Video Super-Resolution using Simultaneous Motion and Intensity Calculations

Sune Høgild Keller*, François Lauze and Mads Nielsen

Abstract—**In this paper, we propose an energy-based algorithm for motion-compensated video super-resolution (VSR) targeted on upscaling of standard definition (SD) video to high definition (HD) video. Since the motion (flow field) of the image sequence is generally unknown, we introduce a formulation for the joint estimation of a super-resolution sequence and its flow field. Via the calculus of variations, this leads to a coupled system of partial differential equations for image sequence and motion estimation. We solve a simplified form of this system and as a by-product we indeed provide a motion field for super-resolved sequences. Computing super-resolved flows has to our knowledge not been done before. Most advanced super-resolution (SR) methods found in literature cannot be applied to general video with arbitrary scene content and/or arbitrary optical flows, as it is possible with our simultaneous VSR method. Series of experiments show that our method outperforms other VSR methods when dealing with general video input, and that it continues to provide good results even for large scaling factors up to 8×8.**

Index Terms—**Super-resolution, video upscaling, video processing, motion compensation, motion super-resolution, variational methods, and partial differential equations (PDEs).**

I. Introduction

SUPER-resolution (SR) is a thoroughly investigated subject in image processing where a majority of the work is focussed on the creation of one high-resolution (HR) still image from n low-resolution (LR) images (see for instance [8] by Chaudhuri). In Video Super-Resolution (VSR) (or "multiframe super-resolution" as it is sometimes called), one instead seeks to recreate n high-resolution frames from n low-resolution ones.

Our main motivation for doing VSR is to solve the problem of showing low-resolution typically standard definition (SD) video signals on high definition (HD) displays at high quality. Processing could be done either at the broadcaster or at the receiver. The SD signals (typically in PAL, NTSC or SECAM) need to be upscaled before they can be displayed on modern HD display devices. Even though high-definition television (HDTV) is gradually taking over from current SD standards, there will be a need for upscaling far into the future as both broadcasters and private homes will have large archives of SD material. The increase in spatial resolution does not need to stop with the current HDTV formats, and the availability of high quality upscaling will potentially allow for lower bit rates and/or higher image quality in encoded video.

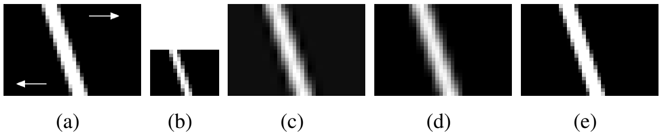


Fig. 1. Frame no. 3 of 5 in the Skew sequence: (a) ground truth (arrows show skew motion), (b) ground truth downsampled by 0.5x0.5 with loss of information. 2x2 super-resolution on (b): (c) bicubic interpolation, (d) bilinear interpolation and (e) our motion-compensated simultaneous VSR algorithm.

Temporal information is generally not taken into account in upscaling algorithms of today's (high end) HD equipment. In fact, simple bilinear or bicubic interpolation is used. (Bilinear interpolation is the choice e.g. in the renowned high end video processors by Faroudja.) In this paper, we propose a full motion-compensated variational method for super-resolution. It can be seen in Fig. 1(c) that bicubic interpolation comes a little closer to the ground truth shown in Fig. 1(a) than bilinear interpolation does (Fig. 1(d)). By comparison, our approach produces results (Fig. 1(e)) indistinguishable from the ground truth.

Upscaling is an inverse problem for which super-resolution methodologies attempt to provide a solution. The direct problem associated with it is the image formation equation. We follow Lin and Shum [29] and start describing the projection $\mathcal{R}$ of a high-resolution (HR) image into a low-resolution (LR) image formulating the *super-resolution constraint*. In the continuous setting, it is:

$$u_L(y) = \mathcal{R}(u_H)(y) + e(y) := \int B(x,y)u_H(x)dx + e(y) \quad (1)$$

where $u_L(y)$ is the low-resolution irradiance field, $u_H(x)$ the high-resolution irradiance field, $e(y)$ some noise and $B(x,y)$ is a blurring kernel. In general, ignoring domain boundaries, this kernel is assumed to be shift invariant and takes the form of a point spread function (PSF), i.e. (1) is a convolution equation $u_L = B * u_H + e$. Replacing images with image sequences, x is replaced by (x,t) where t is the time dimension of the sequence. The general numerical formulation which is the one normally used in SR problems can be written as

$$\mathbf{u}_L = R\mathbf{u}_H + E \quad (2)$$

where $\mathbf{u}_H \in \mathbb{R}^N$, $\mathbf{u}_L \in \mathbb{R}^n$ with $n < N$, R is a surjective linear map $\mathbb{R}^N \to \mathbb{R}^n$, and $E \in \mathbb{R}^n$ is a random vector. R cannot be inversed as $\text{rk}(R) < N$, and generally $\text{rk}(R) << N$ which makes this problem severely ill-posed (just as the

S. H. Keller (sunebio@diku.dk) is with the PET Center, Rigshospitalet (Copenhagen University Hospital), Blegdamsvej 9, DK-2100 Copenhagen, Denmark, phone +45 3545 1633, fax: +45 3545 3898. F. Lauze (francois@diku.dk) and M. Nielsen (madsn@diku.dk) are with The Image Group, Department of Computer Science, Faculty of Science, University of Copenhagen, Universitetsparken 1, DK-2100 Copenhagen, Denmark, phone: +45 3532 1400, fax: +45 3532 1401.

Manuscript number: TIP-05964-2010.R1.

continuous formulation was) even when ignoring the noise. Seen from the frequency analysis view point, the Nyquist-Shannon sampling theorem states that subsamling inevitably leads to a loss of high frequency information and thus a loss of details.

In order to recover HR images, extra information must be added. Usually two types of extra information can be combined: Prior information on the type of expected solutions and extra LR input images. The former relates to interpolation, while the latter constitutes the main idea behind SR, that is taking multiple LR images and fuse them into a HR image. These multiple sources may originate from multiple views or be successive frames from an image sequence.

It is interesting to note that supplying several low-resolution views for detail enhancement also happens in the human visual system (HVS). The input resolution on the retina is not as high as the resolution perceived by the HVS after it has processed the input. The eye constantly makes small, rapid eye movements (REM) supplying the HVS with a number of low-resolution views from which it can construct a more detailed high-resolution view. It has been debated for a long time whether these rapid eye movements do more than just stop the visual input from fading, but it has not been proven until recently by Rucci *et al.* [37] that the REMs are crucial to the super-resolution effect of the HVS.

The unsteadiness of a video camera as well as objects motion will generally give subpixel differences from frame to frame. From such a spatially sparse but temporally abundant data set, one should be able to mimic the HVS in creating higher resolution outputs. No matter if inspiration comes from vision, signal processing or both, it is the basic idea of super-resolution that one can increase the level of details somehow inferring knowledge over a number of time and/or space-shifted low-resolution samples of a scene. If one applies the right modelling in doing (video) super-resolution, one should be able to generate high-resolution images or image sequences that will please the HVS with a higher level of details or at least not disturb it with annoying artifacts.

In typical super-resolution producing just one HR image, the n LR views are registered to a common frame of reference before the HR output is generated. The registration is often simplified by using different known transformations and subsamplings of the same image thus simplifying the registration. If the LR input is an n-frame video sequence, the registration is typically done by computing the optical flow from each frame to the common frame of reference.

VSR can be implemented as an extension of single-frame SR: For each HR frame, define a support area of k LR frames used as super-resolution input for this HR frame and use a sliding window mechanism to produce the HR output sequence. This is computationally expensive so instead we compute the backward and forward flow fields between adjacent frame pairs. The iterative solver of our system then naturally propagates information from and to more distant frames. We also benefit from having increasingly more and more detailed HR frames and not just LR frames to draw information from as we produce the n HR frames simultaneously. Furthermore, we simultaneously update the optical flows of the sequence in high-resolution, which aids accurate detail propagation from frame to frame (to frame...).

The method we propose is derived from a cost function formulation that has been successfully used by the authors to address several image sequence problems: Image sequence inpainting [28], deinterlacing [24] and temporal super-resolution [25].

This paper is organized as follows. In Section II, we discuss related work on super-resolution and modelling aspects, including point-spread functions; In Section III, we introduce our cost function formulation and derive our variational motion-compensated VSR algorithm from it. Finally, we present our experiments and results in Section IV, discuss future work in Section V and conclude in Section VI.

II. Background and Related Work

Pioneering super-resolution, Tsai and Huang [41] modelled their approach of solving the super-resolution problem in the frequency domain. They, however, limited themselves to noise-free images and translational motion. Kim *et al.* extended the formulation of Tsai and Huang to include noise in [26]. A different family of methods emerged directly in the spatial domain where a maximum likelihood estimator (ML) would produce an HR output, which minimized the projection distances to all the sample LR images. Adding priors to these approaches will replace the ML estimator by a regularized ML or a maximum a posteriori (MAP) like estimator (see e.g. the work by Schultz and Stevenson [38]).

An extensive review of different approaches to solving the super-resolution problem is given by Borman and Stevenson in [4]. Chaudhuri collected another extensive bibliography in [8], and an overview including some of the most recent work is found in [40] by Shen *et al.*. In the work of Irani and Peleg [17], motion compensation (MC) is used to extract and register several frames from a given sequence and then create an HR image from them. Schultz and Stevenson [38] also integrate motion compensation as well as prior smoothness constraints more permissive for edges than the Gaussian constraint. Here too, the authors aim at reconstructing one HR frame from a video sequence input. Generalizations of these methods have been used for spatiotemporal super-resolution of video sequences by Shechtman *et al.* in [39]. From a set of low-resolution sequences they generate one high-resolution sequence (multi-camera approach). The recent video super-resolution algorithm by Farsiu *et al.* [11] (combined with demosaicing in [9]) models only affine or similar parametric optical flow and is a typical example of how modelling of the registration is simplified. Most often, the LR input frames are different elementary transforms (shift/rotation/skewing) and subsamplings of one HR image. Thus, the registration is no more than a use of exactly known transforms. Any super-resolution algorithm to be applied on real-world data needs a reliable and precise registration. In the case of video, the registrations become the complex task of computing optical flow (motion estimation).

Super-resolution has limits as shown by Baker and Kanade [1] and recently by Lin and Shum [29]. Good results (in terms of correct high frequency image content) are difficult, if not impossible, to obtain at large magnification factors, 1.6 is the practical limit and 5.7 the theoretical limit given in [29]. In order to overcome these limitations, good priors are needed. Baker and Kanade have proposed doing hallucination, which is to add a generative, trained model to the reconstruction. We want to be able to process any kind of video content, and thus we cannot use the method of Baker and Kanade as it is optimized on a subset of video and image data only. The same goes for the semi-generic, learned priors by Freeman *et al.* [13]. We thus need to develop a generic VSR algorithm which is able to handle arbitrary scene content and optical flows.

A. A Variational Formulation via Bayesian Inference

Our work relies on a Bayesian Inference framework for the recovery of image sequences and motion fields from which a maximum a posteriori (MAP) approach is derived in order to simultaneously compute HR flows and HR image sequences. A simpler version of our variational video super-resolution, building on the same framework but not computing HR flows, was presented in [20].

Several authors have used MAP approaches for jointly computing motion fields and a single HR image from an LR image sequence. In the recent work by Shen *et al.* [40], a flow-based object segmentation is even included in an iterative, cyclic scheme. The model by Shen *et al.* [40] only allows for perspective, parametric motion, the spatial prior is a Tikhonov one (that smooths across edges), and the number of moving objects needs to be known in advance, which makes it unsuitable for our purposes. Simultaneous registration and single HR image creation was also done earlier by Hardie *et al.* [15] but without considering multiple motions in the scene.

Since we cannot use the advanced learned spatial priors of e.g. Baker and Kanade [1], we have chosen to use total variation. More advanced generic regularization models are available, e.g. structure tensor-based methods, which are used for image interpolation (zoom, SR with just one LR input frame) by Tschumperlé and Deriche [43] and by Roussos and Maragos [36]. Tschumperlé and Deriche propose energy minimization without back projection as dictated by the image formation process in (1). Therefore Roussos and Maragos, who do include back projection in their model, get better results when comparing the two methods in [36]. The major disadvantage of structure tensor-based regularization is the computational cost: Recalculating the structure tensor in each iteration is costly, but does give better results than just doing spatial total variation-based regularization (as shown e.g. in [36]). Having temporal coherent frames available in a sequence of images (enabling us to do video super-resolution instead of image interpolation) gives a potential of detail enhancement which is not possible in image enhancement.

Total variation is nonlinear and thus more complex to use than the linear Gaussian distribution, but it will preserve and strengthen edges, i.e. be a source of much needed coherent high frequency information. In order to overcome the locality of differentiation, Farsiu *et al.* introduce in [11] a spatial bilateral total variation filter. They show better denoising performance than standard total variation on an artificial example (noisy text), but no comparisons of the two used on natural images are given in [11] neither for denoising nor for VSR.

So far, we have discussed scientific work on super-resolution focussing on getting as close as possible to the ground truth. In actual stand-alone video processors like the high-end products of Faroudja and DVDO and in built-in processors in high-end video devices (displays, DVD/Blu-ray-players etc.), focus is on visual quality as judged by the human observers. In these devices, the majority of the resources are typically spent on deinterlacing, noise filtering, correction of coding errors and color corrections. Unfortunately, bilinear interpolation is the standard method used for video super-resolution as it is cheap, easy to implement and does not create artifacts. The smoothing of bilinear interpolation is not an artifact severely unpleasing to the HVS like e.g. bad deinterlacing is. But as consumers get used to the quality of HDTV and Blu-ray discs, it will not suffice, and much better, real VSR will be needed.

B. Modelling the Point Spread Function

A very important datum for the super-resolution algorithm is the knowledge of the PSF, B in (1). We need to model how the light is dispersed through the camera lens and sampled on the recording medium, the sensor. The PSF is typically modelled by

$$B = B_{lens} * B_{sensor} \tag{3}$$

where $*$ is the convolution operator. To keep the modelling balanced between correctness and mathematical tractability, Gaussian or uniform (mean) distributions are typically chosen for the two terms. Lenses in cameras are of high quality and blurring is generally not a problem at video resolutions. In e.g. [3], it is discussed how it is most often the opposite, aliasing, due to lack of filtering before sampling on the CCD that is the problem in cameras. Thus, we leave out lens blurring by setting $B_{lens} = Id$ (the identity operator).

Choosing between Gaussian and uniform distributions for the sensor PSF, the uniform distribution is the obvious choice. The Gaussian mainly seems to be used when lens blur is included in the PSF model (e.g. in [9] and [39]). This is done to model web cameras and other cameras with really low quality lenses, or maybe to fit Gaussian downsampling of test data which is often done prior to running the SR algorithm. Downsampling is done to enable a comparison between the SR results and a known HR ground truth. A problem with the Gaussian is that one has to set the variance, but there is nothing in the PSF model that suggests a certain (generic) value, and one will have to measure the PSF camera specifically. The uniform sampling is fixed, and it is the distribution that most truthfully models the sampling across CCDs as pointed out by Barbe in [2]. Most digital video have been sampled using CCDs either when recorded with digital cameras or scanned from film using modern telecines (film scanners). Uniform PSF modelling is used for super-resolution in [1], [29], [35] and [38], and we will use it as well.

Roussos and Maragos [36] use the uniform distribution convolved with a Gaussian of large variation, which they claim helps remove blockiness (a.k.a. jaggedness or 'jaggies' in this case). This kind of restricted use of Gaussians might also remove some moiré and noise, but one will of course risk removing fine details.

Temporal integration or point spread in time to model the temporal aperture of the recording is typically left out of the modelling. In [42], Tschumperlé and Besserer use it to add film-like motion blur to video recordings when the two types of material is edited together (no VSR/SR done). Both Patti *et al.* [35] and Lin and Shum [29] assume the PSF to be uniform in time also, but Lin and Shum point out that it should be time integrated to remove motion blur. This is, however, difficult to do correctly as one needs to know the shutter time used in the recording. We also assume a temporally uniform PSF in our model, which allows motion blur to be left (almost) untouched as it is most likely a desired artistic effect of the film maker(s). With such a choice, the operator $\mathcal{R}$ from equation (1) becomes a moving average filter.

III. Variational Video Super-Resolution

In this section, we will go through the different aspects of designing and developing our algorithm for simultaneous computation of high-resolution image sequences and their optical flows. Some aspects have already been mentioned briefly, for instance our starting point, a Bayesian framework, from which we derive our algorithm.

A. Bayesian Framework for motion-compensated Image Sequence Upscaling and Restoration

The framework we present here was first formulated by Lauze and Nielsen in [28] to be used for simultaneous image sequence inpainting and motion recovery. We have since used it for deinterlacing [24], for temporal super-resolution (frame rate conversion) [25] and for a simpler (non-simultaneous) version of video super-resolution [20].

We wish to model the image sequence content and its optical flow using probability distributions. The locus of missing data given in [28] cannot be introduced in a way as straightforward in the super-resolution problem as it was in the blotch removal one (inpainting), but will have to be replaced by the high-to-low-resolution information loss process, the $\mathcal{R}$ operator introduced in (1). With this modification, we can use the same arguments as in [28] and get a posterior probability distribution for a pair $(u_H, \vec{v})$ of a HR image sequence and its motion field:

$$p(u_H, \vec{v}|u_L, \mathcal{R}) \propto \underbrace{p(u_L|u_H, \mathcal{R})}_{P_0}\,\underbrace{p(u_{H,s})}_{P_1}\,\underbrace{p(u_{H,t}|u_{H,s}, \vec{v})}_{P_2}\,\underbrace{p(\vec{v})}_{P_3} \quad (4)$$

where $u_{H,s}$ and $u_{H,t}$ are the spatial and temporal distribution of intensities respectively. On the left hand side, we have the posterior distribution which we wish to maximize (do MAP). The right hand side terms are: P_0, the image sequence likelihood; P_1, the spatial prior on image sequences; P_3, the prior on motion fields and P_2, a term that acts both as likelihood term for the motion field and as spatiotemporal prior on the image sequence. The term spatiotemporal does not denote the spatial plane and the purely orthogonal temporal component, which is a commonly used, stringent definition of spatiotemporal in a 3D sense. We consider an image sequence to be 2D + 1D, time cannot be juxtaposed with the third spatial dimension as step sizes cannot be said to be the same in time and space. More importantly, with motion in the sequence, the relevant information is found along the motion trajectories, thus we define spatiotemporal as the 2D spatial neighborhood in combination with the information rich time dimension located along the optical flow field.

Noise comes from various sources at image acquisition time. We wish to preserve film granularity, and since we did not encounter any noise problems in our tests, we take $e(y) = 0$ in (1). Then the term $p(u_H|u_L, \mathcal{R})$ becomes a Dirac $\delta_{\mathcal{R}u_H - u_L}$, that is, the super-resolution constraint is $\mathcal{R}u_H = u_L$.

B. From MAP to Variational Energy Minimization

We use the Bayesian to variational rationale by Mumford [32], $E(x) = -\log p(x)$, to get to a variational formulation of our problem and replace MAP with an energy minimization. As discussed above, our original super-resolution constraint is $\mathcal{R}u_H = u_L$. This means that the optimal pair $(u_H, \vec{v})$ minimizes the constrained problem

$$\begin{cases} E(u_H, \vec{v}) = E_1(u_{H,s}) + E_2(u_{H,s}, u_{H,t}, \vec{v}) + E_3(\vec{v}) \\ \mathcal{R}u_H = u_L. \end{cases} \quad (5)$$

Applying calculus of variations, a minimizing pair $(u_H, \vec{v})$ must under mild regularity assumptions be a zero of the energy gradient $\nabla E(u_H, \vec{v}) = 0$. The optimized solution is expressed by the coupled system of equations

$$\begin{cases} \nabla_u E(u_H, \vec{v}) = 0 \\ \nabla_{\vec{v}} E(u_H, \vec{v}) = 0 \end{cases} \quad (6)$$

where $\nabla_u E = 0$ ($u = u_H$, subcript left out for readability) is subject to the constraint $\mathcal{R}u_H = u_L$, the projection back onto the true solution hyperplane. There is no back projection of the flow, $\mathcal{R}\vec{v} = \vec{v_L}$, as there is no ground truth (LR) flow $\vec{v_L}$ governing what the true solution hyperplane is. But the HR flow $\vec{v}$ depends on u_H, and the projection $\mathcal{R}u_H = u_L$ links it to the only known ground truth, u_L.

C. Variational Video Super-Resolution and Optical Flow

We need to "instantiate" the generic terms in the energy formulation (5). For the spatial regularity measure E_1, we choose, as mentioned in previous sections, total variation $E_1(u_H) = \int |\nabla u_H| dx$ (∇ will denote the *spatial* gradient in the sequel). Brox *et al.* have proposed a very high quality variational optical flow algorithm in [5] (detailed description given in [34]). We will use their energy for the terms E_2 and E_3. Let us introduce some notations first: v_1 and v_2 are the x- and y-components of the flow field, i.e. $\vec{v} = (v_1, v_2)^T$. $V = (\vec{v}^t, 1)^T$ is its spatiotemporal counterpart. $J\vec{v}$ will denote the spatio-temporal Jacobian of $(x, t) \to \vec{v}(x, t)$. $\|J\vec{v}\|_F^2$ will be its Frobenius squared norm $|\nabla_3 v_1|^2 + |\nabla_3 v_2|^2$. $\psi(s^2) = \sqrt{s^2 + \varepsilon^2}$ is a strictly convex approximation of the L^1-norm function, regularizing it around the origin, where it is non-differentiable, ε being a small positive constant. ∇_3 denotes the spatiotemporal gradient. We define the directional or Lie derivative along the vector field $\vec{V}$ of a function f by

$$\begin{aligned}\mathcal{L}_{\vec{V}} f &:= \lim_{h \to 0} \frac{f(\mathbf{x} + h\vec{v}, t+h) - f(\mathbf{x}, t)}{h} \\ &= \nabla f \cdot \vec{v} + f_t \\ &= \vec{V}^T \nabla_3 f \\ &\approx f(\mathbf{x} + \vec{v}, t+1) - f(\mathbf{x}, t) \end{aligned} \tag{7}$$

and we extend it component-wise for vector-valued functions such as ∇u_H. The above proposed approximation will *always be used* in the sequel.

Following [34], the part of the energy containing motion, $E_2 + E_3$, is

$$\underbrace{\lambda_2 \int \psi(\left|\mathcal{L}_{\vec{V}} u_H\right|^2 + \gamma \left|\mathcal{L}_{\vec{V}} \nabla u_H\right|^2) dx}_{E_2} + \underbrace{\lambda_3 \int \psi(\|J\vec{v}\|_F^2) dx}_{E_3}$$

with x running over the whole domain of u_H. The λ_i's and γ are some positive constant weights. Using the same regularization ψ for the spatial total variation term (with $\varepsilon = 10^{-4}$ in our experiments), we obtain the following energy from (5):

$$\begin{cases} E(u_H, \vec{v}) = \underbrace{\lambda_1 \int \psi(|\nabla u_H|^2) dx}_{E_1} \\ \qquad + \underbrace{\lambda_2 \int \psi(\left|\mathcal{L}_{\vec{V}} u\right|^2 + \gamma \left|\mathcal{L}_{\vec{V}} \nabla u_H\right|^2) dx}_{E_2} \\ \qquad + \underbrace{\lambda_3 \int \psi(\|J\vec{v}\|_F^2) dx}_{E_3}, \\ \underbrace{\mathcal{R} u_H = u_L}_{E_0}. \end{cases} \tag{8}$$

E_1 is a regularization of the spatial total variation measure and provides a non linear spatial diffusion term in the corresponding Euler-Lagrange equation. E_3 similarly provides a spatiotemporal diffusion of the flow values. E_2, which acts as the spatiotemporal prior on the intensities and as a data term on the flow, is more complex incorporating both the brightness constancy assumption as well as the spatial gradient constancy assumption (GCA), which sharpens motion boundaries and lowers sensitivity to changes in brightness (change of lighting, motions in and out of regions in shadow).

We split the energy in (8) in two parts, $E_I(u_H)$ and $E_F(\vec{v})$, to minimize it according to (6). For the flow, we then get this energy to be minimized:

$$\begin{aligned} E_F(\vec{v}) := &\underbrace{\int \psi(\left|\mathcal{L}_{\vec{V}} u_H\right|^2 + \gamma \left|\mathcal{L}_{\vec{V}} \nabla u_H\right|^2) dx}_{E_2} \\ &+ \underbrace{\lambda_3 \int \psi(\|J\vec{v}\|_F^2) dx}_{E_3}. \end{aligned} \tag{9}$$

According to the survey by Bruhn *et al.* in [7] this energy yields one of the most precise optical flow algorithms (and its worth noting that simpler versions of variational optical flow have been shown to run real time on standard PCs by Bruhn *et al.* in [6]). For details on the Euler-Lagrange equation of (9), we refer to [34] and the thesis [27].

The intensity energy to be minimized is

$$\begin{cases} E_I(u_H) := \underbrace{\lambda_s \int \psi(|\nabla u_H|^2) dx}_{E_1} \\ \qquad + \underbrace{\lambda_t \int \psi(\left|\mathcal{L}_{\vec{V}} u_H\right|^2 + \gamma |\mathcal{L}_{\vec{V}} \nabla u_H|^2) dx}_{E_2}, \\ \underbrace{\mathcal{R} u_H = u_L}_{E_0}. \end{cases} \tag{10}$$

Setting $A = \psi'(|\nabla u_H|^2)$ and $B = \psi'(|\mathcal{L}_{\vec{V}} u_H|^2 + \gamma |\mathcal{L}_{\vec{V}} \nabla u_H|^2)$, the Euler-Lagrange equation of (10) is

$$\begin{aligned} &\nabla_u E_I = -\lambda_s \mathrm{div}_2(A \nabla u_H) - \\ &\lambda_t \left(\mathrm{div}_3(B(\mathcal{L}_{\vec{V}} u_H)\vec{V}) + \gamma\, \mathrm{div}_2 \begin{pmatrix} \mathrm{div}_3(B(\mathcal{L}_{\vec{V}} u_{H,x})\vec{V}) \\ \mathrm{div}_3(B(\mathcal{L}_{\vec{V}} u_{H,y})\vec{V}) \end{pmatrix} \right) \\ &= 0 \end{aligned} \tag{11}$$

where div_2 and div_3 are the 2D and 3D divergence operators respectively. Equation (11) involves 4th order terms, and its numerical solution is computationally very heavy. Since we already have well-segmented flow fields with sharp motion boundaries from using the GCA in $E_F(\vec{v})$, it will most likely not lift the output quality if used in the intensity part as well. Imagine an object moving into a darker region (e.g. a car driving from the sun into the shadow) and take a point p on the object which is in the sun in frame number 1 and in the shadow in frames 2 and 3. Temporal diffusion at p in frame 2 along the flow (backwards and forwards) using just the brightness constancy assumption will mainly come from frame 3 as an temporal edge is detected between frame 1 (light) and 2 (shadow) at p. Adding the GCA will force the gradient to be diffused from both frame 1 (light) and 3 (shadow), which *might* lead to a *slight* increase in details but might also force an incorrect change in intensity value at p depending on the weight γ in E_I. The GCA has already done its work in the flow energy minimization creating high quality flows. (Further discussions on the topic can be found in [21].) We have thus chosen to set $\gamma = 0$ in E_I (while of course keeping it $\neq 0$ in E_F). This choice introduces a theoretical inconsistency with respect to the variational model, but as it will be demonstrated in our experiments, it provides excellent results while keeping numerical complexity reasonable.

The resulting constrained equation is

$$-\lambda_s \mathrm{div}_2(A \nabla u_H) - \lambda_t \mathrm{div}_3(B(\mathcal{L}_{\vec{V}} u_H)\vec{V}) = 0, \quad \mathcal{R} u_H = u_L$$

with $B = \psi'(|\mathcal{L}_{\vec{V}} u_H|^2)$ this time. Note then that the last divergence can be written, after an elementary computation, as

$$\mathcal{L}_{\vec{V}} \left(B \mathcal{L}_{\vec{V}} u_H \right) + B \left(\mathcal{L}_{\vec{V}} u_H \right) \mathrm{div}_2 \vec{v}.$$

The first part is a "pure" flow line, non-linear diffusion term, while the second term compensates for divergence of the flow lines. When zooming in or out, intensity conservation must imply a transport of intensity "mass" (see the work of Florack *et al.* [12] for a closely related topic), and this term can indeed be interpreted as an advection/transport of the intensity with velocity $B(\mathrm{div}_2\vec{v})\vec{V}$ since $\mathcal{L}_{\vec{V}}u_H = \vec{V}\cdot\nabla_3 u_H$. Optical flows computed on natural image sequences have generally small divergence because of object rigidity (at least for not too large velocities), and the GCA clearly has a tendency to amplify it. We thus choose to ignore the advection part and solve the constrained equation, the Euler-Lagrange equation of E_I

$$-\lambda_s \mathrm{div}_2(A\nabla u_H) - \lambda_t\left(\mathcal{L}_{\vec{V}}\left(B\mathcal{L}_{\vec{V}}u_H\right)\right) = 0, \quad \mathcal{R}u_H = u_L. \tag{12}$$

D. Simultaneous Optimization of high-resolution Flows and Intensities

We want to simultaneously compute the flow and intensities to benefit from better and better versions of both in our iterative scheme. It makes no sense to introduce simultaneousness to VSR by using multiresolution as in inpainting [28], unless we want to apply really large magnification factors and can benefit from having the intermediate scales of multiresolution between LR and final HR resolutions. In our simultaneous VSR algorithm as given in this paper, we will therefore iterate between minimizing each of the two parts in (6) at high-resolution directly to simultaneously provide better and better versions of both. This requires a careful control of the iterative process as discussed in Section IV-E on parameter tuning.

E. Discrete Formulation of the Super-Resolution Constraint

The formulation we have developed until now is continuous. In Section II-B, we discussed the choice for the PSF of the sensor (CCD) and stated that we would use the uniform distribution and that the continuous filter $\mathcal{R}$ from equation (1) is a spatiotemporal moving average filter. We proceed to describing the sampling operations in more details.

We take the image spatial domain to be the standard square $(0,1)^2$, the temporal axis to be $[0,1]$ and the complete spatiotemporal domain to be $\Omega = (0,1)^2 \times (0,1)$. We assume the frame rate to be fixed as we only perform spatial (re)sampling. Let T be the number of frame of the image sequence. For the high-resolution sampling, assume a spatial grid of size $M_H \times N_H$ and for low-resolution a grid of size $M_L \times N_L$ with $M_L < M_H$, $N_L < N_H$. The different grid steps sizes are: $h_x^H = 1/M_H$; $h_y^H = 1/N_H$; $h_x^L = 1/M_L$; $h_y^L = 1/N_L$ and $h_t = 1/T$. Given an image sequence u, we define its high-resolution sampling $\mathbf{u}_H$ by

$$(\mathbf{u}_H)_{ijk} = \frac{1}{h_x^H h_y^H h_t}\iiint_{C_{ijk}} u\,dx\,dy\,dt$$

where C_{ijk}, the fine grid cell where the averaging is performed, is the cell $(0,h_x^H)\times(0,h_y^H)\times(0,h_t)$ translated from position $(0,0,0)$ to (ih_x^L, jh_y^L, kh_t) and $i = 0\dots M_H-1$, $j = 0\dots N_H-1$, $k = 0\dots T-1$. Similarly, the low-resolution sampling $\mathbf{u}_L$ of u is given by

$$(\mathbf{u}_L)_{abk} = \frac{1}{h_x^L h_y^L h_t}\iiint_{D_{abk}} u\,dx\,dy\,dt$$

where D_{abk}, the coarse grid cell where the averaging is performed, is the cell $(0,h_x^L)\times(0,h_y^L)\times(0,h_t)$ translated from position $(0,0,0)$ to (ah_x^L, bh_y^L, kh_t) and $a = 0\dots M_L-1$, $b = 0\dots N_L-1$, $c = 0\dots T-1$.

The discrete low-resolution operator R is a discrete moving average that implements the following idea: Given a coarse grid cell, D_{abc}, it is covered by fine grid cells, and the value of $R\mathbf{u}_H$ at this large grid cell should be the average of the values of $\mathbf{u}_H$ at the covering fine grid cells weighted by volume overlap.

For one dimensional signals, $\mathbf{u}_H \in \mathbb{R}^M$, $\mathbf{u}_L \in \mathbb{R}^N$ with $N < M$, the corresponding transformation $R_M^N : \mathbb{R}^M \to \mathbb{R}^N$ can be decomposed as follows: Let $\mathrm{lcm}\,(M,N) = L$ be the least common multiple of M and N. Then R_M^N can be decomposed as a *replication* step, $\mathbb{R}^M \to \mathbb{R}^L$, where each component is replicated L/M times, which is followed by an *averaging* step, $\mathbb{R}^L \to \mathbb{R}^N$, where consecutive blocks of L/N entries are replaced by their average, leading to the final N values.

As $\mathrm{lcm}\,(M,M) = 1$, it is clear that R_M^M reduces to the identity transform in that case. For higher dimensional inputs, both the fine and coarse grid cells are aligned with the axes (R is *separable*), and thus R can be applied by cascading 1D-transforms. In our case, R can be decomposed into

$$R = \underbrace{R_{M_H}^{M_L}}_{columns} \otimes \underbrace{R_{N_H}^{N_L}}_{rows} \otimes Id_T \tag{13}$$

where Id_T is the identity map of $\mathbb{R}^T$ (no temporal averaging) and $\otimes$ is the Kronecker product.

F. Numerical Solution with Super-Resolution Constraint

As discussed in [28], it is natural to introduce both forward and backward motion fields for the discretization of the Euler-Lagrange equation derived for E_F in (9). We compute them using the method proposed in [34] and detailed in [27]. The resolution of the Euler-Lagrange equation for the intensities in (12) follows the same principles: We run an outer loop in which we use a fixed point scheme to freeze the nonlinear components, A and B in (12), leading to a linear system of equations. It is solved iteratively by a modified Gauss-Seidel method that incorporates the super-resolution constraint $R\mathbf{u}_H = \mathbf{u}_L$ where $\mathbf{u}_L$ is the observed low-resolution image sequence.

In order to enforce the numerical super-resolution constraint $R\mathbf{u}_H = \mathbf{u}_L$, we proceed as follows: Assume that $\mathbf{u}_H^n$ is our current estimate of the high-resolution image, and that it *satisfies* the constraint. By one or more iterations of the Gauss-Seidel solver, we obtain an update $d\mathbf{u}_H^{n+1}$. We project *orthogonally* this update into a $\overline{d\mathbf{u}_H}^{n+1}$ in the null-space of R so that $\mathbf{u}_H^{n+1} = \mathbf{u}_H^n + \overline{d\mathbf{u}_H}^{n+1}$ will satisfy the super-resolution

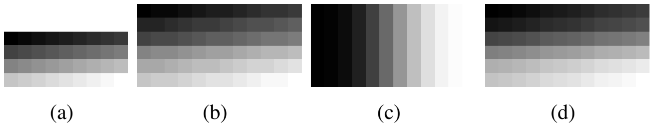


Fig. 2. The importance of the super-resolution constraint. (a) input, (b) HR initialization, (c) VSR without SR constraint destroys the image content, and (d) preservation of true content when running VSR with the SR constraint. The HR initialization (b) and the result of doing VSR with the SR constraint (d) are very similar as there are no sharp edges in this example.

constraint. Using elementary calculations, this orthogonal projection is given by

$$\overline{d\mathbf{u}}_H^{n+1} = d\mathbf{u}_H^{n+1} - R^{\dagger} R d\mathbf{u}_H^{n+1} \tag{14}$$

where $R^{\dagger} = (R^{*}R)^{-1} R^{*}$ is the Moore-Penrose pseudo-inverse of R, R^{*} being the adjoint (transpose) of R (see [14]). A detailed implementation description for those interested is given in the thesis [21], and the code for our algorithm is available as part of the online material for this paper at http://www.image.diku.dk/sunebio/VSR/VSR.zip [23].

IV. Experiments

A. The Importance of the Super-Resolution Constraint

We have stressed the importance of using the super-resolution constraint, and here we give an example of why it is a good idea to add this extra complexity to an already complex regularization scheme. In Fig. 2(a), we have shown one of the four identical frames in a small sequence. It is a 4×9 matrix filled line by line with the values 1 to 36. Fig. 2(b) shows the 6×12 HR frame resulting from initialization with the scheme given in (13). In Fig. 2(c), we see how pure regularization without the SR constraint destroys the image content, whereas it is preserved in Fig. 2(d) when we do orthogonal projection governed by the SR constraint as given in (14). In tests running VSR on real sequences without using the SR constraint, we ended up with cartoon-like frames, which is typical of 2D spatial total variation run for too long on a natural image.

B. Subjective and Objective Evaluation

We have used both subjective and objective evaluation of our results but have focused on the subjective evaluation to best mimic the perceptive evaluation in the human visual system (HVS). Our subjective evaluations have been done by ourselves and in some cases by other image (sequence) processing experts, but a large set of results are given online for the reader to verify our claims [23].

Discussions on objective vs. subjective evaluation have been given by Bellers and de Haan [3], Kanters [19], Keller [21] and Nadenau *et al.* [33]. In short, objective measures can give an idea about the subjective quality and in some limited cases (small data set and/or a specific problem), the two measures might give very similar results. In [24], we have shown how the objective evaluation by mean square error (MSE) does not correlate with subjective evaluation for the case of deinterlacing.

The above discussion covers the evaluation of the output HR image sequences. Evaluation of optical flow quality in literature is typically done by testing the algorithm on artificial sequences with known ground truth flows and measuring the angular error between ground truth and computed flows (see for instance [7]). In our tests, we have used real video sequences, and have been left to do visual evaluation of the quality of the flows. To find the optimal parameter settings for our motion algorithm, we have used both flow field visualization and quality assessment of the main VSR output, the HR image sequences.

C. Benchmarking

We have tested our algorithm against bilinear and bicubic interpolation as they are the methods mostly used for frame upscaling in video systems and against our earlier nonsimultaneous variational VSR from [20]. We have also included a comparison to the method from [11] on some of our data as software is readily available online. It should be noted that the method from [11] does not handle complex (non-global) motion just as other VSR algorithms found in literature. It would only have been fair to include other advanced (motion-compensated) super-resolution algorithms in our test. The main reason why we have not done so are the limitations in what types of video content (e.g. faces or text as in [1]) they can be applied to as it was already discussed in Section II. If someone should want to do a benchmarking of VSR, we have made our source code (and Matlab mex-function dll's) available online [23].

D. Test Material

We aim at applying our VSR algorithm either in end user home video/entertainment systems or in broadcast video scaling systems. Therefore, we have chosen to conduct our tests using standard video material, more specifically PAL DVDs telecined from film. We use a set of 19 sequences, 5-15 frames long, that have been selected to be challenging in terms of detail level and motion complexity. We work only on the luminance channel (8 bit, [0-255]) of the test sequences. As the HVS is less sensitive to details in the color channels than in the luminance channel [30], the two chroma channels are already subsampled in practically any broadcasting or storage system today, and thus simple bilinear interpolation can be used here – at least at lower magnification factors. Of course our algorithm can be applied to the C$_r$ and C$_b$ color channels as well or on an RGB version of the sequence. Some ideas on how to do coupled processing of the three channels are given by Tschumperlé and Deriche in [43].

E. Parameters

As with almost any other image/video processing algorithm, we have a number of parameters that need to be tuned. We have run extensive tuning tests but with nine free parameters, it is of course not complete. Testing just three different settings of each parameter in all possible combinations would result in $3^9 = 19,683$ different test results for evaluation. Thus, we have relied on our common sense and our experience with variational methods for inpainting [27], [28], deinterlacing

[24], frame rate conversion [25] and prior work on simple VSR [20] to get the parameters optimized for general use in VSR. The parameter settings given in this section are the ones used to get all our presented tests results except the purely illustrative experiment with flow parameters given in Fig. 4.

The initial low-resolution flow is calculated running 10 fixed point iterations each with 40 inner relaxation iterations. 5 times 20 iterations or even less will give the same results visually but to be on the safe side, we have run 10 times 40 iterations in our tests. The multiresolution pyramid has 100 levels with a coarse-to-fine scale factor of 1.04. The weight on the gradient constancy assumption in (9) is $\gamma = 200$, and the weight of the smoothing (prior) is $\lambda_3 = 70$.

The actual VSR algorithm runs 10 outer overall iterations. For both the flow and intensity calculations respectively, we run 1 fixed point iteration with 5 relaxation iterations in each overall outer iteration. For the flow, the GCA and smoothing weights are $\gamma = \lambda_3 = 100$ in (9) and for the intensity calculations, the spatial and temporal weights in (12) are $\lambda_s = \lambda_t = 1$. In all computations, the convergence threshold is set to 10^{-7} (and never reached).

On our way to the optimal parameters for the actual VSR algorithm given here, we have made a few interesting discoveries, which we will discuss here. Increasing the number of outer overall iterations does not give any improvements, while lowering the number from 10 and anywhere down to 5 can give just as good results, but 10 is the failsafe setting. The algorithm is fairly sensitive to changes in the number of fixed point and/or relaxation iterations. Iterating too much on either the flow or the intensities stops the other from evolving further: Probably, a local minimum is reached. Lowering the number of inner iterations causes a slowdown in convergence (the system is not sufficiently relaxed). Thus, the number of inner iterations should not be set too high as it might cause a loss of details. Setting it (too) low will give slower convergence but cause no harm.

Changing γ and λ_3, the GCA and smoothing weights of the flow respectively, mainly changes how homogeneous the flow is, but larger changes from the optimal settings result in either too smooth or too detailed intensity outputs (artifact-like details or oversharp edges judged unnatural by the viewer).

Increasing the spatial diffusion by turning up λ_s gives smoother results similar to the ones obtained with our non-simultaneous VSR algorithm [20] (comparisons given later in Section IV-H). Turning up λ_t has no effect on some sequences and on others it slows down development away from the jagged initializations. It seems that the implicit weighing in the variational algorithm is enough to ensure optimal temporal diffusion and pushing it too hard with high λ_t-values is unnecessary or even has a negative effect. We have also experimented with changing λ_s and λ_t over time (e.g. eight outer iterations with $\lambda_s = \lambda_t = 1$ and two with $\lambda_t = 5$) and have got minor improvements on some sequences, whereas the same settings failed on other sequences. Thus, finer parameter tuning might slightly improve some results, but the settings given above ensures optimal or very close to optimal results on all the data that we have tested on.

F. Running Times

We have not focused on optimizing our code for speed (yet). The code is written in C++ using the CImg library (http://cimg.sourceforge.net/) interfacing with Matlab. The initial flow computations (backward and forward) takes 1-4 hours on a (slightly outdated) standard PC (Pentium 4 2.4/2.8 GHz, 2-4 GB RAM) depending on the number of frames. The majority of the time is spent on initial multiresolution LR flow computations. The running time of the VSR algorithm when doing 2x2 magnification from 576×720 SD PAL resolution is app. 19 seconds per outer iteration per frame (so typically 190s per frame with 10 outer iterations). From 576p SD PAL to 720p HD (720×1280), the running time is app. 13 seconds per outer iteration per frame on the same PC. The number of HR pixels being processed is 1.7 times higher in the 2x2 case, but the processing time is only 1.45 times higher, which shows that the more complex back projection in the SD to HD case (80 corrections per pixels contra 4 in the 2x2 case) does give a minor overhead in computation time. It is clear that the need for a speedup is the greatest in the initial flow computations but from [6], we know that simpler variational flow algorithms run in real-time on standard PCs (although on web camera resolution video) and could possibly be used for initial flow computations in VSR.

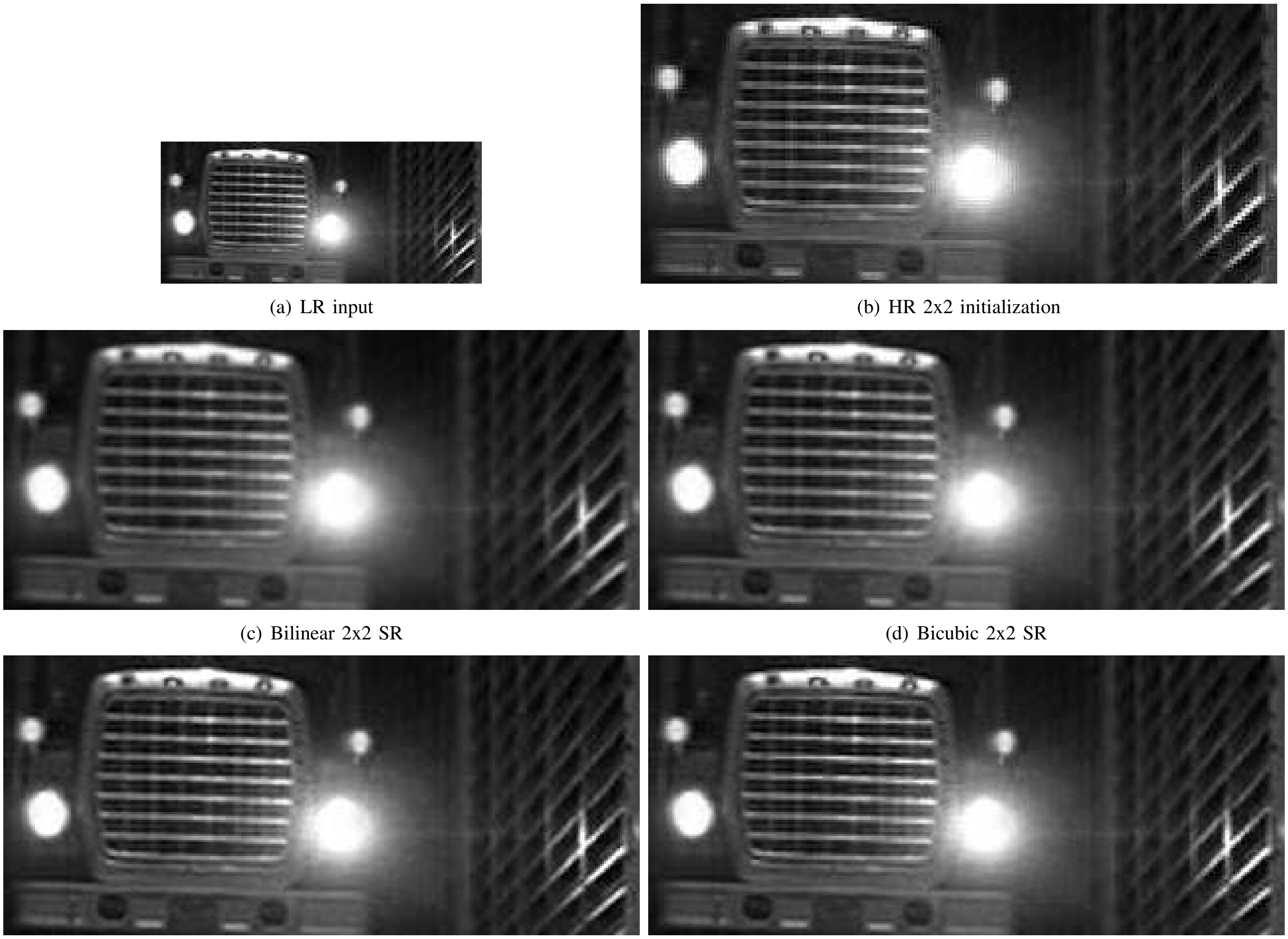
G. Online Material and Correct Viewing of Results

Selected test results are available as video (*.avi) and electronic stills (*.bmp) online at: http://www.image.diku.dk/sunebio/VSR/VSR.zip [23]. The printing process will often blur figures, so the results given as figures in this paper are best viewed on-screen and in some cases with a certain zoom (given in the captions). In the online appendix of this paper [22], the figures are given at the recommended zooms. We still recommend on-screen viewing of the appendix pdf-file (zoom set to 100%) but when available view the bitmap files, which give true 1:1 resolution relation between image and screen. The program `Virtual Dub` that displays avi (and bmp) files at their true 1:1 resolution is included in the zip file [23].

H. Results: 2x2 and SD to 720p VSR

In this section, we focus on subjectively evaluating classic 2x2 magnification and 576p SD PAL to 720p HD VSR (576×720 to 720×1280) VSR results. In the next section, IV-I, we will also give objective results and compare to the algorithm from [11]. Finally, in section IV-J, we will evaluate the results of 4x4 and 8x8 VSR.

Results for 2x2 SR/VSR on the sequence `Truck` is given in Fig. 3 (and zoomed versions in Figures 13 and 14 in the appendix of this paper [22]). The initialization in 3(b) is very jagged (or blocky). Bilinear interpolation produces a very smooth result shown in 3(c), bicubic interpolation produces a clearly sharper result as seen in 3(d) but the output of our two variational VSR methods in 3(e) and 3(f) are even sharper. The new simultaneous VSR (S-VSR) produces a sharper result than the nonsimultaneous VSR from [20], but the quality difference between the two is not as big as



(a) LR input

(b) HR 2x2 initialization

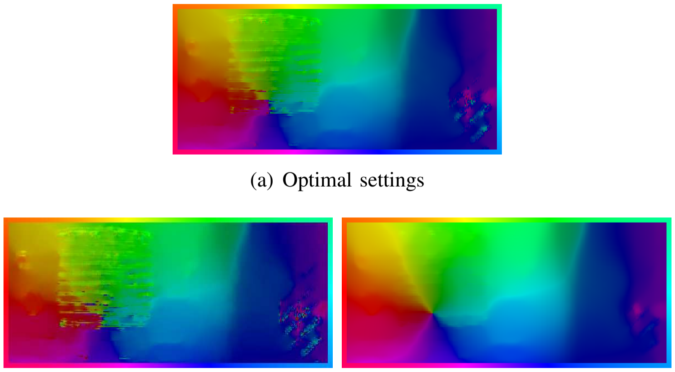
(c) Bilinear 2x2 SR

(d) Bicubic 2x2 SR

(e) Nonsimultaneous 2x2 VSR from [20]

(f) Simultaneous 2x2 VSR (S-VSR)

Fig. 3. 2x2 VSR on the sequence `Truck`. Sharpness increases step by step from (c) through to (f) as seen most clearly when viewed on-screen and zooming to 125% or more (as in Figures 13 and 14 in the online appendix of this paper [22]). The motion in `Truck` is zoom-like as the truck drives towards the camera. A 70 × 160 pixels cutout of the LR input is shown in (a) and the HR cutouts (b)–(f) are 141 × 321 pixels.



(a) Optimal settings

(b) Oversegmented

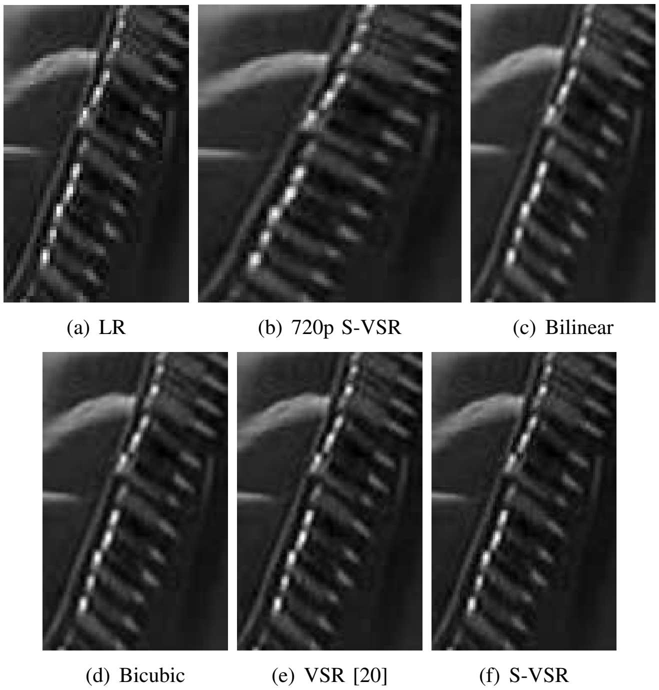
(c) Oversmoothed

Fig. 4. Flows from 2x2 VSR on the sequence `Truck` shown in Fig. 3. The flow directions are given by the hue value on the border, and the magnitude by the intensity (see online color version of this paper).

the differences from bilinear to bicubic or from bicubic to nonsimultaneous VSR. The video versions of Figures 3(c), 3(e) and 3(f) are given in the online material [23] in the folder Truck_2x2_oldAndNewVSRandBilinear.

The flow produced using the settings given in Section IV-E is shown in Fig. 4(a). In Fig. 4(b), we see how lowering the weight on the smoothing of the flow from $\lambda_3 = 100$ to $\lambda_3 = 70$ makes the flow a bit oversegmented, resulting in artifacts in the intensity result (bright, unnatural horizontal lines and single spikes on the front grill). The very smooth flow resulting from setting $\lambda_3 = 250$ shown in Fig. 4(c) seems more correct (i.e. the grill line flows are now part of the overall zoom and not segmented independently), but the intensity output becomes a bit too smooth.

We see a gradual improvement in sharpness from bilinear interpolation over bicubic interpolation and nonsimultaneous VSR to simultaneous VSR on all 19 sequences in our test when doing 2x2 magnification as with `Truck` discussed above. For some sequences like `Bullets` shown in Fig. 5, the differences are less significant. While the sequence `Truck` appears very sharp in its LR version, `Bullets` appears less sharp in LR: Overall the differences between the results from the different methods are more significant in the test sequences containing details to begin with, as there is simply more infor-



(a) LR (b) 720p S-VSR (c) Bilinear

(d) Bicubic (e) VSR [20] (f) S-VSR

Fig. 5. VSR on the sequence `Bullets`. To illustrate the increase in detail level with higher resolution, the LR input (a), the 720p VSR result (b) and the 2x2 results in (c) to (f) are shown at the same height. As with `Truck`, we have increasing sharpness step by step from (c) to (f), although less significant than for `Truck`. Results are best viewed on-screen zoomed to ca. 150%, which is the size they are given at in Fig. 15 of the online appendix [22]. The 720p result is shown at its correct aspect ratio whereas the 1:1.422 PAL anamorphic widescreen pixels of the LR input and 2x2 results are shown as 1:1 pixels. Cutout sizes are: 90×56 (LR), 113×100 (720p) and 179×111 (2x2).

mation available for temporal (and spatial) diffusion/transport and less risk of ending up in local minima due to already smooth regions of image data. When doing 576p SD to 720p HD VSR, the differences between the algorithms are smaller, the exception being that bilinear always perform much worse than the three other algorithms. Comparing the 720p result on `Bullets` in Fig. 5(b) with the 2x2 result in Fig. 5(f) shows how higher pixel density carries more information and gives room for larger improvements (increased magnification factors at constant display size, e.g. more pixels at the same screen size).

For the sequence `Boardwalk`, Fig. 6(b) shows how simultaneous VSR removes the blockiness seen in the LR input in Fig. 6(a). Fig. 6(c) shows how bilinear interpolation removes the blockiness at the price of smoothing. The Figures 6(d) and 6(e) show just how big the gain in detail from SD to 720p HD can actually be, here on `Straw Hat`.

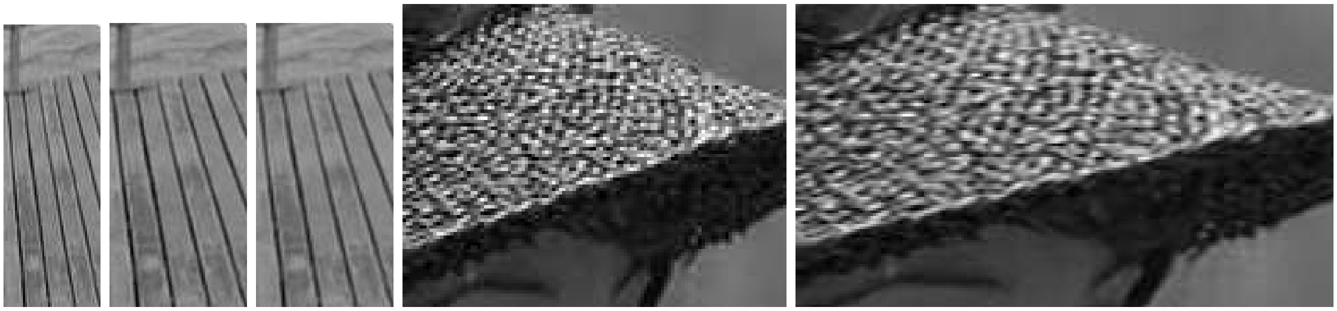
The figures given in this paper do not give the full picture of the differences between the different algorithms as the outputs should be seen as video at large viewing angles. Local gains in sharpness can to a large extend be evaluated on stills, but the sense of overall gain in sharpness of full frames (720×1280 or 1152×1440) is hard to portray in printed figures. To really determine how big an advantage the gain in sharpness is, a test with longer sequences should be conducted under realistic viewing conditions and on large screens, preferably according to the subjective quality evaluation standard ITU-R Rec. 500 [18] or similar. The video results in [23] are short but can at least be viewed at a large viewing angle.

Another improvement which can only be seen when viewing the outputs as video, is the decrease in flicker. It is (almost) impossible to compare differences in flicker between LR and HR versions of a video as they cover different areas of a given screen, and flicker perception is highly dependent on the size of the image projected onto the eye (see Matlin and Foley [31] or Keller [21]). But we can compare the results of different HR algorithms. A video example is found in the folder Boardwalk2_720p_FlickerReduction of the online material [23], where bilinear, bicubic and simultaneous VSR results are given in 720p HD for the sequence `Boardwalk`. On large and bright displays, flicker reduction is a major quality improvement. The results produced using bicubic interpolation flickers the most, while our nonsimultaneous VSR does significantly better and is close to having as little flicker as the two best algorithms here, simultaneous VSR and bilinear interpolation. Bilinear interpolation smoothes out too many details and edges, while the temporal regularization in simultaneous VSR removes just as much flicker but preserves details, sharpness and the film granularity while doing so. (As with motion blur, film granularity is considered an artistic quality of the film, and thus should be preserved.) There is no doubt that simultaneous VSR produces the best results in 576p SD to 720p HD conversion. It is also best at 2x2 VSR where the quality differences in the results from the different algorithms are more significant. To stress this point, we did 2x2 simultaneous VSR on a 25 frame sequence provided to us by the film post production company Digital Film Lab. As film industry professionals, they evaluated our VSR result to be a lot better than anything they would be able to produce with any of their professional post production and editing systems (e.g. Da Vinci systems).

I. Down and Up Again: Objective Results and Comparison to Another VSR Method

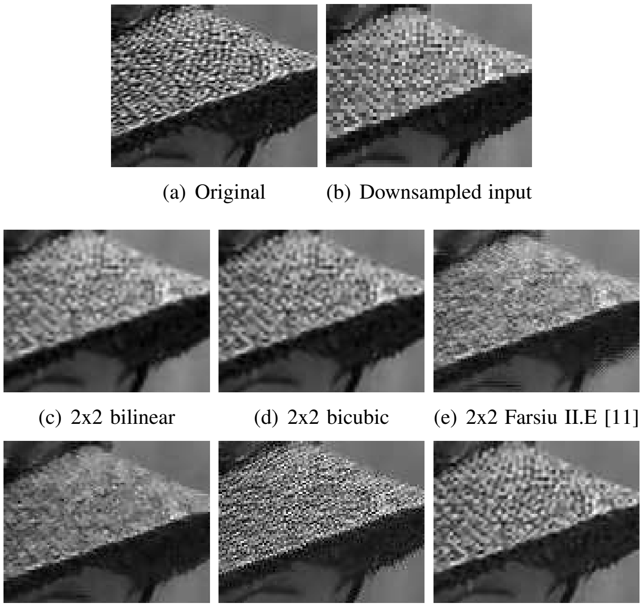
A typical method used in the evaluation of (V)SR algorithms is the following: Before upscaling, the input image (sequence) is downscaled by the inverse of the magnification factor(s), such that there exists a ground truth to compare the upscaling results to. A potential problem by doing 'up and down' tests is that the scheme used for downscaling can effect the final upscaling results. Typically, a Gaussian blur kernel is used prior to downsampling, and the chosen variance of this kernel defines the balance between detail preservation and aliasing. One can argue that using the Gaussian for downsampling will model the image formation process given in (1), but as we have discussed earlier, modern camera lenses are not likely to produce blur, and the camera CCDs sample the signals uniformly over each pixel without blurring. We have therefore chosen to use only the projection R as given in (13) to downscale without performing any pre-blurring. In the tests presented in this section, we do 2x2 VSR, thus we downscale with the factors 0.5x0.5.

To improve the validation of our simultaneous VSR method, we have also produced results with the software used by Farsiu



(a) LR (b) S-VSR (c) Bilinear (d) LR (e) 720p simultaneous VSR

Fig. 6. 720p VSR on the sequences `Boardwalk` and `Straw Hat`. The blockiness in the cracks between the boards in `Boardwalk` is removed going from SD in (a) to 720p HD in (b) and (c). Simultaneous VSR in (b) is still sharp, whereas bilinear interpolation (c) blurs out details. On `Straw Hat`, a major gain in details is obtained going from (d) SD to (e) 720p HD. The Figures are best viewed on-screen zoomed to 150% as in Fig. 16 in the online appendix [22]. Cutout sizes are `Boardwalk`: 146×50 (LR) and 182×88 (720p), and `Straw Hat`: 89×113 (LR) and 113×201 (720p).



(a) Original (b) Downsampled input

(c) 2x2 bilinear (d) 2x2 bicubic (e) 2x2 Farsiu II.E [11]

(f) 2x2 Farsiu II.D [11] (g) 2x2 Farsiu KV [10] (h) 2x2 S-VSR

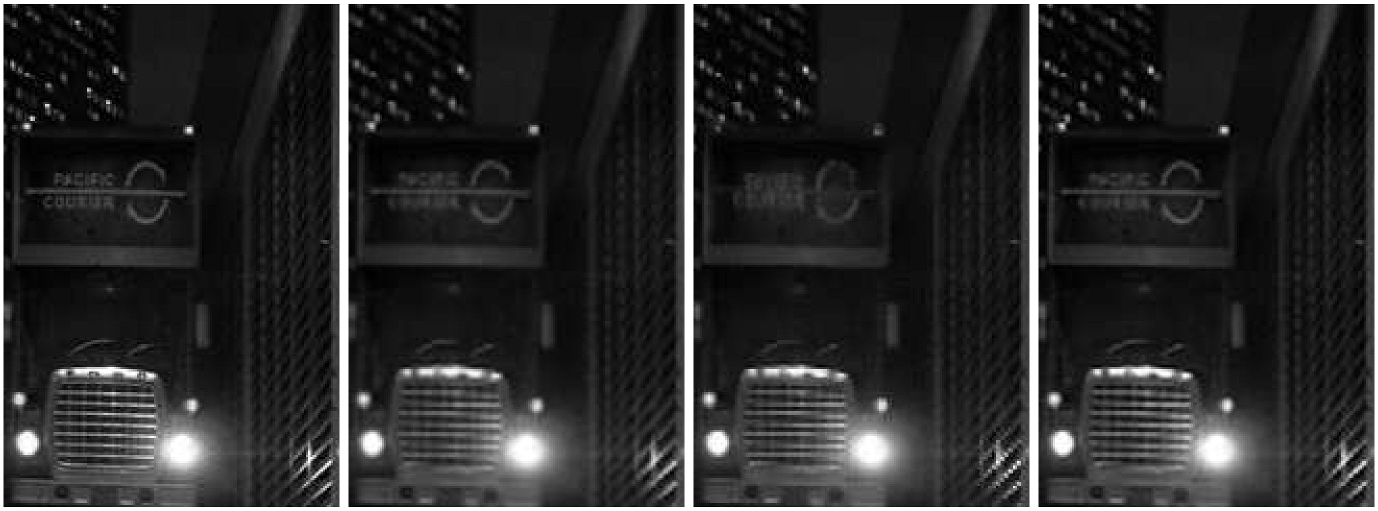
Fig. 7. Down- and upscaling of the sequence `Straw Hat`. (a) is downsampled with a factor 0.5 in height and width to give (b). (c)-(h) are 2x2 (V)SR results computed from (b). Cutout sizes: (b) 45×57 pixel, the rest 89×113 pixels. Best viewed on-screen, optimally switching between the bitmap files provided in the online material [23].

et al. in [11]. We have computed results using the main L_1-norm + bilateral total variation method (denoted Farsiu II.E referring to the section in [11] where it is given) and the faster median shift and add + bilateral TV method (Farsiu II.D). The software also implements a Kalman (black and white) video method that similar to our VSR produces an n frame HR video from an n frame LR video. It uses the same methodology as the above two methods and is described in [10]. We denote it Farsiu KV. For Farsiu II.D we have used the default SW settings (as an average over the four different parameter settings used in experiements in [11] gave bad results). With Farsiu II.E, we tried both parameter settings used in [11] and the default settings of the software, and for Farsiu KV, we used the default settings. In all three methods, we used the recommended progressive motion estimation option of the software.

As can be seen in Fig. 7, we are not able to recreate the original data in 7(a) when upscaling from 7(b). The results from doing bilinear and bicubic interpolation shown in Figures 7(c) and 7(d) respectively are very smooth, while the simultaneous VSR result in Fig. 7(h) is significantly sharper and has much more detail. There is only small and affine motion in the region of the hat itself as shown in Fig 7. The results from the three Farsiu methods are not so good, probably because of the large motions in the sequence outside the shown cutout. The results from methods II.D and II.E (settings from Fig. 12 example in [11] used) are more unsharp than the bicubic result, and the result of the KV method seems over-deblurred.

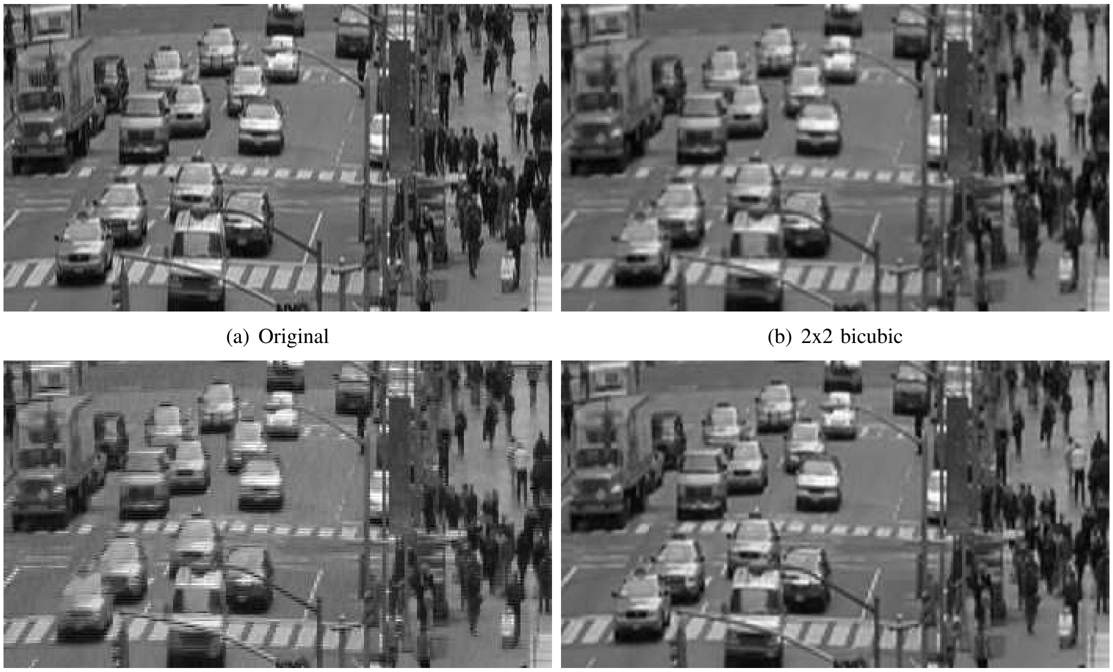
For the sequence `Straw Hat` and the two other sequences in test in this section, `Truck` and `Street`, full frame bitmap stills are given in the online material [23] (subfolder: Truck_StrawHat_Street_2x2_DownAndUp) showing the ground truth and simultaneous VSR, bicubic and bilinear results electronically. The folder also contains videos of the simultaneous VSR results. In the subfolder Farsiu_Method_Results bitmaps for II.D, KV (final deblurred and pre-deblurring) and II.E (best result from the three settings tested) are given. For unknown reasons, the software was unable to produce any outputs for the II.D and KV methods on the sequence `Truck`. Therefore only the best II.E result is given for `Truck`. For the two sequences `Truck` and `Street`, the conclusions are the same as with `Straw Hat`: Simultaneous VSR gives much sharper and more detailed results. With some settings, Farsiu results are as sharp, but at the price of added artifacts as can be seen very clearly in the online results [23]. Selected results (ground truth, bicubic, best Farsiu and simultaneous VSR) are shown in Figures 8 and 9.

When switching between the bitmaps of `Truck` [23], it shows how the brightly lit windows in the building in the background seems to light up in the result of simultaneous VSR compared to the other results. This illustrates how our S-VSR method preserves and enhances details (e.g. the front grille of the Truck) without artifacts being introduced or noise being amplified. Comparing the simultaneous VSR result on `Truck` with the ground truth in Fig. 8(a), we still lack some



(a) Original (b) 2x2 bicubic (c) 2x2 method II.E from [11] (d) 2x2 simultaneous VSR

Fig. 8. Down- and upscaling of the sequence `Truck` (size of shown cutout: 241 × 161 pixels). Best viewed on-screen, optimally switching between the bitmap files provided in the online material [23] (bilinear result is also provided).



(a) Original (b) 2x2 bicubic

(c) 2x2 method II.D from [11] (d) 2x2 simultaneous VSR

Fig. 9. Down- and upscaling of the sequence `Street` (size of shown cutout: 196 × 350 pixels). Best viewed on-screen, optimally switching between the bitmap files provided in the online material [23] (bilinear result is also provided). Zoomed versions are given in Fig. 17 of the online appendix [22].

detail in the result and there is some blockiness around high contrast edges. The blockiness is found in the results of all the tested upscaling methods, but is worst in the Farsiu II.E result. Look for instance on the wall grille to the right of the truck. Most likely, doing some Gaussian blurring prior to downsampling or using a semi-Gaussian PSF like in [36] will remove the blockiness but will also result in a loss of details.

On `Street`, the Farsiu KV result is almost as good as the II.D result shown in Fig. 9(c) but has more artifacts (blockiness) as seen on the lampposts. The pre-deblurring KV result does not have these artifacts but is then as unsharp as the bicubic result.

As we have original ground truth sequences available, we have computed the mean square error (MSE) and the peak signal to noise ratio (PSNR) for the three sequences in this test, and the results are given in Table I. The MSE is

$$MSE = \frac{1}{N}\sum_{\Omega}(u - u_{gt})^2 \tag{15}$$

where u_{gt} is the ground truth, N the number of pixels in the

TABLE I
OBJECTIVE QUALITY ASSESSMENT, MSE AND PSNR FOR THE DOWN- AND UPSCALING EXPERIMENT.

	Method	Sequence		
		`Straw Hat`	`Truck`	`Street`
MSE	Bilinear interpolation	83.03	30.56	178.6
	Bicubic interpolation	73.63	24.97	151.1
	Farsiu VSR [11]	397.8	26.36	120.6
	Simultaneous VSR	44.72	15.16	102.7
PSNR	Bilinear interpolation	28.94	33.28	25.62
	Bicubic interpolation	29.46	34.16	26.34
	Farsiu VSR [11]	22.13	33.92	27.32
	Simultaneous VSR	31.63	36.32	28.02

domain Ω of the sequence. The PSNR is

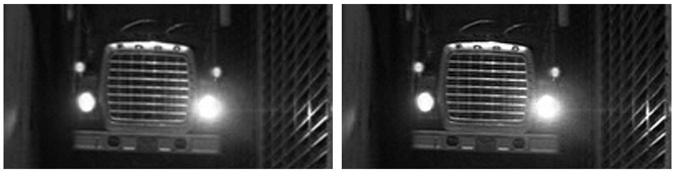
$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \tag{16}$$

where 255 is the maximum possible grey value.

As with the subjective evaluation, we can conclude from the results in Table I that bicubic interpolation performs better than bilinear interpolation, and that simultaneous VSR is by far the best of the four (highest PSNR / lowest MSE). The Farsiu MSE's are computed on the II.E result for `Truck` as it matches the middle frame 3 of the sequence used for MSE measurements. For the two other sequences, we have used the frames from the n frame KV videos produced from the n-frame inputs. The resulting MSE's are in the medium range on `Street` and `Truck` with only small motions, while it is very high on `Straw Hat`, which has large object motion.The mixed objective results for the Farsiu methods just confirm the subjective results: It is not possible to clearly evaluate the performance of the Farsiu methods on video with general motion content, unless they were to be combined with a better motion estimation algorithm.

J. Attempting to Break the Limits of Super-Resolution.

Baker and Kanade discuss the limits of super-resolution in [1] and claim that it is mainly the ability of the prior to mimic or model the image content, which decides how much one can magnify: Too large magnification factors will impose too much noise in the result. To break these limits, Baker and Kanade suggest using hallucination, a prior learned on specific image content types, e.g. faces or text. This gives them highly detailed HR images at rather large magnification factors, but they do not avoid some ringing and enhancement of unwanted details (noise) in their results. Using advanced, content specific, learned priors on our problem of upscaling general video would require a complete and nearly perfect image content detection and segmentation system, which does not exist (yet). We try instead to push more details into each pixel using the temporal filter support along the flow field. But since optical flow computation on arbitrary video is still a much harder problem than simple rigid image registration (on self-downscaled and self-transformed images), we do not get the same detail level as seen e.g. in the work of Baker and Kanade [1] at high magnification factors.



(a) 4x4 bicubic (b) 4x4 S-VSR

Fig. 11. Breaking the limits on `Truck`. Results are best evaluated on-screen either by viewing the bitmap files given in the online material [23] or the zoomed versions given in Fig. 22 of the online appendix [22]. Images are 400 × 800.

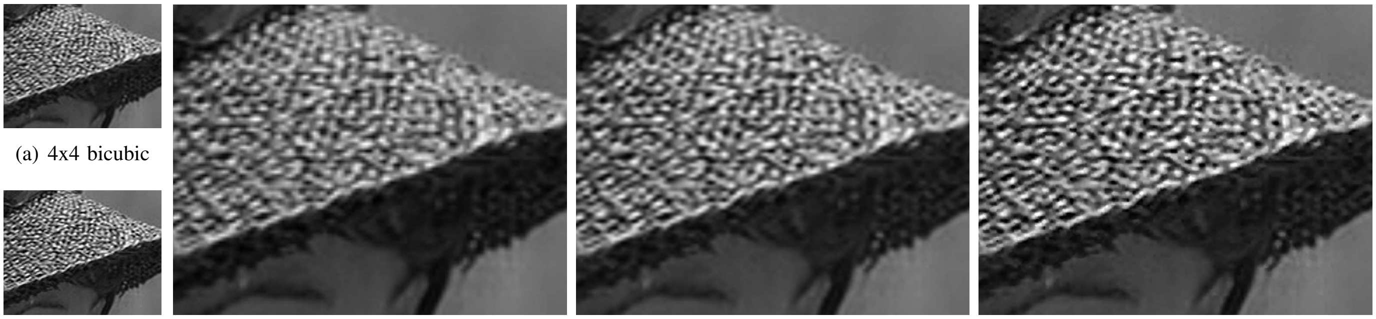
We have tested our simultaneous VSR at 4x4 and 8x8 magnifications to find the limits of our algorithm and show its modeling capabilities in case of very low information availability. To obtain the 4x4 and 8x8 magnifications, we have run our algorithm with 2x2 magnification in succession two, respectively, three times, thus doing multiresolution simultaneous VSR, which helps to optimize results at high magnification factors.

On the sequence `Straw Hat`, it is seen clearly in Fig. 10 how simultaneous VSR performs much better than bicubic interpolation at both 4x4 and 8x8 magnification – and how bad bilinear interpolation really is (goes for 4x4 as well, although it is not shown.)

As can be seen in Fig. 10(b) and more clearly in Fig. 10(e), we do get a touch of the cartoon-like look typical for total variation, but it is a small price to pay given the gain in details and sharpness over bicubic interpolation – and that without producing artifacts such as noise or ringing as in many other SR algorithms (see for instance several of the examples given in [1], [11] and [16]). The conclusions drawn from the test on `Straw Hat` are all confirmed by the test on `Truck` as shown in Figures 11 and 12. For both sequences bilinear, bicubic and S-VSR (4x4 and 8x8) results are given as bitmaps in the online material [23] in the folder Truck_StrawHat_4x4_8x8. In our opinion, the loss in naturalness when using simultaneous VSR is small, but it is a matter of individual preferences. Doing 8x8 magnification is borderline with respect to the limits of super-resolution. We do not increase noise noticeably nor do we create artifacts, but the spatial total variation prior and the temporal diffusion cannot bring out sufficient detail to give a fully natural look. When looking at the 8x8 simultaneous VSR results, the biggest problem is not lack of sharpness but lack of what could be called natural detailedness. Even with better priors (e.g. learning-based or structure tensor-based priors) and/or more reliable and accurate optical flows, we believe these details have to be added, optimally by good modeling. They cannot be pulled out of the image at high magnification factors since they are simply nonexisting in the LR input recordings.

On one hand, we do not get the over-enhancement problems that Baker and Kanade [1] and others do, but on the other hand, we do not get the same level of details either. Baker and Kanade replace the problem of noise amplification with ringing artifacts, and we get the cartoon look of total variation. In both cases, one loses the naturalness of the images/frames.

Doing SR or VSR at high magnification factors is bending



(a) 4x4 bicubic

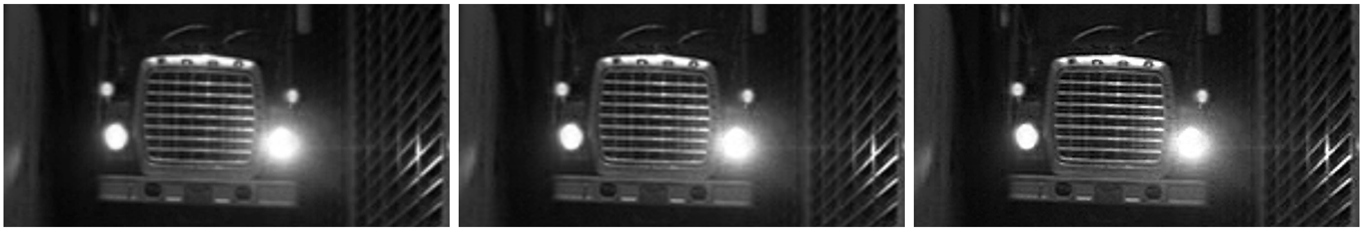
(b) 4x4 S-VSR

(c) 8x8 bilinear

(d) 8x8 bicubic

(e) 8x8 S-VSR

Fig. 10. 4x4 and 8x8 VSR on `Straw Hat`. Results are best evaluated on-screen either by viewing the bitmap files given in the online material [23] (bilinear 4x4 result is also provided) or the zoomed versions given in Figures 18 to 21 of the online appendix [22]. Image sizes are 356×453 pixel (4x4) and 711×905 pixels (8x8).



(a) 8x8 bilinear

(b) 8x8 bicubic

(c) 8x8 S-VSR

Fig. 12. Breaking the limits on `Truck`. Results are best evaluated on-screen either by viewing the bitmap files given in the online material [23] or the zoomed versions given in Figures 22 and 23 of the online appendix [22]. Images are 800×1600 pixels.

the (sub)sampling theorem too far. Thus, breaking the limits of super-resolution is not straightforward, not even with Baker and Kanade's hallucinations [1] and similar methods. Natural looking results at above 4x4 magnifications are (still) out of reach. It is also questionable whether such large magnifications will be useful for anything but saving bits in image and video coding, which is of course very useful in itself.

V. Future Work: Improved Video Super-Resolution

Implementing our simultaneous VSR in realtime hardware (e.g using field-programmable gate arrays, FPGAs, or graphics processing units, GPUs) should be realizable at a reasonable price as our algorithm performs the exact same operations on all pixels, making it highly parallelizable.

There is still plenty of room for improving the output quality in possible later versions of simultaneous VSR. Firstly, the distribution (total variation) and filters we use in our model could be improved. Learned priors as those of Baker and Kanade [1] could be one possibility. But as we know that the eye is able to do SR, improved modelling could also come from learning what filters are used in the human visual system. Lower level vision is already to some degree modelled by Gaussians and Gaussian derivative filter kernels. HVS inspired priors might be rather complex and computationally heavy, so we consider structure tensor based VSR to be the most likely next model used in VSR. It could also be interesting to see a combination of a variational flow algorithm (LR flows) with the method from [11].

A possible but unknown improvement of our algorithm would be to add the gradient constancy assumption to the intensity energy, as it might help temporal information transport.

A much higher gain in quality (more details) is expected to come from improving the accuracy of the optical flows computed. How much the flows can be improved is an open question as most development of optical flow is targeted on either minimizing the angular error on computer-generated sequences, or solving highly specialized and limited problems, e.g. satellite recordings of cloud system movements or spatiotemporal medical imaging data for a given study (e.g. heart gating).

The 3D local spatiotemporal prior on the flow, E_3 in (9), is reported to give better flow results than a purely spatial 2D flow prior on sequences with slow temporal changes (e.g. `Yosemite`) [5], [7]. The 3D flow prior is likely to cause problems in case of accelerated motion (changes in motion direction) [25]. A possible solution would be to make the prior 2D+1D instead of the unnatural 3D, so that the flow becomes linked naturally: The local temporal neighborhood is along the flow field, not at the same spatial position.

VI. Conclusion

The variational video super-resolution algorithm presented in this paper simultaneously computes high-resolution image sequences and the corresponding high-resolution flow. The algorithm is in terms of output quality clearly better than bicubic and bilinear interpolation (the latter widely used in video processing systems of today) and also outperforms professional film post production and editing systems. It also outperforms the VSR method from [11], but since its motion estimation is limited to affine (global) motion, the comparison is hard to conclude from. But it can be concluded that the method from [11] is not applicable to general motion content. There are super-resolution methods found in literature that are likely to perform better than simultaneous VSR, but only on limited cases, say, faces only. Our method is applicable to general video with arbitrary (natural) content and motion. We also show that our simultaneous VSR algorithm does not increase noise nor produce ringing artifacts at high magnification factors like some other VSR/SR algorithms do, although some (less objectionable) total variation cartoon-effect is seen at 8x8 magnification. Real time applications of variational methods do exist [6], and we therefore hope to see realtime simultaneous VSR implemented in video processing systems in the near future.

References

[1] S. Baker and T. Kanade, "Limits on Super-Resolution and How to Break Them." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1167–1183, 2002.

[2] D. F. Barbe, "Charge-coupled devices," in *Topics in Applied Physics*, D. F. Barbe, Ed. Springer, 1980.

[3] E. Bellers and G. de Haan, *De-interlacing. A Key Technology for Scan Rate Conversion*. Elsevier Sciences Publishers, Amsterdam, 2000.

[4] S. Borman and R. Stevenson, "Spatial Resolution Enhancement of Low-Resolution Image Sequences: A Comprehensive Review with Directions for Future Research," Laboratory for Image and Sequence Analysis (LISA), University of Notre Dame, Tech. Rep., Jul. 1998.

[5] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High Accuracy Optical Flow Estimation Based on a Theory for Warping," in *Proceedings of the 8th European Conference on Computer Vision*, T. Pajdla and J. Matas, Eds., vol. 4. Prague, Czech Republic: Springer–Verlag, 2004, pp. 25–36.

[6] A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnörr, "Variational Optic Flow Computation in Real-Time," *IEEE Trans. on Image Processing*, vol. 14, no. 5, pp. 608–615, 2005.

[7] A. Bruhn, J. Weickert, and C. Schnörr, "Lucas/Kanade Meets Horn/Schunck: Combining Local and Global Optic Flow Methods," *International Journal of Computer Vision*, vol. 61, no. 3, pp. 211–231, 2005.

[8] S. Chaudhuri, Ed., *Super-Resolution Imaging*, ser. The International Series in Engineering and Computer Science. Springer, 2001.

[9] S. Farsiu, M. Elad, and P. Milanfar, "Multiframe demosaicing and super-resolution of color images," *IEEE Transactions on Image Processing*, vol. 15, no. 1, pp. 141–159, 2006.

[10] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Dynamic demosaicing and color super-resolution of video sequences," in *Proceedings of SPIE Conference on Image Reconstruction from Incomplete Data III*, vol. 5562, 2004.

[11] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and Robust Multiframe Super Resolution," *IEEE Trans. on Image Processing*, vol. 13, no. 10, pp. 1327–1344, 2004.

[12] L. Florack, W. Niessen, and M. Nielsen, "The intrinsic structure of optic flow incorporating measurement duality," *The International Journal of Computer Vision*, vol. 27, no. 3, pp. 263–286, 1998.

[13] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, "Learning low-level vision," *International Journal of Computer Vision*, vol. 40, no. 1, pp. 25–47, 2000.

[14] G. Golub and C. van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: The John Hopkins University Press, 1996.

[15] R. C. Hardie, K. J. Barnard, and E. E. Armstrong, "Joint map registration and high-resolution image estimation using asequence of undersampled images," *Image Processing, IEEE Transactions on*, vol. 6, no. 12, pp. 1621–1633, Dec. 1997.

[16] H. He and P. Kondi, "An image super-resolution algorithm for different error levels per frame," *IEEE Transactions on Image Processing*, vol. 15, no. 3, pp. 592–603, 2006.

[17] M. Irani and S. Peleg, "Motion Analysis for Image Enhancement: Resolution, Occlusion, and Transparency," *Journal on Visual Communications and Image Representation*, vol. 4, no. 4, pp. 324–335, 1993.

[18] ITU, "ITU-R recommendation BT.500-11: Methodology for the subjective assessment of the quality of television pictures," Geneve, Switzerland, 6 2002.

[19] F. Kanters, "Towards object-based image editing," Ph.D. dissertation, Eindhoven Technical University, 2006.

[20] S. H. Keller, F. Lauze, and M. Nielsen, "Motion compensated video super resolution," in *Scale Space and Variational Methods in Computer Vision, SSVM 2007 Proceedings*, ser. LNCS, F. Sgallari, A. Murli, and N. Paragios, Eds., vol. 4485. Berlin: Springer, 2007, pp. 801–812.

[21] S. H. Keller, "Video Upscaling Using Variational Methods," Ph.D. dissertation, Faculty of Science, University of Copenhagen, 2007, accessed 16 Nov. 2009. [Online]. Available: http://image.diku.dk/sunebio/Afh/SuneKeller.pdf

[22] ——. (2010) Appendix of this paper containing zooms of selected results figures. [Online]. Available: http://image.diku.dk/sunebio/VSR/VSRappendix.pdf

[23] ——. (2010) Selected electronic VSR results and source code. [Online]. Available: http://image.diku.dk/sunebio/VSR/VSR.zip

[24] S. H. Keller, F. Lauze, and M. Nielsen, "Deinterlacing using variational methods," *IEEE Transactions on Image Processing*, vol. 17, no. 11, pp. 2015–2028, 2008.

[25] ——, "Temporal super resolution using variational methods," in *High-Quality Visual Experience: Creation, Processing and Interactivity of High-Resolution and High-Dimensional Video Signals*, M. Mrak, M. Grgic, and M. Kunt, Eds. Springer, 2010.

[26] S. Kim, N. Bose, and H. Valenzuela, "Recursive reconstruction of high resolution image from noisy undersampled multiframes," *IEEE Transactions on Acoustics, Speech, Signal Processing*, vol. 38, no. 6, pp. 1013–1027, 1990.

[27] F. Lauze, "Computational methods for motion recovery, motion compensated inpainting and applications," Ph.D. dissertation, IT University of Copenhagen, 2004.

[28] F. Lauze and M. Nielsen, "A Variational Algorithm for Motion Compensated Inpainting," in *British Machine Vision Conference*, S. B. A. Hoppe and T. Ellis, Eds., vol. 2. BMVA, 2004, pp. 777–787.

[29] Z. Lin and H.-Y. Shum, "Fundamental Limits of Reconstruction-Based Superresolution Algorithms under Local Translation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, pp. 83–97, 2004.

[30] G. Lu, *Communication and Computing for Distributed Multimedia Systems*. Boston, MA: Artech House, 1996.

[31] M. W. Matlin and H. J. Foley, *Sensation and Perception*, 4th ed. Allyn and Bacon, 1997.

[32] D. Mumford, "Bayesian rationale for the variational formulation," in *Geometry-Driven Diffusion In Computer Vision*, B. M. ter Haar Romeny, Ed. Kluwer Academic Publishers, 1994, pp. 135–146.

[33] M. Nadenau, S. Winkler, D. Alleysson, and M. Kunt. (2000) Human vision models for perceptually optimized image processing – a review. [Online]. Available: http://citeseer.ist.psu.edu/nadenau00human.html

[34] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert, "Highly Accurate Optic Flow Computation With Theoretically Justified Warping," *International Journal of Computer Vision*, vol. 67, no. 2, pp. 141–158, April 2006.

[35] A. J. Patti, M. I. Sezan, and A. M. Tekalp, "Super resolution video reconstruction with arbitrary sampling lattices and non-zero aperture time," *IEEE Transactions on Image Processing*, vol. 6, no. 8, p. 1064 1076, 1997.

[36] A. Roussos and P. Maragos, "Vector-valued image interpolation by an anisotropic diffusion-projection PDE," in *Scale Space and Variational Methods in Computer Vision, SSVM 2007 Proceedings*, ser. LNCS, F. Sgallari, A. Murli, and N. Paragios, Eds., vol. 4485. Berlin: Springer, 2007, pp. 104–115.

[37] M. Rucci, R. Iovin, M. Poletti, and F. Santini, "Miniature eye movements enhance fine spatial detail," *Nature*, vol. 447, no. 7146, pp. 851–854, June 2007. [Online]. Available: http://www.nature.com/nature/journal/v447/n7146/pdf/nature05866.pdf

[38] R. Schultz and R. Stevenson, "Extraction of High Resolution Frames from Video Sequences," *IEEE Trans. on Image Processing*, vol. 5, no. 6, pp. 996–1011, 1996.

[39] E. Shechtman, Y. Caspi, and M. Irani, "Space-Time Super-Resolution," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 531–545, 2005.

[40] H. Shen, L. Zhang, B. Huang, and P. Li, "A map approach for joint motion estimation, segmentation, and super resolution," *Image Processing, IEEE Transactions on*, vol. 16, no. 2, pp. 479–490, Feb. 2007.

[41] R. Tsai and T. Huang, "Multiframe Image Restoration and Registration," in *Advances in Computer Vision and Image Processing*, vol. 1, 1984, pp. 317–319.

[42] D. Tschumperlé and B. Besserer, "High quality deinterlacing using inpainting and shutter-model directed temporal interpolation," in *Proc. of ICCVG Intl. Conf. on Computer Vision and Graphics*. Kluwer, 2004, pp. 301–307.

[43] D. Tschumperlé and R. Deriche, "Vector-valued image regularization with PDEs: A common framework for different applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 506 – 517, 2005.