Space-Time Super-Resolution from a Single Video

Oded Shahar   Alon Faktor   Michal Irani
Dept. of Computer Science and Applied Math
The Weizmann Institute of Science,   ISRAEL

Abstract

Spatial *Super Resolution (SR) aims to recover fine image details, smaller than a pixel size.* Temporal *SR aims to recover rapid dynamic events that occur faster than the video frame-rate, and are therefore invisible or seen incorrectly in the video sequence. Previous methods for Space-Time SR combined information from multiple video recordings of the same dynamic scene. In this paper we show how this can be done from a single video recording. Our approach is based on the observation that small space-time patches ('ST-patches', e.g.,* $5{\times}5{\times}3$*) of a single 'natural video', recur many times inside the same video sequence at multiple spatio-temporal scales. We statistically explore the degree of these ST-patch recurrences inside 'natural videos', and show that this is a very strong statistical phenomenon. Space-time SR is obtained by combining information from multiple ST-patches at sub-frame accuracy. We show how finding similar ST-patches can be done both efficiently (with a randomized-based search in space-time), and at sub-frame accuracy (despite severe motion aliasing). Our approach is particularly useful for temporal SR, resolving both severe motion aliasing and severe motion blur in complex 'natural videos'.*

1. Introduction

A video camera has limited spatial and temporal resolutions. The spatial resolution is determined by the spatial pixel density in each frame and by the camera Point Spread Function. These factors limit the *minimal size* of spatial features that can be visible in an image. The temporal resolution is determined by the frame-rate and by the exposure-time of the camera. These limit the *maximal speed* of dynamic events that can be observed in a video sequence.

Increasing the spatial resolution in video sequences was obtained by combining information from *multiple frames* at sub-pixel accuracy (e.g., [5, 10]). More recent methods have extended these ideas to increase both the spatial and the temporal resolution, by combining information from *multiple video recordings* of a dynamic scene, at sub-pixel and *sub-frame* accuracy [14]. Such multiple recordings can be obtained either from multiple video cameras recording simultaneously the same dynamic scene, or else from a single video camera recording multiple repetitions of a cyclic motion [12, 16] (thus acquiring the multiple recordings sequentially). These requirements restrict the applicability of space-time Super-Resolution (**SR**).

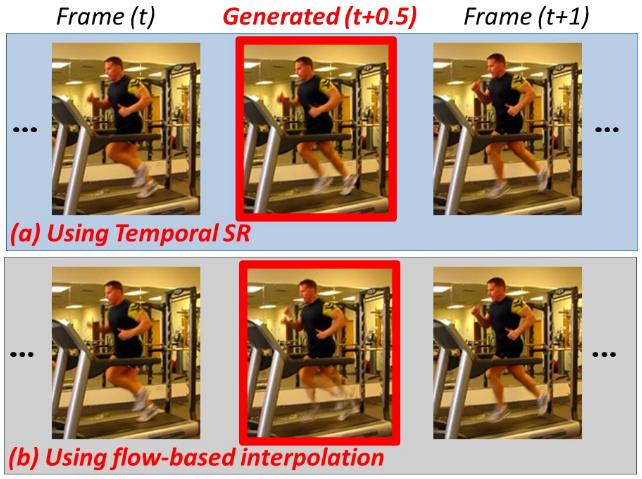


Figure 1. **Temporal SR vs. Frame-Interpolation:** *A fast runner induces large and complex motions. Temporal SR increases the frame-rate of a video sequence, providing 'true slow motion', while resolving motion aliasing and motion blur. In contrast, even sophisticated flow-based frame-interpolation fails to handle such complex motions. See video on the project website.*

In this paper we show that space-time SR can be achieved from a *single video recording of general dynamic scenes*. Our approach extends the recent image-based SR method of Glasner *et al.* [9] into space-time. We observe that small space-time patches (**ST-patches**, e.g., $5{\times}5{\times}3$) of any 'natural video' tend to recur repeatedly inside the same video at multiple spatio-temporal scales. We show that this is a strong property of 'natural videos', even if we do not visually perceive any repetitive behavior. This is statistically verified on a large dataset of 'natural videos'. Recurrence of small ST-patches within the input scale of the video sequence induce *'Classical' Space-Time SR constraints* (a la [14], but from a single video recording). Recurrence of ST-patches across coarser spatio-temporal video scales provide low-res/high-res pairs of ST-patches. Such 'example' pairs suggest how the space-time resolution of the input ST-patch might possibly be increased, thus inducing *"Example-Based" Space-Time SR constraints*.

We further show how these similar ST-patches can be found. This is done both efficiently (with a randomized-based search in space-time over the huge search space), and at sub-frame accuracy (despite severe motion aliasing). Our approach is particularly powerful for increasing the temporal resolution in video. It allows recovery of very fast dy-

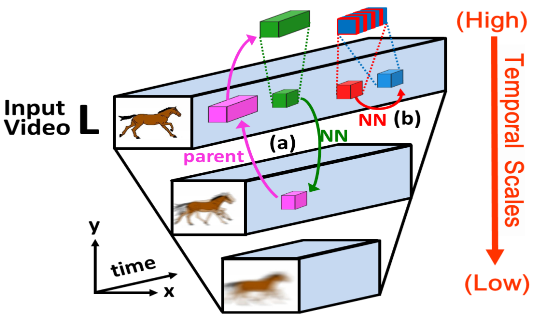


Figure 2. **ST-Patch recurrence within and across temporal scales of a single video:** *(a)* ***"Across scales":*** *A ST-patch in the input video L (small green) is found elsewhere in a coarser temporal scale (small pink). The high-res corresponding parent ST-patch (large pink) indicates how the slow motion version of the input ST-patch might look like (large green). (b)* ***"Within scale":*** *A ST-patch in L (red) is found elsewhere in the same scale (blue) at sub-frame misalignment. This can be exploited to obtain finer sampling rate of the continuous dynamic event (red+blue).*

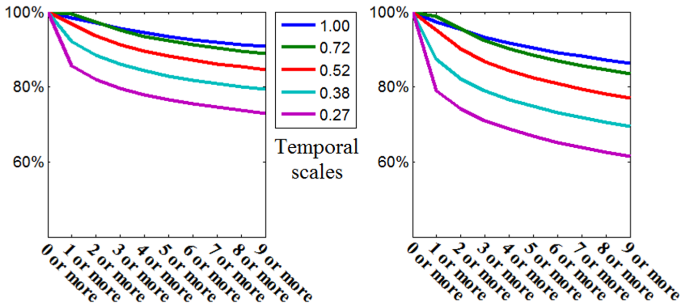
namic events (beyond the temporal Nyquist frequency of the video), that were otherwise invisible or else induced false apparent visual motions in the video. Unlike frame-interpolation, temporal SR provides "true slow-motion", *resolving both severe motion aliasing and severe motion blur* in videos of very complex dynamic scenes (see Fig. 1).

A related work [15] used a simplified notion of information repetition across video scales by matching 2D image patches (as opposed to 3D ST-patches). 2D spatial image patches cannot account for sub-frame temporal misalignments, cannot resolve temporal (motion) aliasing, and cannot represent complex non-linear motions. ST-patches from multiple space-time video scales were used in [7] for various applications. However, the Epitomic representation does not account for sub-frame misalignments of ST-patches, nor can it undo severe motion blur and motion aliasing. In [17], a method is presented for sophisticated upscaling of intensities in space and in time, by parametrically modeling local intensity variations using their $3D$ Taylor expansion. This yields smooth upscaling of local intensities, but cannot undo motion aliasing.

The rest of this paper is organized as follows: In Sec. 2 we statistically examine the observation of small ST-patch recurrences in a single 'natural video'. Sec. 3 presents our space-time SR framework. Sec. 4 discusses strong visual phenomena in videos, that are unique to the added temporal dimension. Sec. 5 explains how 'similar' ST-patches can be found at sub-frame accuracy, despite severe motion aliasing. Results are provided in Sec. 6, and in *www.wisdom.weizmann.ac.il/~vision/SingleVideoSR.html*.

2. Recurrence of ST-Patches in a Single Video

Similarly to the observation made by Glasner *et al.* [9] regarding natural images, we observe that 'natural videos' contain repetitive dynamic visual content. In particular, small ST-patches (e.g. 5x5x3) tend to recur within the sequence itself, both within the input scale ("within scale"), as well as across coarser scales in space and time ("across scales"). This observation is crucial for the success of our space-time SR framework. While repetition of ST-patches across coarser spatial scales may not be surprising (a natural extension of the observation in [9] regarding spatial patches), what is less intuitive, and to some degree more powerful, is the observation that ST-patches recur across coarser *temporal scales*. In this section we try to empirically quantify this notion of ST-patch recurrence "within" and "across" temporal scales of a single video, and show that it is indeed a strong property present in a wide variety of 'natural videos', *even if no repetitive behavior is perceived.* Intuitively, small ST-patches have very simple local structures (space-time corners, moving edges, etc.) Thus, ST-patches from totally different global motions can locally resemble each other.



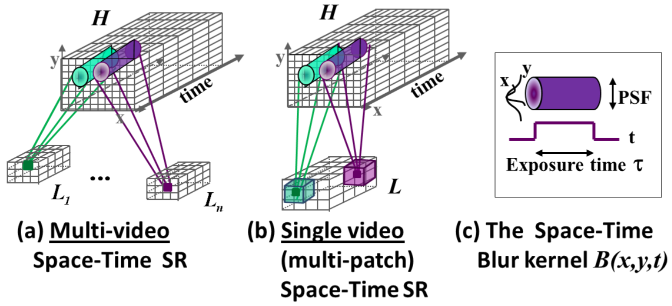
(a) KTH dataset

(b) HOHA dataset

Figure 3. **Average ST-Patch recurrence within and across temporal scales of a single video:** *(averaged over dozens of videos - see text for more details). The graphs show for each of the datasets (KTH and HOHA) the percent of ST-patches for which there exist n or more similar patches, measured at several temporal scales.*

Fig. 2 schematically illustrates what we mean by "ST-patch recurrence" *within and across scales* of a single video sequence. An input ST-patch "recurs" in *another* scale if it appears "as is" (without downscale) in a coarser temporal scale of the sequence. Having found a similar ST-patch in a coarser temporal scale, we can extract its parent ST-patch from the higher temporal scale (Fig. 2.a). This provides an indication of how the slow motion version of the input ST-patch might look like. When an input ST-patch "recurs" in the *same* scale, the corresponding ST-patch will inevitably have a sub-frame misalignment with the input ST-patch (Fig. 2.b). This can be exploited to obtain finer sampling rate of the continuous dynamic event.

We statistically tested this observation on a large dataset of close to 100 video clips containing a wide variety of 'natural videos' taken from the KTH Human Actions dataset (*www.nada.kth.se/cvap/actions/*) and the Hollywood Human Actions dataset (HOHA *www.irisa.fr/vista/Equipe/People/Laptev/download.html*). The



(a) Multi-video Space-Time SR

(b) Single video (multi-patch) Space-Time SR

(c) The Space-Time Blur kernel *B(x,y,t)*

Figure 4. **Multi-video vs. Single-video SR:** *(a) Pixels in multiple low-res video sequences impose multiple linear constraints on the high-res unknowns within the support of their space-time blur kernels. (b) Similar ST-patches within a single low-res video L can be regarded as if extracted from multiple different low-res videos of the same high-res dynamic scene, thus inducing multiple linear constraints on the high-res unknowns in H. (c) The space-time blur function $B(x, y, t)$ is determined by the spatial Point Spread Function and the temporal exposure time.*

KTH dataset contains repetitive human behaviors like walking, hand waving, etc. with static background. Though repetitive, the motions are quite complex and non-rigid. The HOHA dataset includes a wider variety of complex foreground and background dynamics, with no visually obvious repetition, as well as camera motion. The hypothesis was tested on 5x5x3 overlapping ST-patches, after removing their DC (mean intensity value). Each sequence was blurred and subsampled in time, generating a cascade of temporal scales of 0.85^l compared to the input scale, where $l = 0, 1, \ldots, 8$. The size of the smallest scale was 0.27 (i.e., things move 4 times faster than the input video). For each ST-patch in each sequence separately, we computed how many "similar" ST-patches it has in each of the temporal scales of the same sequence, including the input scale. In order to get reliable statistics, we excluded all static ST-patches (which obviously recur repeatedly), performing the test only on the dynamical ST-patches.

For the purpose of the statistics, we used a simple SSD-based distances between ST-patches. Note that dynamical ST-patches (with strong temporal derivatives) tend to have much larger SSD errors than static ones, when compared to very similar looking ST-patches. Thus, for each ST-patch we compute a patch-specific "good distance", which is the SSD normalized by the mean temporal derivative of the ST-patch. When this normalized SSD is below the threshold then this ST-patch will be considered similar. In Sec. 5 we present a more sophisticated distance, which also accounts for sub-frame misalignments and handles motion aliasing.

The statistics was averaged over all the 10^7 ST-patches in the dataset, resulting in Fig. 3. Almost all the dynamical ST-patches (90% in KTH, 86% in HOHA) have 9 or more similar ST-patches in the input scale. 85% KTH ST-patches and 77% HOHA ST-patches have 9 or more similar ST-patches in 0.52 temporal scale, and 70% and 61% in temporal scale 0.27, respectively. As expected, KTH shows higher recurrence since it contains repetitive behaviors. However, HOHA also exhibits a high degree of ST-patch recurrence, although no repetitive dynamics is observed.

3. Single Video SR – Overview of the Approach

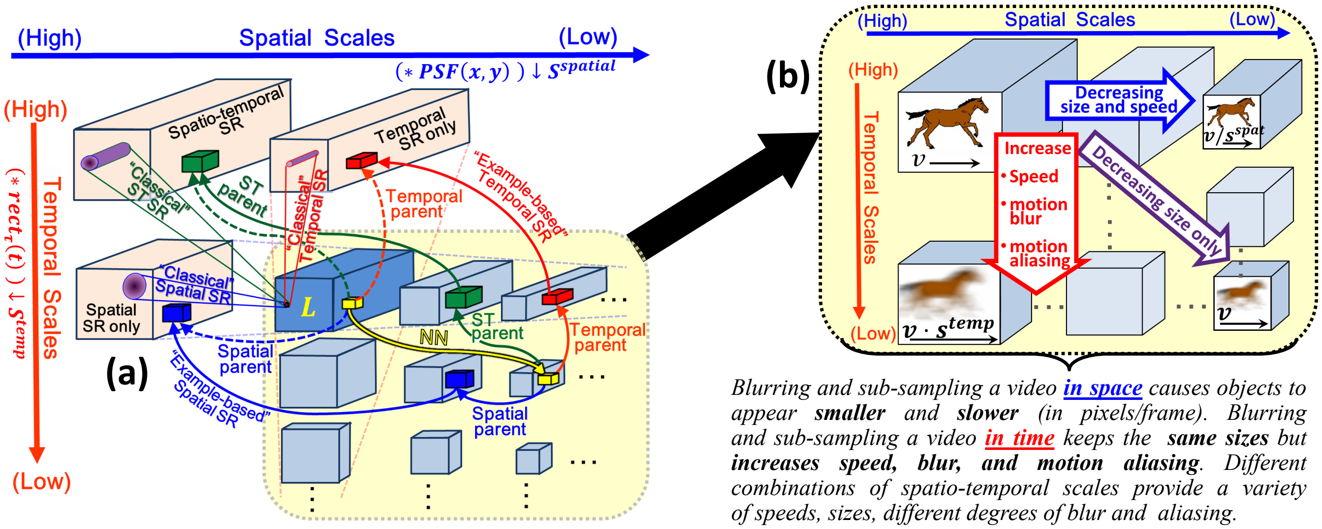
We extend the 2D image-based framework and algorithm of single-image SR [9], into the 3D space-time volume: Recurrence of small ST-patches within the input scale of the video forms the basis for *'Classical' Space-Time SR* [14] (but *from a single video recording*). Recurrence of ST-patches across coarser spatio-temporal video scales gives rise to *"Example-Based" Space-Time SR* (but learned internally from a single video). We emphasize the issues that go beyond a simple extension of [9] from 2D to 3D, and are unique to the added temporal dimension. We further elaborate on increasing the *temporal* resolution (turning low frame-rate videos into high frame-rate ones). Unlike frame-interpolation, temporal SR provides "true slow-motion", *resolving both severe motion aliasing and severe motion blur* in videos of complex dynamic scenes.

3.1. 'Classical' space-time SR: In multi-sequence space-time SR [14], a set of low-resolution video sequences $\{L_1, ..., L_n\}$ recording the same dynamic scene is given, and the goal is to recover a video sequence H of *higher spatial resolution* (higher pixel-density, with higher spatial frequencies) and of *higher temporal resolution* (higher frame-rate, with higher temporal frequencies). Each low-res sequence L_j $(j = 1, .., n)$ is assumed to have been generated from H by convolution with a *space-time blur function* B_j, followed by subsampling in space and in time (see Fig. 4.a): $L_j = \left(H * B_j\right) \downarrow s_j^{spatial} \downarrow s_j^{temporal}$, where $\downarrow$ denotes a subsampling operation and $s_j^{spatial}$,$s_j^{temporal}$ are the *spatial* and *temporal* subsampling rates. The space-time blur kernel B_j is composed of a *spatial blur* $B_j^{spatial}(x, y) = PSF_j(x, y)$ (the camera ***Point Spread Function***), convolved with a *temporal blur* $B_j^{temporal}(t) = rect_{\tau_j}(t)$ (which is a temporal rectangular function whose support is the camera ***Exposure-Time***[1] τ_j). Thus, each low-res pixel $p = (x, y, t)$ in each low-res video L_j induces one linear constraint on the unknown high-res intensity values within a local *space-time neighborhood* around its corresponding high-res pixel $q \in H$ (the size of the neighborhood is determined by the support of the space-time blur kernel B_j):

$$L_j(p)=\left(H{*}B_j\right)(q)=\textstyle\sum_{q_i\in Support(B_j)}H(q_i)\,B_j(q_i{-}q) \quad (1)$$

where $\{H(q_i)\}$ are the unknown high-res intensity values.

[1] The exposure time τ ranges between *instantaneous exposure time* ($\tau \approx 0$, in which case the temporal blur function reduces to a delta function: $B^{temporal}(t) \approx \delta(t)$), to *full exposure time* (in which case τ equals the temporal distance between 2 consecutive frames: $\tau = \frac{1}{(InputFrameRate)}$ *sec*).



Blurring and sub-sampling a video ***in space*** *causes objects to appear* ***smaller*** *and* ***slower*** *(in pixels/frame). Blurring and sub-sampling a video* ***in time*** *keeps the* ***same sizes*** *but* ***increases speed, blur, and motion aliasing****. Different combinations of spatio-temporal scales provide a variety of speeds, sizes, different degrees of blur and aliasing.*

Figure 5. **The space-time pyramid:** *Blurring and sub-sampling the input video L (dark blue) in space and in time, generates a cascade of spatio-temporal resolutions (light blue). Each input ST-patch (yellow) searches for a similar ST-patches (Nearest Neighbors - NN) in lower pyramid levels. Each matching ST-patch may have a spatial parent (blue), a temporal parent (red), or a spatio-temporal parent (green). These low-res/high-res pairs of ST-patches induce "Example-based" spatial-SR / temporal-SR / space-time SR. These result in constrains relating the input video L to the 3 different possible high-res outputs H (the pink volumes). In addition, "Classical" SR constraints (spatial, temporal, or space-time) induce additional linear constraints relating the input L to the unknown high-res outputs.*

If enough low-res videos are available (at *sub-pixel* spatial shifts and *sub-frame* temporal shifts), then the number of independent equations exceeds the number of unknowns.

The space-time SR equations can be written separably: $L_j = \left((H * PSF_j(x,y)) \downarrow s_j^{spatial} * rect_{\tau_j}(t)\right) \downarrow s_j^{temporal}$
This leads to two interesting special sub-cases:
(i) ***Temporal SR only***: $L_j = \left(H * rect_{\tau_j}(t)\right) \downarrow s_j^{temporal}$.
(ii) ***Spatial SR only***: $L_j = \left(H * PSF_j(x,y)\right) \downarrow s_j^{spatial}$.

3.2. Employing in-scale patch recurrence: When there is only a single low-resolution video sequence L, the 'classical' SR problem of recovering a high-resolution video H becomes under-determined. However, as observed in Sec. 2, ST-patches recur many times within a single video L. Let $p = (x, y, t)$ be a pixel in L, and $\mathcal{P}$ be its surrounding ST-patch (e.g., 5×5×3), then it has multiple ST-patches recurrences $\mathcal{P}_1, ...\mathcal{P}_k$ in L (inevitably, at subpixel shifts, and – more importantly – at subframe shifts). These patches can be treated as if taken from k different low-res videos of the same high-res "dynamic scene", thus inducing k times more linear constraints (Eq. 1) on the high-res intensities of pixels within the space-time neighborhood of $q \in H$ (see Fig. 4.b). For increased numerical stability, each equation induced by a similar ST-patch $\mathcal{P}_i$ is weighted by the degree of its match to its source ST-patch $\mathcal{P}$. Thus, ST-patches with a better match to $\mathcal{P}$ will have a stronger influence on the recovered high-res pixel values than ST-patches of lower match.

3.3. Employing cross-scale patch recurrence: As observed in Sec. 2, ST-patches recur not only within the input scale, but also across other spatio-temporal scales of a single video L. As in [9], the low-res/high-res patch correspondences can be employed to obtain 'hypotheses' on what the high-res space-time parent of the input ST-patch might look like, thus providing additional linear constraints on the unknown pixel intensities in the high-res output video (see [9] for more details). However, unlike [9], here each ST-patch can have 3 *different types of high-res 'parents'*: (i) a *temporal* parent, (ii) a *spatial* parent, and (iii) a *spatio-temporal* parent (see Fig. 5.a). Thus, a single match between an input ST-patch and a ST-patch in a different spatio-temporal scale, provides information for any of the following 3 *different SR schemes*: (i) Spatial SR only, (ii) Temporal SR only, and (iii) combined Spatio-Temporal SR. Of particular interest and applicability are the first two schemes:

(i) ***Spatial SR only***: The purpose here is to increase the spatial resolution in each video frame. However, there are two major differences between the spatial SR scheme of [9] and that proposed here: (a) The scheme of [9] employs 2D image patches. Even if their Nearest-Neighbor (**NN**) patch search is allowed to use information from the *entire video* (all frames), the patches are still only 2D image patches, thus the SR reconstruction is solved independently for each frame. This does not enforce consistency between consecutive frames, and may therefore lead to flickering effects in time. In contrast, searching for NN of volumetric ST-patches enforces consistency of the spatial SR between consecutive frames. (b) The scheme of [9] searches for NN only across different spatial scales. Here we extend the NN search also across different temporal scales of each spatial scale, even when the goal is only to increase the spatial resolution. Matches across different scales (both in space and in time) can provide information for increasing the spatial resolution (see Fig. 5.b). This provides a larger and more expressive variety of low-res/high-res ST-patch pairs.

(ii) ***Temporal SR only***: The purpose here is to increase the temporal resolution (frame-rate) in each video frame, while *resolving both motion aliasing and motion blur*. While "Example-Based" spatial SR using an *external database* of 'natural images' may be feasible (and was done before [8]), *"Example-Based" temporal SR using an external database of 'natural videos' is NOT practically realistic* (and to our best knowledge, was never done). This is because a 'representative' databases of 'natural video sequences' is bound to be unrealistically large if required to provide enough variety of all natural ST-patches (covering the huge diversity of $Appearance \times Dynamics$). This is where internal video redundancies are particularly useful, especially in light of the strong internal recurrence of ST-patches across multiple *temporal scales* of a single 'natural video' (Sec. 2), which gives rise to "Example-Based" temporal SR using internal examples. Once again, the NN search for similar ST-patches is not restricted to temporal scales of the input video, but is also done across different spatial scales of each temporal scale, even when the goal is only to increase the temporal resolution. Matches across different scales (both in space and in time) provide a larger and more expressive variety of low-res/high-res ST-patch pairs (see Fig. 5.b).

3.4. Combining 'Classical' and "Example-Based" constraints: Similarly to Glasner *et al.* [9], the above two types of SR constraints can be combined into a single computational framework: Let $\mathcal{P}$ denote a ST-patch in the input video. We can search for similar ST-patches (NN) within the entire space-time pyramid. Let $\mathcal{P}'$ be a good matching ST-patch found somewhere in a lower-res pyramid level. Then its higher-res 'parent' ST-patch, $\mathcal{Q}'$, provides a 'suggestion' (constraints) on what the unknown high-res parent ST-patch $\mathcal{Q}$ of the input ST-patch $\mathcal{P}$ might possibly look like. Note that the parent ST-patch $\mathcal{Q}'$ of $\mathcal{P}'$ can be a *spatial parent*, a *temporal parent*, or a *spatio-temporal parent*. It will accordingly induce constraints on the corresponding space-time pyramid level which is *spatially higher*, *temporally higher*, or *spatio-temporally higher* than the input level (see Fig. 5.a). If this higher pyramid level coincides with the target high-res level (the output resolution level of H), then it provides explicit constraints on the unknown high-res intensities of H. If, however, the parent pyramid level does not coincide with the target (output) level, the ST-parent patch $\mathcal{Q}'$ can still induce linear constraints on the target-level intensities, indirectly. This is done by bridging the 'residual gap' between those two pyramid levels using the 'Classical' SR equations (Eq. (1)). When the gap between the levels is a spatial gap, it is bridged using equations induced by a scaled version of the PSF (usually with a smaller support), according to the residual spatial gap between the two levels (see [9] for more details). Similarly, when the gap between the levels is temporal, it is bridged using equations induced by a scaled version of the temporal blur $rect_\tau$ (usually with a smaller exposure-time τ), according to the residual temporal gap between the two levels. The closer the 'learned' parent patches are to the target resolution H, the better conditioned the resulting set of equations.

The above ideas can be translated to the following algorithm: For each pixel (x, y, t) in the input video L, extract its surrounding ST-patch, and find its k nearest ST-patch neighbors (e.g., $k = 5$) in the spatio-temporal pyramid constructed from L (Fig. 5a). Each such 'learned' low-res/high-res pair of ST-patches induces linear constraints (directly or indirectly) on the unknown intensities of the high-res output video H. Each such linear constraint is globally weighted by its reliability (determined by its ST-patch similarity score – see Sec. 5). Repeating the above for all pixels in L leads to a very large, yet *sparse*, set of linear equations on H, which can be efficiently solved using *conjugate gradient*.

4. Handling Motion-Blur and Motion-Aliasing

Rapid dynamic events that occur faster than the video frame-rate are not visible (or else observed incorrectly) in the recorded video. This problem is evident in sports videos (tennis, baseball, hockey), where it is often impossible to see the fast moving ball/puck in any individual frame. There are two well-known visual artifacts caused when fast motions are recorded by slow cameras: *motion-blur* and *motion-aliasing*. Both of these can be treated (partially reduced, and sometimes fully resolved) by *temporal SR*.

(i) Motion Blur: Fast moving objects produce a notable blur along their trajectory, often resulting in distorted or even unrecognizable object shapes. The faster the object moves, the stronger this effect is, especially if the trajectory of the moving object is not linear. Motion blur is thus a *spatial manifestation of a temporal phenomenon*. It is caused by *temporal blurring*, not by spatial blurring, and should therefore be treated temporally. Yet, most motion deblurring algorithms address the problem spatially (e.g., [3]). Spatial deblurring is complicated – it requires using different spatial blur kernels for differently moving parts of the image. This in turn requires segmentation of the moving objects and the estimation of their motions for obtaining these spatial blur kernels. Motion estimation may be impossible in the presence of severe motion blur and shape distortions (as in Fig. 6). A combined spatial and temporal treatment for handling motion blur (using specialized camera/acquisition design) was further suggested in [11, 13]. However, the objects were restricted to simple motions.

In contrast, when the video sequence is available, motion deblurring becomes simple by pure temporal treatment – simple *temporal* deblurring with the camera exposure

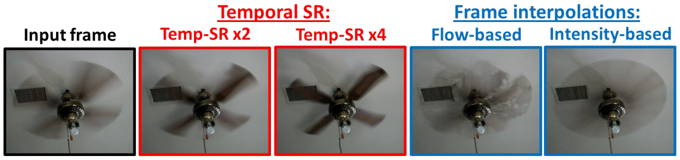


Figure 6. **Handling motion blur:** *A fast rotating fan induces severe motion blur. Temporal-SR reduces motion blur while frame interpolations cannot resolve this issue. See video on website.*

$rect_\tau(t)$. No special camera design, no restrictions on motions, *regardless of the type, speeds, or complexity of the motions*. The observation that motion blur is a *purely temporal* artifact is not intuitive. After all, the blur is visible in a single image frame. However, the blur results from integration over time during the frame exposure. *All pixels (static or dynamic) experience the same amount of temporal blur* (a convolution with $rect_\tau(t)$). Naturally, the temporal blur is visible only in image locations where there are temporal changes (moving objects, camera motion, illumination changes, etc.) Our algorithm addresses temporal blur directly by reducing the exposure time via temporal SR. This does not require any motion estimation or segmentation. As such, it is able to significantly reduce motion blur in very complex dynamic scenes (e.g., Fig. 6). The reduction in exposure time is obtained in two ways: *(i) "Example-Based" temporal SR constraints:* Low-res/high-res pairs of ST patches, extracted from different *temporal scales* in the space-time pyramid, provide examples of how to locally reduce exposure time when increasing the temporal resolution (see illustrations in Figs. 2 and 5.a). *(ii) 'Classical' temporal SR constraints:* The linear equations relating the input video sequence L to its higher temporal resolution output H, are expressed in terms of the temporal blur: $L_j = \big(H * rect_{\tau_j}(t)\big) \downarrow s_j^{temporal}$. Solving these equations when increasing the temporal resolution implicitly performs temporal deblurring, thus reducing the exposure time in H.

(ii) Motion Aliasing (Temporal Aliasing): Temporal/motion aliasing occurs when a very fast moving object induces temporal frequencies higher than the temporal sampling rate of the camera (the camera frame-rate). High temporal frequencies are then "folded" into the low temporal frequencies, generating temporal aliasing effects. The observable result is a distorted or even false trajectory of the moving object (such as in the "Turbine" video). Playing the video in 'slow-motion' or applying sophisticated temporal interpolation cannot recover the information lost in the temporal sampling process. A well-known example of motion aliasing is the "wagon wheel effect", where a very fast spinning wheel appears to be rotating in the "wrong" direction beyond a certain speed (see video on the project website).

'Classical' SR (from multiple images/videos) is all about 'unfolding' the 'folded' (aliased) high frequencies. 'Classical' *spatial SR* was shown to provide small (≤ 2) SR increases [2]. This is because the spatial blur (the PSF) is a pretty-good low-pass filter, reducing most high-frequencies to zero prior to sampling. In contrast, the temporal blur is a bad low-pass filter – a very narrow-support $rect$ function. The temporal sampling thus preserves many high temporal frequencies, in aliased form. 'Classical' *temporal SR* was indeed shown to provide much higher increases in temporal resolution [14]. In general, videos are characterized by much higher temporal aliasing than spatial aliasing.

Unfolding (recovering) the aliased high temporal frequencies was done in [14] by combining information from *multiple video recordings*. Here we show that this can be done from a *single video recording* L, as follows: *(i) 'Classical' temporal SR constraints:* Due to the repetitive nature of ST-patches in L (inevitably, at sub-frame offsets), the very high temporal frequencies are *scattered* across multiple ST-patches, but *in aliased form*. Detecting these repetitive ST-patches in L and combining their information at *sub-frame accuracy*, provides a denser sampling-rate in time. This allows recovery of the 'lost' high temporal frequencies (the correct motions of very fast moving objects) that are beyond the theoretical temporal Nyquist limit of the input video. An illustrative example of combining information from two ST-patches from the same temporal scale at sub-frame accuracy is shown in Fig. 2. This, of course, requires finding 'similar' ST-patches at sub-frame shifts (preferably, at sub-frame shifts of $\{n/s\}_{n=1}^{s-1}$, if the resolution is to be increased by a factor of s). A method for finding 'similar' patches at the desired sub-frame shifts (despite the severe temporal aliasing) is described next, in Sec. 5. *(ii) "Example-Based" temporal SR constraints:* Low-res/high-res pairs of ST-patches extracted from different *temporal scales* in the space-time pyramid provide examples of how to locally reduce motion aliasing when increasing the temporal resolution (see illustration in Fig. 5.a).

This yields a high-res video H from a single low-res video L, displaying more correctly the motions of the fast moving objects (see videos on the project website).

5. Find Similar ST-Patches at Sub-Frame Shifts

We saw that temporal aliasing is good for temporal SR as it preserves many of the high temporal frequencies (in aliased form). This, however, complicates the task of finding matching ST-patches at accurate *sub-frame alignment*. Interpolation-based SSD (which is adequate for matching spatial patches at sub-pixel shifts [9]), cannot be used here. Due to temporal aliasing, temporal interpolation at sub-frame shifts will generate false visual artifacts. Moreover, we cannot use the sequence-to-sequence alignment method of [6] (which was used for finding the global temporal offsets between videos in [14]), since it relies on coarsening and subsampling the videos in time (thus eliminating the aliased high temporal frequencies). But tiny ST-patches cannot be further coarsened and subsampled in time.

Inspired by the image-alignment method of [18], we develop a method for alignment of *small* ST-patches at sub-frame accuracy. We further show how this can be used for finding 'similar' ST-patches at desired sub-frame offsets, despite severe temporal aliasing.

Let $\mathcal{P}_0$ and $\mathcal{P}_1$ be two ST-patches which we wish to align. For simplicity, let us first assume that these ST-patches are 1-D discrete signals (vectors) in the temporal direction with length N (the number of frames in the video). Obviously, the temporal length of a ST-patch is much shorter, as will be addressed later. Following the footsteps of [18], assume that $\mathcal{P}_0$ and $\mathcal{P}_1$ were sampled from the same time-continuous signal $u(t)$ (the high-res dynamics we wish to recover), with an offset dt between them: $\mathcal{P}_0[n] = u\left(\frac{n}{N}\right)$, $\mathcal{P}_1[n] = u\left(\frac{n+dt}{N}\right)$. Finally, assume that $u(t)$ is band-limited with maximal temporal frequency Ω, and was sampled in a frequency smaller than Nyquist to generate $\mathcal{P}_0$ and $\mathcal{P}_1$ ($N \leq 2\Omega$, i.e., $\mathcal{P}_0$ and $\mathcal{P}_1$ exhibit aliasing). $\mathcal{P}_0$ and $\mathcal{P}_1$ can be written as a function of the Fourier coefficients $U[\omega]$ of $u(t)$: $\mathcal{P}_0[n] = \sum_{\omega \in [-\Omega,..,\Omega]} U[\omega] e^{\frac{i2\pi\omega n}{N}}$

$$\mathcal{P}_1[n] = \sum_{\omega \in [-\Omega,..,\Omega]} U[\omega] e^{\frac{i2\pi\omega(n+dt)}{N}} = \sum_{\omega \in [-\Omega,..,\Omega]} U[\omega] e^{\frac{i2\pi\omega n}{N}} e^{\frac{i2\pi\omega dt}{N}}$$

$e^{\frac{i2\pi\omega dt}{N}}$ is the phase shift resulting from the offset dt.

Writing this in matrix notation: $\mathcal{P}_0 = F_0 U$, $\mathcal{P}_1 = F_{dt} U$ where $U = \big(U[-\Omega], \ldots, U[\Omega]\big)^T$ and F_0 and F_{dt} are the appropriate matrices (Note that F_{dt} contains the information regarding the unknown sub-frame dt). Combining this into a single set of linear equations in U: $\mathcal{P} = FU$ (2) where $\mathcal{P} = [\mathcal{P}_0, \mathcal{P}_1]^T$ is a $2N \times 1$ vector and $F = [F_0, F_{dt}]^T$ is a $2N \times (2\Omega + 1)$ matrix.

From Eq.(2), we get that the sample vector $\mathcal{P}$ belongs to the subspace spanned by the columns of F. Note that F is determined by dt. We shall therefore seek an offset dt which *minimizes* the deviation of $\mathcal{P}$ from the subspace spanned by the columns of F:

$$dt = \text{argmin}_{dt'} e_{rr}(\mathcal{P}_0, \mathcal{P}_1, dt') = \text{argmin}_{dt'} \|\mathcal{P} - \hat{\mathcal{P}}\|^2 \quad (3)$$

where $\hat{\mathcal{P}}$ is the projection of $\mathcal{P}$ onto that subspace, obtained using the pseudo-inverse of F (assuming $2N \geq 2\Omega + 1$): $\hat{\mathcal{P}} = F(F^TF)^{-1}F^T\mathcal{P} \triangleq G\mathcal{P}$. G is a matrix of size $2N \times 2N$. A small error means high alignment quality. Eq.(3) is not simple to solve in closed form. However, one can try a small range of sub-frame shifts (e.g. $0, 0.1, .., 0.9$), and choose the one which minimizes Eq.(3). This provides alignment up to some sub-frame accuracy.

However, we wish to deal with small ST-patches and relax the assumption of temporal length N. Let $\mathcal{P}_0^M$ and $\mathcal{P}_1^M$ be short ST-patches (with only few frames $M \ll N$). We can regard them as long extended temporal patches (of length of the entire sequence - N), $\tilde{\mathcal{P}}_0$ and $\tilde{\mathcal{P}}_1$, multiplied by a Gaussian window $g[n]$, which has only M significant values in its center. We obtain: $\tilde{\mathcal{P}}_0[n] = g[n]\mathcal{P}_0[n]$ and $\tilde{\mathcal{P}}_1[n] = g[n]\mathcal{P}_1[n]$. Then the same derivations from before hold, only that now we use the Fourier coefficients $\tilde{U}(\omega) = U(\omega) * G(\omega)$, where $G(\omega)$ is the Fourier transform of $g[n]$. So we remain with same error $e_{rr}(\tilde{\mathcal{P}}_0, \tilde{\mathcal{P}}_1, dt)$ as before and the same formula for $\hat{\mathcal{P}}$.

Since most of the elements in $\tilde{\mathcal{P}}$ are now zero, we can neglect them and compute $e_{rr}(\tilde{\mathcal{P}}_0, \tilde{\mathcal{P}}_1, dt)$ by taking into consideration only the central non-zero parts of $\mathcal{P}_0$ and $\mathcal{P}_1$, $\mathcal{P}_0^M$ and $\mathcal{P}_1^M$. We will show that Eq.(3) can be written such that it depends only on $\mathcal{P}_0^M$ and $\mathcal{P}_1^M$ and a small $2M \times 2M$ (e.g. 6×6) matrix $\tilde{G}$. Lets define $\tilde{F}$ as the matrix obtained from F after removing from it the rows corresponding to the zero elements in $\tilde{\mathcal{P}}$ (i.e. if the i-th element in $\tilde{\mathcal{P}}$ is zero then we remove the i-th row in F). It is easy to show that: $F(F^TF)^{-1}F^T\tilde{\mathcal{P}} = \tilde{F}(F^TF)^{-1}\tilde{F}^T\mathcal{P}^M \triangleq \tilde{G}\mathcal{P}^M$. So it holds that: $\|\tilde{\mathcal{P}} - G\tilde{\mathcal{P}}\|^2 = \|\mathcal{P}^M - \tilde{G}\mathcal{P}^M\|^2$. Thus, we get a new measure distance for short ST-patches:

$$e_{rr}(\mathcal{P}_0^M, \mathcal{P}_1^M, dt) = \|\mathcal{P}^M - \tilde{G}\mathcal{P}^M\|^2 \quad (4)$$

For ST-patches with a spatial size bigger than 1×1, the vectors $\mathcal{P}_0^M, \mathcal{P}_1^M$ can be replaced by matrices whose columns are the 1D temporal vectors extracted from the ST-patch at each of its spatial locations.

Now, given a ST-patch $\mathcal{P}_0^M$, we would like to search for its NN patches at a desired sub-frame shift dt (e.g., $dt = 0.5$ is ideal for increasing the temporal resolution by x2). For every ST-patch $\mathcal{P}_1^M$, we can use the above distance measure (Eq. 4) as an indication of the quality of its match to $\mathcal{P}_0^M$ at the desired sub-frame misalignment dt. However, this may not be the optimal sub-frame alignment between these two ST-patches. For example, a ST-patch may have a reasonable match with itself at half a frame shift, but will have a better match with itself at zero shift. We would like to discard such candidate ST-patches. To guarantee that a match is not only good, but also reliable (i.e. provides a local optimum at dt), we use the following distance measure, which combines a quality term $e_{rr}(dt)$ and a local optimality term $Opt(dt)$:

$$D(\mathcal{P}_0^M, \mathcal{P}_1^M) = e_{rr}(\mathcal{P}_0^M, \mathcal{P}_1^M, dt) \cdot Opt(\mathcal{P}_0^M, \mathcal{P}_1^M, dt)^{\lambda} \quad (5)$$

where $Opt(\mathcal{P}_0^M, \mathcal{P}_1^M, dt) = \frac{e_{rr}(\mathcal{P}_0^M, \mathcal{P}_1^M, dt)}{\min_{dt_j \neq dt} e_{rr}(\mathcal{P}_0^M, \mathcal{P}_1^M, dt_j)}$ and $0 \leq dt_j \leq 1$ are a few discrete sub-frame shifts.
For example, when searching for 'similar' ST-patches at half-frame offset ($dt = 0.5$), our optimality term is: $Opt(\mathcal{P}_0^M, \mathcal{P}_1^M, 0.5) = \frac{(e_{rr}(\mathcal{P}_0^M, \mathcal{P}_1^M, 0.5))}{\min\{e_{rr}(\mathcal{P}_0^M, \mathcal{P}_1^M, 0), e_{rr}(\mathcal{P}_0^M, \mathcal{P}_1^M, 1)\}}$.

The $e_{rr}(dt)$ term prefers solutions with high absolute quality and the $Opt(dt)$ term prefers solutions for which the misalignment dt is superior over all other possible local misalignments. The parameter $\lambda > 0$ determines which term is more dominant (for high values the optimality term is more dominant). In our experiments we used $\lambda = 2$.

5.1. Efficient ANN Search in Space-Time: Using traditional ANN (Approximate Nearest Neighbor) Search algorithms, such as kd-tree [1], is not feasible in our application.

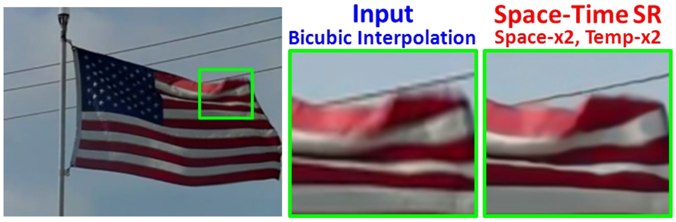


Figure 7. **Space-Time SR of Natural Video:** *Strong wind causes the flag to wave in complex non-rigid motions, inducing motion blur. Space-time SR provides both an increase in Spatial resolution (e.g., the electricity wire), and increase in Temporal resolution (reducing motion blur). See video on the project website.*

This is due to the large memory requirements (a huge number of ST-patches in every video, each producing a long $1D$ vector in the kd-tree), and the long running time (which increases significantly with the length of the vectors). Moreover, kd-tree is limited to vector similarity based on the L_2 norm. However, our similarity measure (Eq. 5) is not L_2. In order to search for matching ST-patches we extended the randomized algorithm of [4] from 2D images to 3D space-time volumes. The algorithm in [4] finds ANN matches between image patches, by choosing randomly neighbors, and then exploiting the natural coherence in the imagery to propagate good patch matches to surrounding areas. Our extension allows us to find patch matches for 3D ST-patches across multiple videos using any distance function we desire. This is done by choosing randomly neighbors across the videos, and then propagating the good matches in both the temporal and the spatial dimensions. In addition, we can search for matches for a variety of different ST-patch sizes around each pixel, and compare their match quality by normalizing the similarity measure by the ST-patch size. Using this randomized search algorithm we can find NN at a reasonable time, low memory cost, and more importantly, we can use complex distance functions such as Eq. 5.

6. Results

We tested our framework on a variety of videos, illustrating that it can handle severe motion blur, motion aliasing, as well as general 'natural videos' with complex motions. In our experiments we built the spatio-temporal pyramid using a cascade of blurring and subsampling in space (at 0.9^l spatial scales, $l = 1,..,6$). Each video in the spatial pyramid was then blurred and subsampled in time (by $1/n$ temporal factors, $n = 1,..,6$). Due to the temporal aliasing induced by the temporal blur (the estimated camera exposure time) and subsampling, we kept all the temporally subsampled versions, enriching the temporal pyramid. Increase of temporal resolution by large factors was performed gradually, each time adding the intermediate result to the pyramid.

The "Fan" and "Flag" videos (Figs. 6 and 7) demonstrate how our temporal SR reduces *motion blur*. The "Treadmill" (Fig. 1.a), "Fan" and "Turbine" videos demonstrate how our temporal SR handles severe *motion aliasing*. Note that the "Treadmill" video is not truly repetitive. The runner changes his speed, changes his body tilt, etc. Only when working with ST-patches can small consistent segments from different cycles be combined. The "Flag" video further demonstrates space-time SR (×2 in space and ×2 in time) of a complex 'natural video'. The flow-based frame interpolation compared against in some of the figures and videos, was generated using MV-Tools (*www.avisynth.org.ru*). Please see full videos on: *www.wisdom.weizmann.ac.il/~vision/SingleVideoSR.html*. When there are no severe temporal artifacts, (i.e., no severe motion aliasing or motion blur, as is typical of most feature movies), our temporal-SR algorithm gracefully reduces to good temporal interpolation. See examples on website.

Acknowledgment: The authors would like to thank Y. Caspi for his helpful comments on the paper. This work was partially funded by the Israel Science Foundation.

References

[1] S. Arya and D. M. Mount. Approximate nearest neighbor queries in fixed dimensions. In *SODA*, 1993.

[2] S. Baker and T. Kanade. Limits on super-resolution and how to break them. *PAMI*, 24(1), September 2002.

[3] B. Bascle, A. Blake, and A. Zisserman. Motion deblurring and super-resolution from an image sequence,ECCV,1996.

[4] C. Barnes, E. Shechtman, D. Goldman, A. Finkelstein. The generalized patchmatch correspondence algorithm. In *ECCV*, 2010.

[5] D. Capel. *Image Mosaicing and Super-Resolution*. Springer–Verlag, 2004.

[6] Y. Caspi and M. Irani. Spatio-temporal alignment of sequences. *PAMI*, 24(11), 2002.

[7] V. Cheung. B. J. Frey, N. Jojic. Video epitomes. *IJCV,2008*.

[8] W. Freeman, T. Jones, and E. Pasztor. Example-based super-resolution. *Comp. Graph. Appl.*, 22, 2002.

[9] D. Glasner, S. Bagon, and M. Irani. Super-resolution from a single image. In *ICCV*, 2009.

[10] M. Irani and S. Peleg. Improving resolution by image registration. *CVGIP*, 53, 1991.

[11] A. Levin, P. Sand, T. S. Cho, F. Durand, and W. T. Freeman. Motion-invariant photography. *SIGGRAPH*, 2008.

[12] Y. Makihara, A. Mori, and Y. Yagi. Temporal super resolution from a single quasi-periodic image sequence based on phase registration. *ACCV*, 2010.

[13] R. Raskar, A. Agrawal, and J. Tumblin. Coded exposure photography: Motion deblurring using fluttered shutter. *ACM Trans. on Graphics*, 25(3), 2006.

[14] E. Shechtman, Y. Caspi, and M. Irani. Space-time super-resolution. *PAMI*, 27(4), 2005.

[15] M. Shimano, T. Okabe, I. Sato, and Y. Sato. Video temporal super-resolution based on self-similarity. *ACCV*, 2010.

[16] M. Singh, A. Basu, and M. K. Mandal. Event dynamics based temporal registration. *T-Multimedia*, 2007.

[17] H. Takeda. P. Milanfar, M. Protter, M. Elad. Super-resolution without explicit subpixel motion estimation. *T-IP*, 2009.

[18] P. Vandewalle, L. Sbaiz, M. Vetterli, and S. Susstrunk. Super-resolution from highly undersampled images. *ICIP*, 2005.