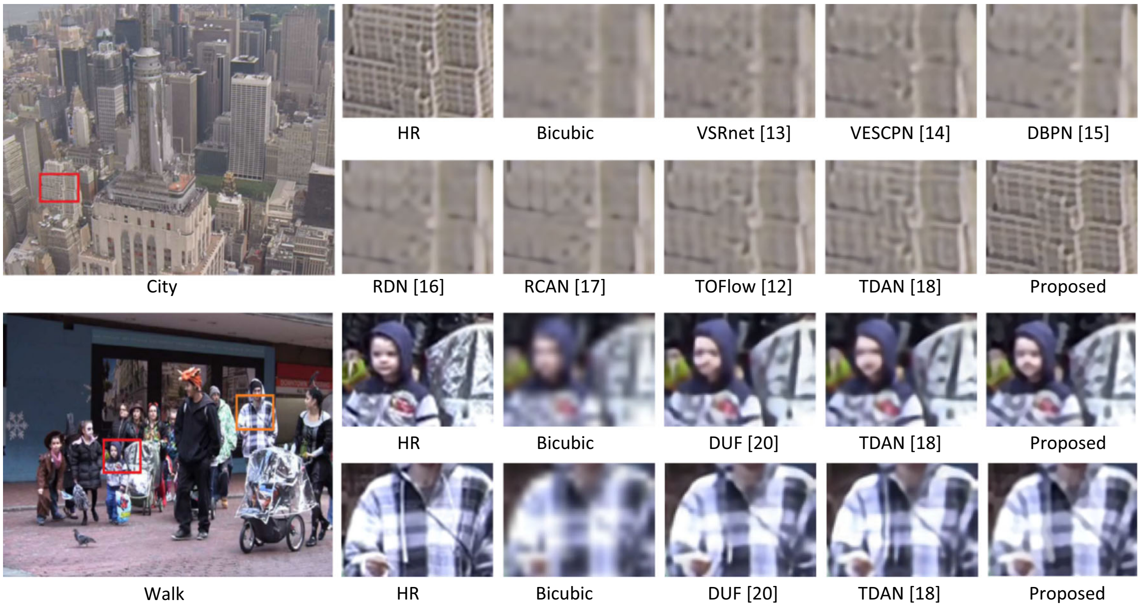


Fig. 3 Comparison results of our proposed method with SOTA methods on the Vid4 dataset

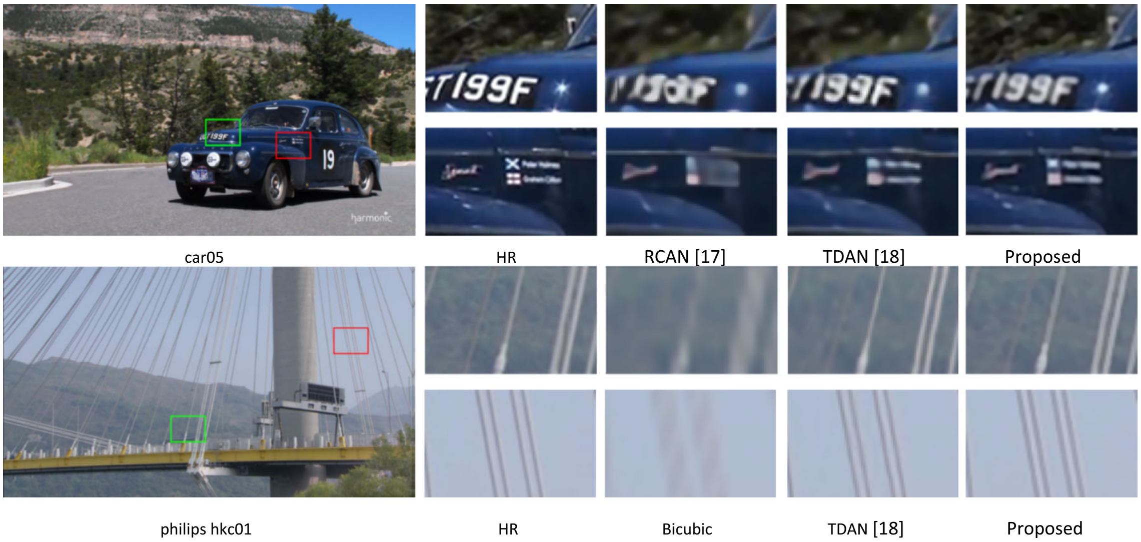


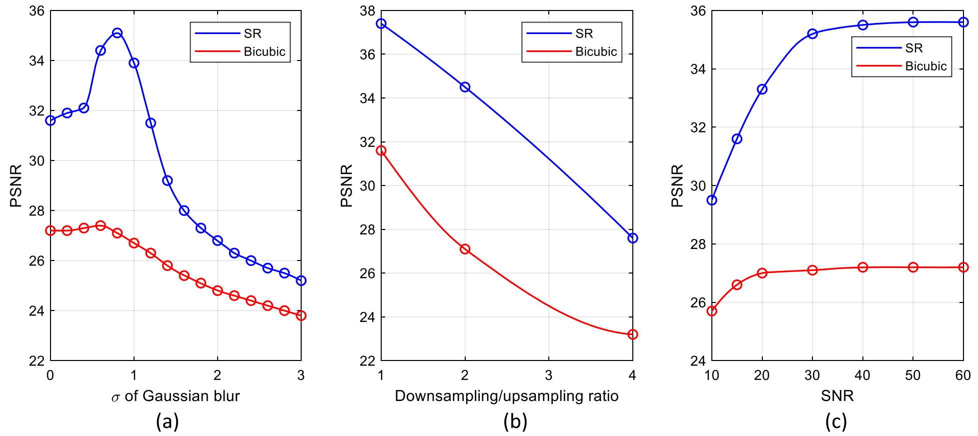
Fig. 4 Comparison results of our proposed method with SOTA methods on the SPMCs-30 dataset

Table 1 Quantitative comparison of our proposed method with SOTA methods in terms of PSNR (dB) and SSIM quality metrics

Methods	Bicubic	VSRnet [13]	VESCPN [14]	DBPN [15]	RDN [16]	RCAN [17]	TOFlow [12]	TDAN [18]	Proposed
Vid4	23.79/0.633	24.73/0.697	25.34/0.730	25.33/0.731	25.40/0.735	25.42/0.737	25.90/0.765	26.86/0.814	**26.89/0.821**
SPMCs-30	27.08/0.744	–	–	29.76/0.830	29.92/0.836	30.07/0.841	29.47/0.831	30.80/**0.869**	**30.86**/0.858

The bold numbers are the largest ones

Fig. 5 Variation of peak signal to noise ratio (PSNR) with respect to the variation of the following SR parameters: **a** Standard deviation (σ) of Gaussian blur; **b** Downsampling/upsampling ratio; **c** signal to noise ratio (SNR)



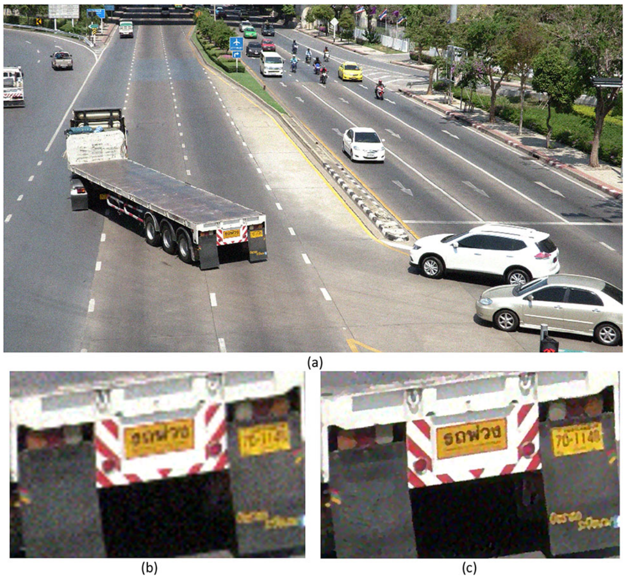


Fig. 6 Improving the quality of a traffic camera video. **a** One video frame; **b** closeup view of the frame upsampled by Bicubic; **c** improved closeup using our proposed method

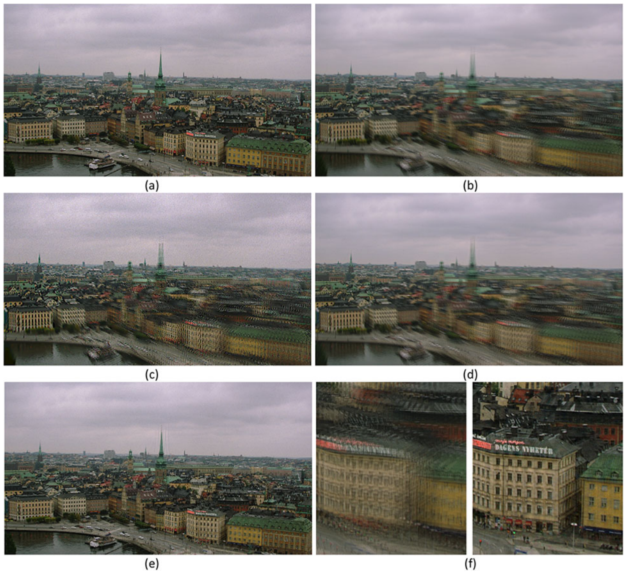


Fig. 7 Removing space-variant motion blur for the Old Town Cross video. **a** Ground-truth frame; **b** LR frame; **c** result by [37]; **d** result by [38]; **e** result by our method through temporal deburring; **f** close-up from images (**b**) and (**e**)

deblurring method proposed in [37] and the online GAN[3]-based deblurring and upscaling tool [38], respectively. These methods have failed to perform any noticeable improvement due to the space-variant nature of the perceived motion blur. The result of our proposed method is presented in Fig. 7e, and closeups from the LR and restored frames are shown in Fig. 7f. Our method has successfully removed the space-variant blur by improving the video's temporal resolution.

Our proposed method can also be used to create high-quality slow-motion videos or interpolated views. Tested on several videos, including building structures, fast-moving clouds, and candle fumes, the frame rate is set to increase by a factor of 15. These results are not shown here due to the difficulty of perceiving the temporal effect on a still manuscript. Please refer to the package accompanying this article for a few examples.

4 Conclusion

Improving the resolution of natural videos using the super-resolution (SR) technique is a highly ill-posed problem. This paper presents a space–time super-resolution method to increase a single video's spatial and temporal resolutions to alleviate aliasing, blurring, and noise artifacts. A new motion estimation framework is proposed to first estimate motion sequentially and then derive motion between the central and neighboring frames. An initial estimate of

[3] Generative Adversarial Networks.

the output frame is obtained using a non-uniform interpolation technique to derive the upsampled frame, followed by a deblurring step. An optimization formulation is derived using the maximum a posteriori probability (MAP) estimator, which estimates a high-resolution (HR) video frame from a few neighboring low-resolution (LR) frames of the input video. We incorporate an edge-sharpening operation into the optimization problem to further enhance the edges. We also improve temporal consistency in the reconstructed video by minimizing the error between successive estimated frames. The results of the proposed method are compared with a few SOTA methods, including two deep learning-based ones. The results confirm the effectiveness of the proposed SR method in improving the quality of natural videos.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11760-023-02675-z.

Author contributions Both authors contributed to all parts of the manuscripts.

Funding There is no funding or grant supporting this manuscript.

Data availability No dataset is used by this research.

Declarations

Conflict of interest The authors declare that there is no conflict of interest associated with this study.

Ethical approval This research is not sponsored by any industrial company and does not aim to endorse any commercial product.

References

1. Park, S.C., Park, M.K., Kang, M.G.: Super-resolution image reconstruction: a technical overview. IEEE Signal Process. Mag. Inst. Electr. Electron. Eng. Inc **20**(3), 21–36 (2003). https://doi.org/10.1109/MSP.2003.1203207
2. Tian, J., Ma, K.K.: A survey on super-resolution imaging. Signal Image Video Process **5**(3), 329–342 (2011). https://doi.org/10.1007/s11760-010-0204-6
3. Yue, L., Shen, H., Li, J., Yuan, Q., Zhang, H., Zhang, L.: Image super-resolution: the techniques, applications, and future. Signal Process. **128**, 389–408 (2016). https://doi.org/10.1016/j.sigpro.2016.05.002
4. Liu, C., Sun, D.: A Bayesian approach to adaptive video super resolution. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 209–216. (2011). https://doi.org/10.1109/CVPR.2011.5995614
5. Faramarzi, E., Rajan, D., Fernandes, F.C.A., Christensen, M.P.: Blind super resolution of real-life video sequences. IEEE Trans. Image Process. **25**(4), 1544–1555 (2016). https://doi.org/10.1109/TIP.2016.2523344
6. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-based super-resolution. IEEE Comput. Gr. Appl. **22**(2), 56–65 (2002). https://doi.org/10.1109/38.988747
7. Nasrollahi, K., Moeslund, T.B.: Super-resolution: a comprehensive survey. Mach. Vis. Appl. **25**, 1423–1468 (2014). https://doi.org/10.1007/s00138-014-0623-4
8. Singh, A., Singh, J.: Survey on single image based super-resolution—implementation challenges and solutions. Multimed. Tools Appl. **79**(3), 1641–1672 (2019). https://doi.org/10.1007/s11042-019-08254-0
9. Shahar, O., Faktor, A., Irani, M.: Space-time super-resolution from a single video. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 3353–3360, (2011). https://doi.org/10.1109/CVPR.2011.5995360
10. Son, S. et al.: NTIRE 2021 challenge on video super-resolution. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 166–181, (2021). https://doi.org/10.1109/CVPRW53098.2021.00026
11. Xiang, X., Tian, Y., Zhang, Y., Fu, Y., Allebach, J.P., Xu, C.: Zooming slow-mo: fast and accurate one-stage space-time video super-resolution. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, pp. 3367–3376, (2020). https://doi.org/10.1109/CVPR42600.2020.00343
12. Xue, T., Chen, B., Wu, J., Wei, D., Freeman, W.T.: Video enhancement with task-oriented flow. Int. J. Comput. Vis. **127**(8), 1106–1125 (2017). https://doi.org/10.1007/s11263-018-01144-2
13. Kappeler, A., Yoo, S., Dai, Q., Katsaggelos, A.K.: Video super-resolution with convolutional neural networks. IEEE Trans. Comput. Imaging **2**(2), 109–122 (2016). https://doi.org/10.1109/TCI.2016.2532323
14. Caballero, J. et al.: Real-time video super-resolution with spatio-temporal networks and motion compensation. In: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, vol. 2017-January, pp. 2848–2857, (2016), https://doi.org/10.1109/CVPR.2017.304
15. Haris, M., Shakhnarovich, G., Ukita, N.: Deep back-projection networks for super-resolution. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1664–1673, (2018). https://doi.org/10.1109/CVPR.2018.00179
16. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. (2018), Accessed: May 14, 2023. [Online]. Available: https://arxiv.org/abs/1802.08797v2
17. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Yun, F.: Image super-resolution using very deep residual channel attention networks. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp. 294–310. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-030-01234-2_18
18. Tian, Y., Zhang, Y., Fu, Y., Xu, C.: TDAN: temporally deformable alignment network for video super-resolution. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 3357–3366, (2018). https://doi.org/10.1109/CVPR42600.2020.00342
19. Wang, Z., Chen, J., Hoi, S.C.H.: Deep learning for image super-resolution: a survey. IEEE Trans. Pattern. Anal. Mach. Intell. (2020). https://doi.org/10.1109/tpami.2020.2982166
20. Jo, Y., Oh, W.S., Kang, J., Kim, S.J.: Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 3224–3232, (2018). https://doi.org/10.1109/CVPR.2018.00340
21. Isobe, T., et al.: Video super-resolution with temporal group attention. In: Proceedings of the IEEE Computer Society Conference on

Computer Vision and Pattern Recognition, pp. 8005–8014, (2020). https://doi.org/10.1109/CVPR42600.2020.00803

22. Kawulok, M., Benecki, P., Piechaczek, S., Hrynczenko, K., Kostrzewa, D., Nalepa, J.: Deep learning for multiple-image super-resolution. IEEE Geosci. Remote Sens. Lett. **17**(6), 1062–1066 (2020). https://doi.org/10.1109/LGRS.2019.2940483
23. Salvetti, F., Mazzia, V., Khaliq, A., Chiaberge, M.: Multi-image super resolution of remotely sensed images using residual attention deep neural networks. Remote Sens. **12**(14), 2207 (2020). https://doi.org/10.3390/rs12142207
24. Molini, A.B., Valsesia, D., Fracastoro, G., Magli, E.: DeepSUM: deep neural network for super-resolution of unregistered multitemporal images. IEEE Trans. Geosci. Remote Sens. **58**(5), 3644–3656 (2019). https://doi.org/10.1109/TGRS.2019.2959248
25. Shechtman, E., Caspi, Y., Irani, M.: Space-time super-resolution. IEEE Trans. Pattern Anal. Mach. Intell. (2005). https://doi.org/10.1109/TPAMI.2005.85
26. Faramarzi, E., Rajan, D., Christensen, M.P.: Space-time super-resolution from multiple-videos. In: 2012 11th International Conference on Information Science, Signal Processing and their Applications, ISSPA 2012, (2012). https://doi.org/10.1109/ISSPA.2012.6310553
27. Mudenagudi, U., Banerjee, S., Kalra, P.K.: Space-time super-resolution using graph-cut optimization. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 995–1008 (2011). https://doi.org/10.1109/TPAMI.2010.167
28. Caspi, Y., Irani, M.: Spatio-temporal alignment of sequences. IEEE Trans. Pattern Anal. Mach. Intell. **24**(11), 1409–1424 (2002). https://doi.org/10.1109/TPAMI.2002.1046148
29. Takeda, H., Milanfar, P., Protter, M., Elad, M.: Super-resolution without explicit subpixel motion estimation. IEEE Trans. Image Process. **18**(9), 1958–1975 (2009). https://doi.org/10.1109/TIP.2009.2023703
30. Faramarzi, E., Rajan, D., Christensen, M.P.: Space-time super-resolution from multiple-videos. In: The 11th International Conference on Information Sciences, Signal Processing and their Applications, pp. 23–28, (2012)
31. Gonzalez, R.C., Woods, R.E.: Digital image processing, 4th edn. Pearson, London (2018)
32. Burrus, C.S.: Iterative reweighted least squares. *OpenStax CNX module: m45285*, (2014)
33. Shewchuk, J.R.: An introduction to the conjugate gradient method without the agonizing Pain, Edition 1 ¼. In: School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, (1994)
34. Tao, X., Gao, H., Liao, R., Wang, J., Jia, J.: Detail-revealing Deep Video Super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision, vol. 2017-October, pp. 4482–4490, (2017). https://doi.org/10.1109/ICCV.2017.479
35. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004). https://doi.org/10.1109/TIP.2003.819861
36. Takeda, H., Milanfar, P.: Removing motion blur with space-time processing. IEEE Trans. Image Process. **20**(10), 2990–3000 (2011). https://doi.org/10.1109/TIP.2011.2131666
37. Cho, S., Lee, S.: Fast motion deblurring, p. 1. ACM Transactions on Graphics, ACM Press, New York (2009)
38. Deblurring–imageupscaler. https://imageupscaler.com/deblurring/. Accessed 01 Jan 2021

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.