Journal of Mathematical Imaging and Vision (2021) 63:144–156
https://doi.org/10.1007/s10851-020-00973-6

Towards PDE-Based Video Compression with Optimal Masks Prolongated by Optic Flow

Michael Breuß[1] · Laurent Hoeltgen[1] · Georg Radow[1]

Received: 23 October 2019 / Accepted: 8 June 2020 / Published online: 13 July 2020

Abstract
Lossy image compression methods based on partial differential equations have received much attention in recent years. They may yield high-quality results but rely on the computationally expensive task of finding an optimal selection of data. For the possible extension to video compression, this data selection is a crucial issue. In this context, one could either analyse the video sequence as a whole or perform a frame-by-frame optimisation strategy. Both approaches are prohibitive in terms of memory and run time. In this work, we propose to restrict the expensive computation of optimal data to a single frame and to approximate the optimal reconstruction data for the remaining frames by prolongating it by means of an optic flow field. In this way, we achieve a notable decrease in the computational complexity. As a proof-of-concept, we evaluate the proposed approach for multiple sequences with different characteristics. In doing this, we discuss in detail the influence of possible computational setups. We show that the approach preserves a reasonable quality in the reconstruction and is very robust against errors in the flow field.

Keywords Partial differential equations · Inpainting · Laplace interpolation · Optic flow · Video reconstruction

1 Introduction

Transform-based image and video compression algorithms are still the preferred choice in many applications [33]. However, in recent years there has been a growing interest in alternative approaches [1,11,18,30]. It has been shown that partial differential equation (PDE)-based methods represent a viable alternative in the context of image compression. In order to achieve a competitive level with state-of-the-art codecs, the PDE-based methods require sophisticated data optimisation schemes and fast numerical algorithms. The most important task is the choice of a small subset of pixels, often called mask, from which the original image can be accurately reconstructed by solving a PDE.

This data selection problem has proven to be delicate, see [6,8,12,13,22,39] for some strategies considered in the past. Most approaches are either very fast but yield suboptimal results or they are relatively slow and return very appropriate data for the reconstruction. A thorough optimisation of a whole image sequence yielding high reconstruction quality is therefore computationally rather demanding. Most approaches have resorted to a frame-by-frame consideration. Yet, even such a frame-wise tuning can be computationally expensive, especially for longer videos.

In this paper, we discuss a simple and fast approach to skip the costly data selection step in a certain number of frames. Instead, we perform a computationally much less demanding data transport along the temporal axis of the video sequence. In order to evaluate important properties occurring by the realisation of this approach, we focus on the interplay between reconstruction quality and the accuracy of the transporting vector field. The actual data compression rate that can be achieved is the subject of future research.

To give some more details of our approach, we consider an image sequence and compute a highly optimised pixel mask used for a PDE-based reconstruction within the first, single frame. Next, we compute the displacement between the individual subsequent frames by means of an optic flow

✉ Georg Radow
radow@b-tu.de

Michael Breuß
breuss@b-tu.de

Laurent Hoeltgen
hoeltgen@b-tu.de

[1] Chair for Applied Mathematics, BTU Cottbus-Senftenberg, Cottbus, Germany

method. We shift the carefully selected pixels from the pixel mask of the first frame according to the optic flow field. The shifted data are then used for the reconstruction process, in this case PDE-based inpainting. The effects of erroneous or suboptimal shifts of mask pixels on the resulting video reconstruction quality can then be evaluated.

The framework for video compression presented in [1] has some technical similarities to our approach. The conceptual difference is that in their work a reconstructed image is shifted via optic flow fields from the first to subsequent frames. In contrast, we use optic flow fields only for the propagation of mask locations and deal with an inpainting problem in each frame.

The current paper is based on our conference paper [19]. In comparison with that work, we present here some novelties and a much broader numerical study. The most apparent novelty is that we propose here a variation of the original approach which circumvents the accumulation of rounding errors. With this new algorithm, we are able to significantly decrease reconstruction errors at the negligible computational expense of a bilinear interpolation. We augment the numerical evaluation of our approach, e.g. by considering several optic flow algorithms. Furthermore, we have added a numerical experiment on the development of the mask pixel density during the video sequence, illuminating a basic property of the approach that could be explored in future work.

Our paper is structured as follows. First we briefly recall the considered models and methods. Next we describe how they are concatenated in our strategy. Finally, all components are carefully evaluated, where we focus here on quality in terms of reconstruction error. Let us note again that we will not consider the impact on the file compression efficiency, as a detailed analysis of the complete, resulting data compression pipeline would be beyond the scope of this work.

2 Discussion of Considered Models and Methods

The recovery of images, as in a video sequence, by means of interpolation is often called inpainting. Since the main issue in our approach is concerned with the selection of data for a corresponding PDE-based inpainting task, it will be useful to elaborate on the inpainting problem in some detail. After discussing possible extensions from image to video inpainting, we consider the optic flow methods and some algorithmic aspects of them as employed in this work.

2.1 Image Inpainting with PDEs

The inpainting problem goes back to the works of Masnou and Morel as well as Bertalmío and colleagues [3,25], although similar problems have been considered in other fields already before. There exist many inpainting techniques, often based on interpolation algorithms, but PDE-based approaches are among the most successful ones, see e.g. [14,15,31]. For the latter, strategies based on the Laplacian are often advocated [5,23,28,32]. Mathematically, the simplest model is given by the elliptic mixed boundary value problem

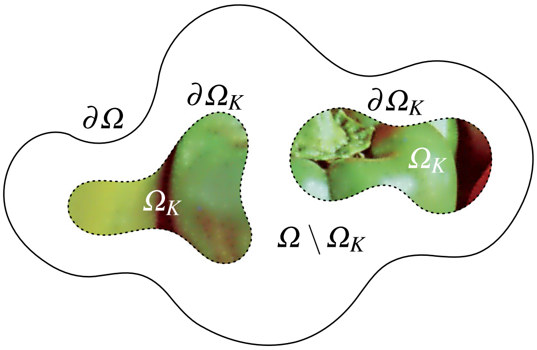


Fig. 1 Inpainting model as given in (1) with known image data f in Ω_K (see [37] for the source image). The task consists in recovering a reasonable reconstruction of the image f in $\Omega \setminus \Omega_K$ by solving the PDE in (1)

$$\begin{cases} -\Delta u(x) = 0, & \text{in } \Omega \setminus \Omega_K, \\ u(x) = f(x), & \text{in } \partial\Omega_K, \\ \partial_n u(x) = 0, & \text{in } \partial\Omega \setminus \partial\Omega_K, \end{cases} \tag{1}$$

see the sketch in Fig. 1 and the related discussion in [20]. Here, f represents known image data in a region $\Omega_K \subset \Omega$ (resp. on the boundary $\partial\Omega_K$) of the whole image domain Ω. Further, $\partial_n u$ denotes the derivative in direction of the outer normal. In an image compression context, the image f is known on the whole domain Ω, and one would like to identify the smallest set Ω_K that yields a good reconstruction u when solving (1).

While solving (1) numerically is a rather straightforward task, finding an optimal subset Ω_K is much more challenging. Mainberger et al. [24] consider a combinatorial strategy while Belhachmi and colleagues [2] approach the topic from the analytic side. Recently [17], the "hard" boundary conditions in (1) have been replaced by softer weighting schemes, cf. again [20]. If we denote the weighting function by $c : \Omega \to \mathbb{R}$, then (1) becomes:

$$\begin{cases} (1 - c(x))(-\Delta)u(x) & \\ +c(x)(u(x) - f(x)) = 0, & \text{in } \Omega, \\ \partial_n u(x) = 0, & \text{in } \partial\Omega \setminus \partial\Omega_K. \end{cases} \tag{2}$$

In the case where c is the indicator function of Ω_K, (2) coincides with the PDE in (1). Whenever $c(x) = 1$, we require $u(x) - f(x) = 0$ and $c(x) = 0$ implies $-\Delta u(x) = 0$.

Optimising a weighting function c which maps to $\mathbb{R}$ is notably simpler than solving a combinatorial optimisation problem when the mask c maps to $\{0, 1\}$. As the optimal set Ω_K is given by the support of the function c, the benefit of the formulation (2) is that one may adopt ideas from sparse signal processing to find such a good mask. To this end, Hoeltgen et al. [17] have proposed the following optimal control formulation:

$$\arg\min_{u,c} \left\{ \int_\Omega \frac{1}{2} \left(u\left(x\right) - f\left(x\right)\right)^2 + \lambda |c\left(x\right)| + \frac{\varepsilon}{2} c\left(x\right)^2 \, \mathrm{d}x , \right\}$$
$$\begin{cases} \left(1 - c\left(x\right)\right)\left(-\Delta\right) u\left(x\right) & \\ +c\left(x\right)\left(u\left(x\right) - f\left(x\right)\right) = 0, & \text{in } \Omega, \\ \partial_n u(x) = 0, & \text{in } \partial\Omega \setminus \partial\Omega_K . \end{cases} \tag{3}$$

Equation (3) can be solved by an iterative linearisation of the PDE in terms of (u, c), followed by a primal-dual optimisation strategy such as [9] for the occurring convex problem with linear constraints. As reported in [17], a few hundred linearisations need to be performed to obtain a good solution. This also implies that an equal amount of convex optimisation problems needs to be solved. Even if highly efficient solvers are used for the latter, the run time will still be considerable. An alternative approach for solving (3) was also presented in [26].

Besides optimising Ω_K (resp. c), it is also possible to optimise the Dirichlet boundary data in such a way that the global error is minimal. If $M(c)$ denotes the linear solution operator with mask c that yields the solution of (2), then we can write this tonal optimisation as

$$\arg\min_g \left\{ \| M(c) g - f \|_2^2 \right\} . \tag{4}$$

This idea has originally been presented in [24]. In [16], it is shown that there exists a dependence between non-binary optimal c (i.e. mapping to $\mathbb{R}$ instead of $\{0, 1\}$) and optimal tonal values g. More specifically, the results obtained with binary masks and tonal optimisation are equivalent to those obtained with non-binary masks and no tonal optimisation. Efficient algorithms for solving (4) can be found in [16,24]. These algorithms are faster than solving (3), yet their run times still range from a few seconds to a minute on standard desktop computers, e.g. the system detailed in Sect. 4.1.

2.2 Extension from Images to Videos

The strategies for inpainting discussed so far have been applied to grey-value or colour images almost exclusively. However, straightforward extensions to video sequences would be possible, in principle. The simplest strategy would be to consider a frame-by-frame approach. Alternatively, in (3) one could also extend the Laplacian into the temporal direction to compute an optimal mask in space-time. This would reduce the temporal redundancy (assuming that the content of subsequent frames does not change much) in the mask c compared to a frame-wise approach. Unfortunately, the latter strategy is prohibitively expensive. A one-second-long video sequence in 4K resolution (3860 × 2160 pixels) with a framerate of 60 Hz would require analysing approximately 500 million pixels. A frame-by-frame optimisation would be more memory efficient, since the whole sequence does not need to be loaded at once, but it would still require solving 60 expensive optimisation problems.

In this context, let us note again that our approach modifies the frame-wise proceeding by computing a displacement field and shifting optimised mask locations from one frame to the next. We refer to [34] for a general overview on the concepts and ideas employed in modern video compression codecs such as MPEG.

2.3 Optic Flow

For the sake of simplicity, we opt for two classic variational models that illustrate a certain variation in quality and flow field properties. We opt for the well-understood model of Horn and Schunck proposed originally in [21] and the TV-L_1 model for which we refer to [40] for a detailed description.

Given an image sequence $f(x, y, t)$, where x and y are the spatial dimensions and t the temporal dimension, the considered optic flow methods compute a displacement field $(u(x, y), v(x, y))$ that maps the frame at time t onto the frame at time $t + 1$. In the Horn–Schunck (HS) model, this is done by minimising the energy functional

$$\int_\Omega \left(f_x u + f_y v + f_t\right)^2 + \alpha \left\| \begin{pmatrix} \nabla u \\ \nabla v \end{pmatrix} \right\|_2^2 \mathrm{d}x\mathrm{d}y \tag{5}$$

where f_x, f_y and f_t denote the partial derivatives of f with respect to x, y and t and where $\Omega \subset \mathbb{R}^2$ denotes the image domain. The HS model is very popular, and highly efficient numerical schemes exist that are capable of solving (5) in real-time (30 frames per second), see [7]. Obviously, replacing already a single computation of c with the computation of a displacement field (u, v) will save a significant amount of time. If the movements in the image sequence are small and smooth enough, it appears to be very likely, that several masks c can be replaced by making use of such a flow field, thus saving even more run time.

As indicated, in addition to the HS model we also consider the TV-L_1 model. Loosely speaking, this model can be derived from (5) by changing the L_2 norm in the data fidelity term to the L_1 norm and changing the L_2 norm regulariser to a total variation (TV) seminorm. For the TV seminorm, one can choose from multiple possible realisations, from which we consider two options as follows. In [40], a method was

Algorithm 1: Data Selection with Rounding

```
Data: Image sequence $f$
Result: Optimised data for the reconstruction
Optimise mask $c$ for first frame of the sequence $f$ by solving (3)
Binarise $c$ by setting all non-zero entries to 1
Find optic flow between all frames of the sequence by minimising (5)
Round flow field entries to nearest integer
Transport $c$ according to the flow field
Perform tonal optimisation by solving (4) on the mask locations in each frame
```

Algorithm 2: Data Selection with Interpolation

```
Data: Image sequence $f$
Result: Optimised data for the reconstruction
Optimise mask $c$ for first frame of the sequence $f$ by solving (3)
Binarise $c$ by setting all non-zero entries to 1
Find optic flow between all frames of the sequence by minimising (5)
Transport $c$ according to the flow field interpolated at non-zero locations of $c$
Round mask locations to nearest integer
Perform tonal optimisation by solving (4) on the mask locations in each frame
```

proposed to minimise an approximation of the energy

$$\int_\Omega |f_x u + f_y v + f_t| + \alpha \left(\|\nabla u\|_2 + \|\nabla v\|_2\right) \,\mathrm{d}x\mathrm{d}y. \tag{6}$$

Here the regularisation of u and v is decoupled. This is not the case for the energy

$$\int_\Omega |f_x u + f_y v + f_t| + \alpha \left\| \begin{pmatrix} \nabla u \\ \nabla v \end{pmatrix} \right\|_2 \mathrm{d}x\mathrm{d}y, \tag{7}$$

which was recently investigated in detail in [29].

3 Combining Optimal Masks with Flow Data

Given an image sequence f, we compute a sparse inpainting mask for the first frame with the method from [17]. According to the results in [16], we threshold the mask c and set all non-zero values to 1. Next, we compute the displacement field between all subsequent frames in the sequence by solving (5) for each pair of consecutive frames. For prolongating the mask locations, we now consider two approaches.

The first approach is identical to the one presented in [19]. The obtained flow fields (u, v) are rounded point-wise to their nearest integers to assert that they point exactly onto a grid point. Then, the mask points from the first frame are simply moved according to the displacement field.

If the displacement points outside of the image or if it points onto a position where a mask point is already located, then we drop the current mask point. Since we are considering sparse sets of mask points, the probability of two mask points being shifted to the same location is rather low such that hardly any data get lost because of such an event. For displacements pointing outside of the image, we refer to an experimental study presented in Sect. 4.4.

Once the mask has been set for each frame, we perform a tonal optimisation of the data as discussed in [16]. The reconstruction can then simply be done by solving (2) for each frame. The complete procedure is also detailed in Algorithm 1.

Instead of rounding the flow field vectors, one could also follow the idea to perform a forward warping [27] and spread a single mask point on all neighbouring mask points. With this strategy, flow fields that point to the same location would simply add up the mask values. Even though this appears as a mathematically clean approach since the sum of mask values is preserved, our experiments showed that the smearing of the mask values caused strong blurring effects in the reconstructions and lead to overall worse results. Therefore, we do not elaborate on this modification in detail.

In the second approach, which we propose as a novelty in this paper, the flow fields are not rounded towards the nearest integers. Instead the mask locations are shifted according to the exact displacement fields. The new mask locations will typically not be on a grid point; therefore, the values of the surrounding optic flow field defined at the grid points are interpolated for shifting the mask locations to the next frame. The mask locations are only rounded to the nearest grid point for computing an inpainting mask in the current frame.

Again, if the displacement points outside of the image the corresponding mask point is dropped. However, if two mask points have the same rounded position, the exact position will usually still differ. Therefore, in this case a mask point is dropped only for the computation of the inpainting mask in the current frame. Finally, based on the rounded mask locations, the tonal optimisation [16] is performed. This second approach is detailed in Algorithm 2.

The data that need to be stored for the reconstruction consists of the mask point positions in the first frame, the flow fields that move the mask points along the image sequence (resp. the mask positions in the subsequent frames), and the corresponding tonal optimised pixel values. We emphasise that it is not necessary to store the whole displacement field but only at the locations of a mask point in each frame. Thus, the memory requirements for the storage remain the same as when optimising the mask in each frame. Yet, the whole approach is considerably faster compared to a frame-wise mask optimisation. We also remark that the considered strategy is rather generic. One may exchange the mask selection

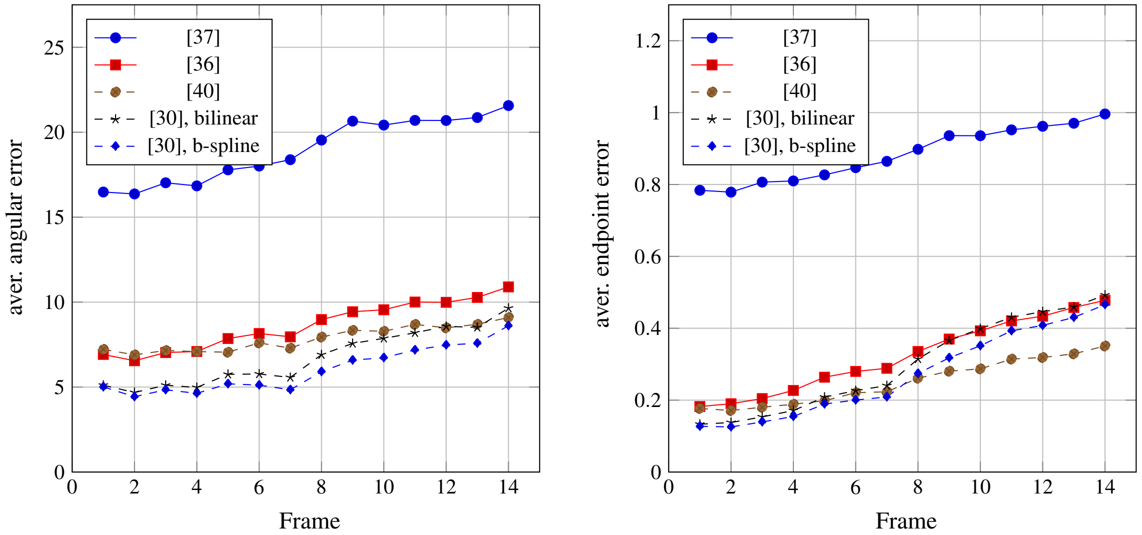


Fig. 2 Angular errors (in degree) and endpoint errors (in pixel width) in the optic flow field of the Yosemite sequence in-between frame i and $i+1$ for the considered methods. The regularisation weight was optimised for each pair of frames to minimise the angular error. The methods with coarse-to-fine strategies are at least twice as accurate as [36]. The methods [29,40] based on TV-L_1 models exhibit lower errors than the HS implementation [35] in most cases

algorithm and the optic flow computation with any other method that yields similar data.

4 Experimental Evaluation

To evaluate the proposed approach, we give further details on our experimental setup, including a rough comparison of run times for the different stages of Algorithms 1 and 2.

We discuss the influence of the quality of the flow fields by means of an example. Then we proceed by evaluating the proposed methods for a number of image sequences.

4.1 Details on the Considered Methods

As already mentioned, we compute the inpainting masks with the algorithm from [17] and use the LSQR-based algorithm from [16] for tonal optimisation. In terms of quality, these methods are among the best performing ones for Laplace reconstruction. However, alternative solvers such as presented in [10,24] may be used as well.

For a reasonable comparison of optic flow methods, we have resorted to the builtin MATLAB implementation [36] of the HS method and a more sophisticated implementation available from [35]. Additionally, we test multiple implementations of more modern TV-L_1 models, namely the implementations presented in [40] and [29]. Let us note again that in doing this we extend our previous conference paper.

All but the builtin MATLAB implementation include a coarse-to-fine warping strategy [27]. For the implementation from [29], we test this strategy here in combination with both bilinear and b-spline interpolation of order 4. Evaluations on the Yosemite sequence have shown that the implementations including coarse-to-fine and warping frameworks are usually twice as accurate (see Fig. 2) as the builtin MATLAB function, but in case of the TV-L_1 models they also exhibit larger run times. However, the computation of an accurate displacement field is still significantly faster than a thorough optimisation of the mask point locations.

All methods have been implemented in MATLAB. On a desktop computer with an Intel Core i9-7920X CPU with 12 cores clocked at 2.90 GHz and 64GB of memory, the average run time of the MATLAB optic flow implementation (10000 iterations at most) on the $512 \times 512 \times 10$ "Toy Vehicle" sequence from [37] was 14 seconds for each flow field between two frames. For the other implementations, we always used 8 coarse-to-fine levels with 10 warping steps at most. The implementation of the HS model from [35] took 13 seconds. The average computation times for the TV-L_1 implementations were higher, as can be expected. Here the underlying optimisation problem in one warping step is solved iteratively, with 200 iterations at most. The implementation from [40] took 105 seconds and the implementation from [29] took 85 seconds with bilinear and 128 seconds with b-spline interpolation. The tonal optimisation (360 iterations at most) took on average 20 seconds per frame.

The optimal control-based mask optimisation (1500 linearisation and 3000 primal dual iterations at most) required on average 2-26 seconds per linearisation and usually all 1500 linearisations are carried out. A complete optimisation

Table 1 Evaluation of the Yosemite sequence

λ	Density	Optimal c	Ground truth	[36]	[35]	[40]	[29], bilinear	[29], b-spline	Zero	Random
Average MSE with Algorithm 1										
0.0030	15.75%	25.60	130.15	128.49	124.28	122.22	127.48	126.32	131.41	236.32
0.0063	9.21%	52.16	179.33	178.79	174.42	172.95	176.96	175.81	185.59	283.68
0.0125	5.51%	88.75	227.84	227.81	221.56	220.84	224.10	222.95	239.25	333.49
0.0250	3.11%	139.51	284.17	285.54	277.28	275.32	278.45	277.35	299.29	399.72
Average MSE with Algorithm 2										
0.0030	15.75%	25.60	83.77	111.76	83.65	78.38	85.51	80.14	131.41	212.41
0.0063	9.21%	52.16	118.16	156.70	118.90	112.86	120.36	114.35	185.59	266.31
0.0125	5.51%	88.75	157.40	201.26	157.51	151.66	159.33	152.96	239.25	323.90
0.0250	3.11%	139.51	208.33	252.22	204.47	200.04	207.17	201.27	299.29	395.86

The density specifies the percentage of non-zero mask pixels in the first frame. The errors in the fifth-to-ninth column correspond (in order) to the builtin MATLAB function [36], the HS implementation from [35], the TV-L_1 implementation from [40] and the TV-L_1 implementation from [29] using bilinear and b-spline interpolation of order 4. The second-to-last column sets the flow field in every pixel to 0. The last column shows the error when using a random flow field in the same numerical range as the ground truth. The upper part of the table represents the results of Algorithm 1, where the flow fields are rounded. Here the results with shifted masks are very similar. They outperform zero flow especially in the first frames, cf. Fig. 3. For Algorithm 2 displayed in the bottom part of the table, there is a clear performance gain when using more accurate flow fields than the ones generated with [36], which in turn are clearly outperforming zero flow, cf. Fig. 4. In all eight experiments, the TV-L_1 implementation from [40] delivers the lowest average MSE

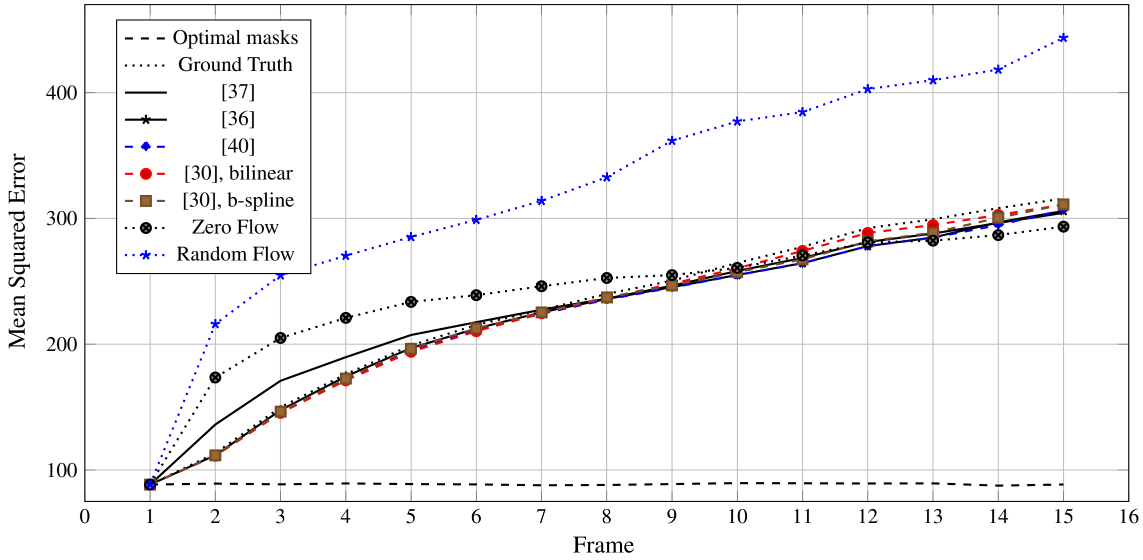


Fig. 3 Reconstruction error with Algorithm 1 for the Yosemite sequence in each frame using a mask with density 5.51% shifted by different flow fields. The average angular error over all frames of the method from [36] is 18.95 and 17.04 if measured at mask points only. For the method from [35], the corresponding errors are 8.62 and 5.30. For methods [29,40], the error values are similar to the ones with [35], cf. Fig. 2. The error in the reconstruction is hardly influenced by the quality of the optic flow. The dashed line at the bottom indicates the error in the reconstruction from an optimal mask

takes therefore about 6 hours per frame. The large variations in the run times of the single linearisations stem from the fact that the sparser the mask becomes the more ill-posed the optimisation problem becomes and the more iterations are needed to achieve the desired accuracy. All in all, the mask optimisation is at least 150 times slower than any of the optic flow computations or the tonal optimisation.

4.2 Evaluation

We evaluate the proposed Algorithm 1 on several image sequences. At first, we consider the Yosemite sequence with clouds, available from [4]. Since the ground truth of the displacement field is completely known, we can also analyse the impact of the quality of the flow on the reconstruction. Further, we evaluate the image sequences from the USC-SIPI

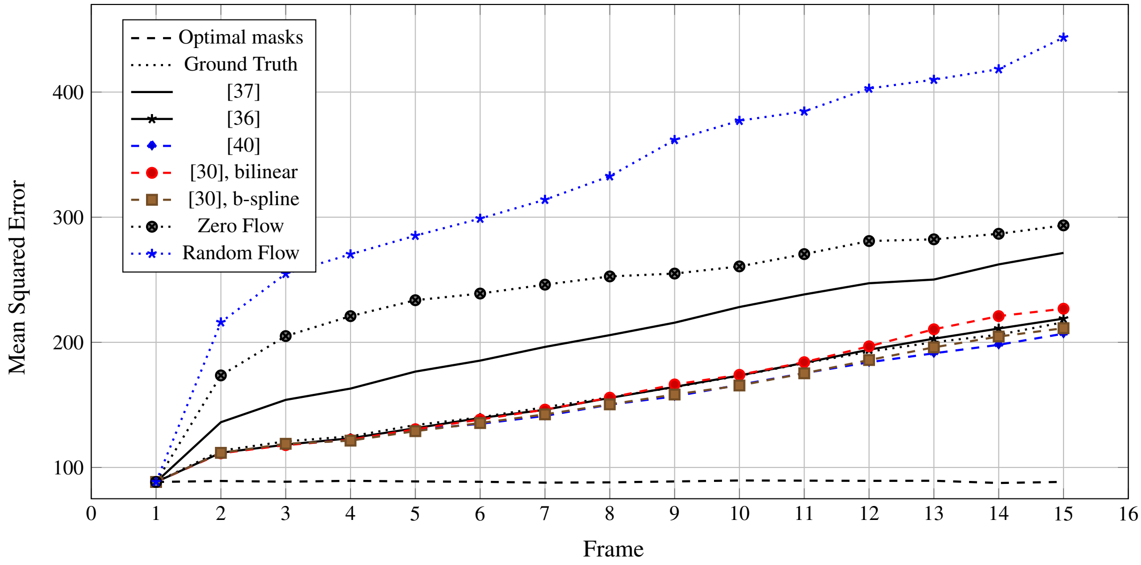


Fig. 4 Reconstruction error with Algorithm 2 for the Yosemite sequence in each frame using a mask with density 5.51% shifted by different flow fields. Methods incorporating a computed flow field are clearly outperforming the static mask (zero flow). The flow field from [36] is outperformed by more accurate methods [29,35,40], which are similar to the reconstructions with the ground truth flow. The dashed line at the bottom indicates the error in the reconstruction from an optimal mask

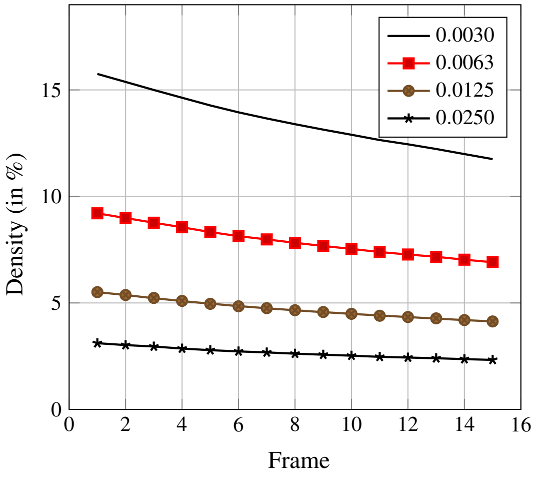


Fig. 5 Density of mask pixels for the Yosemite sequence and different parameter choices of λ in (3). The density is steadily decreasing as objects move out of the image plane

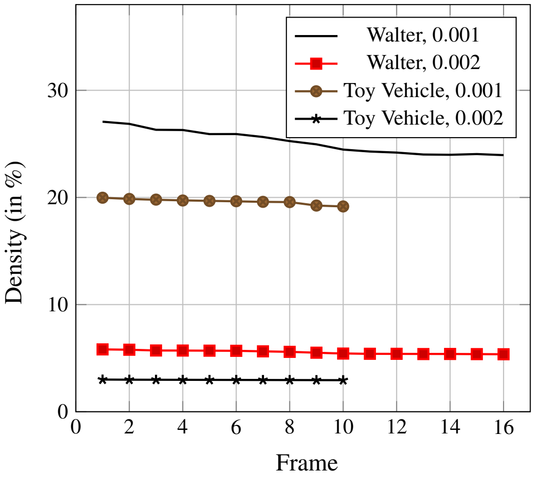


Fig. 6 Density of mask pixels for the Walter and Toy Vehicle sequences and different parameter choices of λ in (3). The density is relatively stable, since the perspective is constant in both scenes

Image Database [37]. The database contains four sequences of different length with varying image characteristics. For the latter sequences, no ground truth displacement field is known. As a such we can only report the reconstruction error in terms of mean squared error (MSE) and structural similarity index (SSIM) [38].

4.3 Influence of the Optic Flow

In Table 1, we present the evaluation of our approach on the Yosemite sequence for different choices of parameters of the mask optimisation algorithm and the corresponding reconstruction. In all these experiments, we set the stabilising trust-region parameter μ to 1.25 (see [17] for a definition of this parameter) and ε from (3) to 10^{-9} in the mask optimisation algorithm. The regularisation weight in (5), (6) or (7)

Table 2 Evaluations of the MSE and SSIM on Image Sequences from the USC-SIPI Image Database [37]

Sequence	Density	First	Last	Min.	Max	Mean		
						[40]	Optimal c	Zero
MSE								
Yosemite	5.51%	88.38	206.89	88.38	206.89	151.66	88.75	239.25
	15.75%	25.34	119.95	25.34	119.95	78.38	25.60	131.41
Walter	5.82%	7.33	29.70	7.33	29.70	18.86	7.23	82.29
	27.07%	1.88	10.94	1.88	11.21	5.88	1.81	21.20
Toy Vehicle	3.00%	4.11	19.85	4.11	36.22	25.29	4.45	30.16
	19.97%	1.26	7.83	1.26	16.90	10.91	1.27	12.79
Plant (close)	3.20%	111.99	320.56	111.99	361.73	265.30	124.01	425.80
	5.36%	73.74	261.93	73.74	300.08	216.09	79.68	355.92
	19.05%	18.50	160.15	18.50	194.85	132.06	18.87	164.23
Plant (far)	3.65%	137.99	238.06	137.99	241.70	203.82	143.10	326.08
	6.64%	87.32	191.24	87.32	193.69	156.70	92.74	268.54
	23.18%	19.71	115.22	19.71	115.36	82.91	22.06	141.85
SSIM								
Yosemite	5.51%	0.8196	0.6678	0.6678	0.8196	0.7321	0.8150	0.6433
	15.75%	0.9372	0.7881	0.7881	0.9372	0.8487	0.9333	0.7848
Walter	5.82%	0.9577	0.9111	0.9111	0.9577	0.9285	0.9581	0.8744
	27.07%	0.9810	0.9603	0.9603	0.9810	0.9686	0.9817	0.9514
Toy Vehicle	3.00%	0.9692	0.9451	0.9357	0.9692	0.9467	0.9681	0.9389
	19.97%	0.9856	0.9724	0.9637	0.9856	0.9718	0.9858	0.9693
Plant (close)	3.20%	0.7177	0.5756	0.5725	0.7177	0.6177	0.7295	0.4870
	5.36%	0.7843	0.6370	0.6287	0.7843	0.6761	0.7997	0.5512
	19.05%	0.9230	0.7791	0.7509	0.9230	0.8077	0.9336	0.7537
Plant (far)	3.65%	0.6739	0.5217	0.5217	0.6739	0.5671	0.6513	0.4584
	6.64%	0.7667	0.5895	0.5895	0.7667	0.6433	0.7436	0.5297
	23.18%	0.9300	0.7397	0.7397	0.9300	0.7967	0.9237	0.7348

An optimal mask is computed on the first frame and shifted via Algorithm 2 according to the flow field computed with the method from [40]. In (6), the parameter $\alpha = 10$ was used. The error for the Toy Vehicle sequence is not monotonically increasing due to large motions and strong occlusions in certain frames. In the last two columns, the results are displayed for frame-wise optimisation and for static masks with zero flow fields

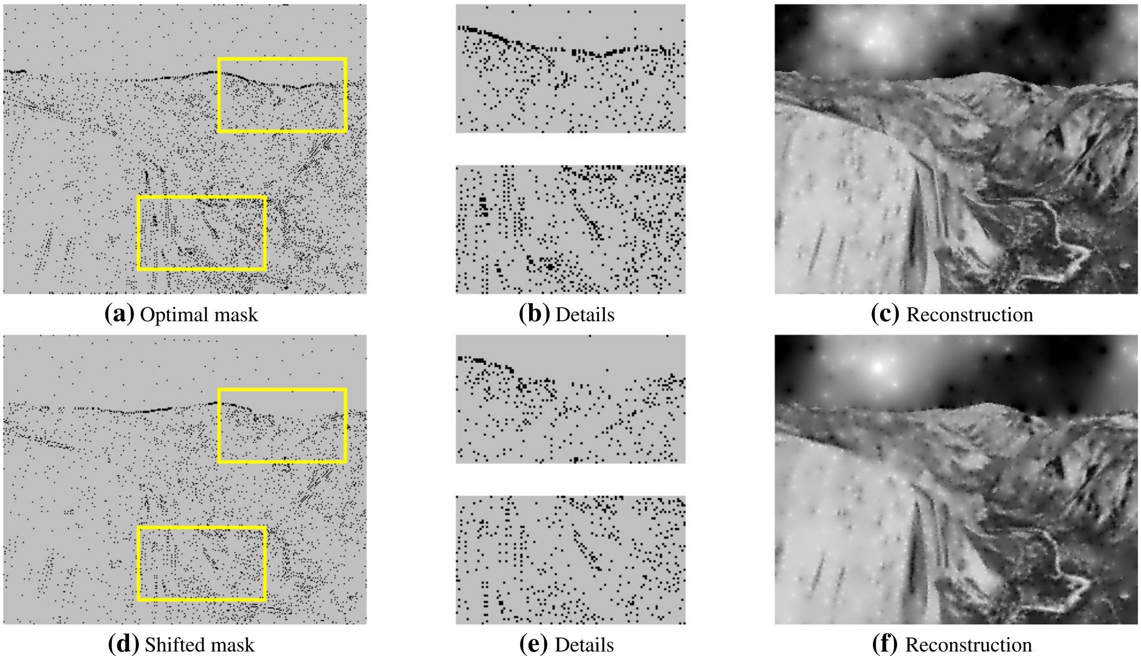
was always optimised for low angular error by means of a line search strategy.

The first column of the table lists the parameter λ which is responsible for the mask density and the second column contains the corresponding mask density in the first frame. The other columns list the average reconstruction error over all 15 frames when (i) using an optimised mask obtained from the optimal control framework explained in [17] in all the frames, (ii) the optimised mask from the first frame shifted in accordance with the ground truth displacement field, (iii) the mask from the first frame shifted in accordance with the computed displacement fields for all considered optic flow implementations, (iv) the mask from the first frame used for all subsequent frames (i.e. using a zero flow field), and (v) the mask from the first frame shifted by a random flow field within the same numerical range between each pair of frames as the ground truth.

All reconstructions in the upper half of the table have been done according to Algorithm 1. The lower half exhibits the same experiment but according to Algorithm 2, without rounding of the flow fields.

The error evolution with random flow fields serves as a worst case example. The shifted masks are not completely random, but the resulting image quality (in terms of MSE) deteriorates stronger than in all other experiments. As indicated, the random flow fields are in the same numerical range as the groundtruth flow, i.e. in $[-4.0, 2.0] \times [-0.051, 4.1]$.

As expected, a higher mask density yields a smaller error in the reconstruction in all cases. When shifting the masks according to Algorithm 1, the reconstruction errors are in a very similar range for all considered optic flow methods. Interestingly, we observe that computed flow fields are accurate enough to outperform in many cases the ground truth flow (rounded to the nearest grid point).


(a) Optimal mask **(b)** Details **(c)** Reconstruction

(d) Shifted mask **(e)** Details **(f)** Reconstruction

Fig. 7 **a**, **d** Inpainting masks (5.63% density in **a** and 4.13% in **d**) with **b**, **e** magnified details and **c**, **f** corresponding reconstructions for frame 15 of the Yosemite sequence. Black pixels indicate mask pixels, grey regions are to be inpainted. *Top:* optimal mask, *Bottom:* shifted mask

The best results are achieved with the TV-L_1 model presented in [40]. When considering the plots in Fig. 3, one sees that there is a clear benefit of using computed flow fields in the first 7 or 8 frames of the sequence, when comparing to a flow field that is zero everywhere. Afterwards, the iterative shifting of the masks has accumulated too many errors to outperform a zero flow. This suggests that the usage of a flow field is mostly beneficial for a short time prediction of the mask. Let us also note that the impact of the quality of the computed optic flow is visible over a shorter period within the first 5 frames.

The outcome of this experiment is very different when using Algorithm 2, as can also be seen in Fig. 4. Since the rounding errors are not accumulated across all frames, reconstructions with any of the considered optic flow methods clearly outperform static masks. Also, when comparing with Fig. 2, one can see that more accurate optic flow methods lead to lower reconstruction errors. The more modern methods (involving coarse-to-fine warping strategies) are accurate enough to lead to reconstructions with similar quality compared to those obtained with the ground truth flow. The TV-L_1 methods from [40] and [29] with b-spline interpolation can even outperform the ground truth flow across all frames.

4.4 Evaluation of the Density

Here we briefly evaluate the development of the density of mask pixels at the basis of masks shifted according to Algorithm 2.

The Yosemite sequence is a simulated flight through the Yosemite valley. Therefore, between two frames there is always some image content that is moved outside of the image plane. Consequently, and since the considered optic flow models also include regularisers, some regions of the flow field are pointing outside of the image. As can be seen in Fig. 5, the density descent is rather steadily, reflecting the smooth change in perspective across the image sequence. On average there are 25.1% of mask points dropped across the sequence. Between two frames there are 2.05% of the mask points dropped on average.

In Fig. 6, the density is displayed for the Walter and Toy Vehicle sequences. The scenes are more static than the one of the Yosemite sequence. The perspective is constant and the background is not changing. Movement of image content is not in proximity of the image boundary in most cases. As a result, the density of mask pixels is relatively stable, with on average 9.66%/2.98% dropped mask points per sequence and 0.674%/0.336% of dropped mask points between two frames for the Walter / Toy Vehicle sequence.

The number of mask points can be viewed as the budget for the reconstruction process. In a scenario were this budget is constant for all frames, one could redistribute the dropped mask points within the image plane. This may also be coupled with the detection of occlusions, such that mask points are redistributed in regions of objects that are not visible in previous frames. However, this is not investigated further in our current work.

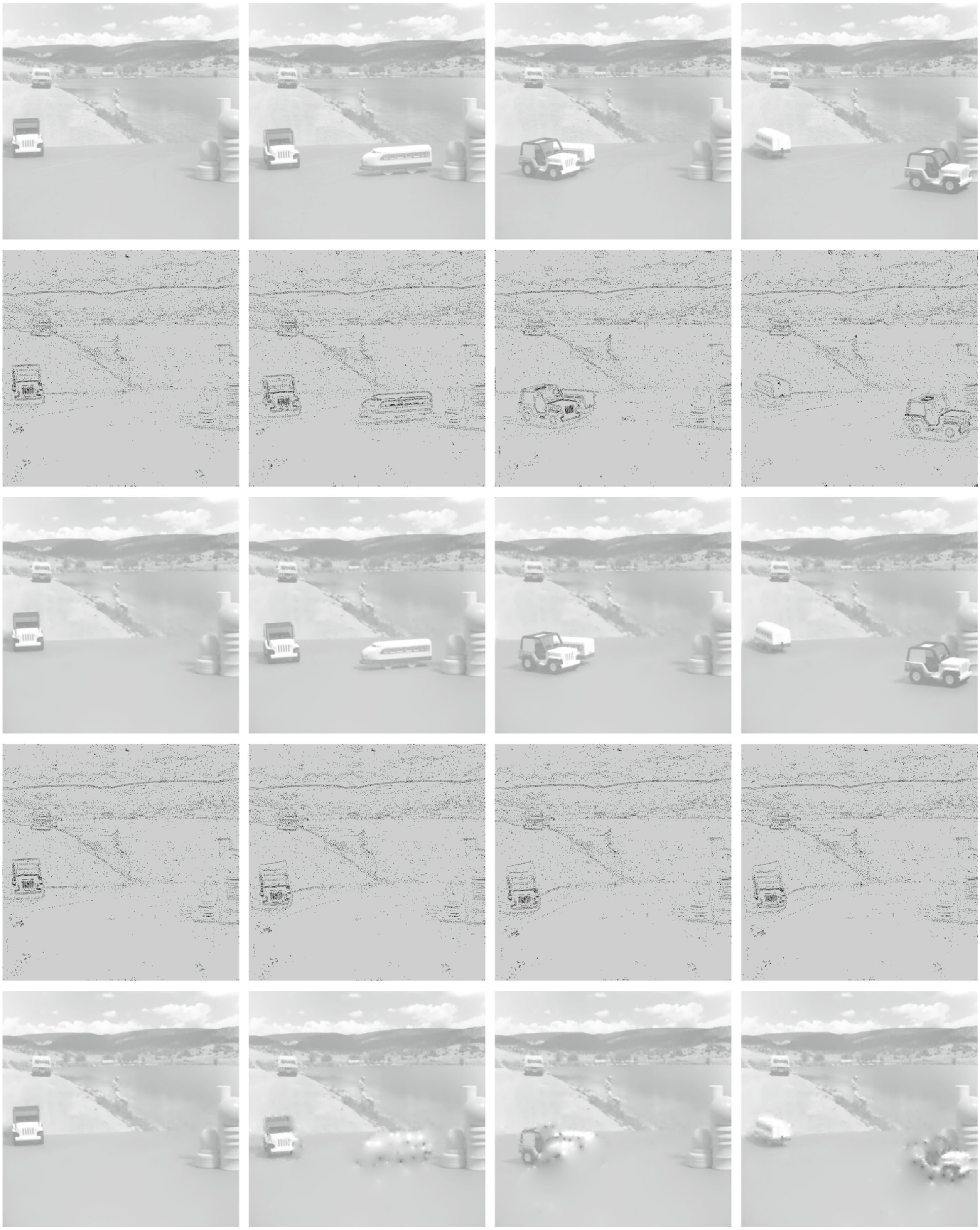


Fig. 8 Toy Vehicle sequence, frames *(left-to-right)* 1, 4, 7 and 10. Displayed are *(top-to-bottom)* the original images, optimal masks with densities of 3.00% to 3.50%, images reconstructed with optimal masks, shifted masks with densities of 2.98% to 3.00%, images reconstructed with shifted masks. Optical flow was computed according to [40] with $\alpha = 10$ in (6). The train model is not present in the first frame, hence it has too few mask points allocated. The magnitude of the estimated flow field for the car is too small, hence the associated mask points stay on the left side

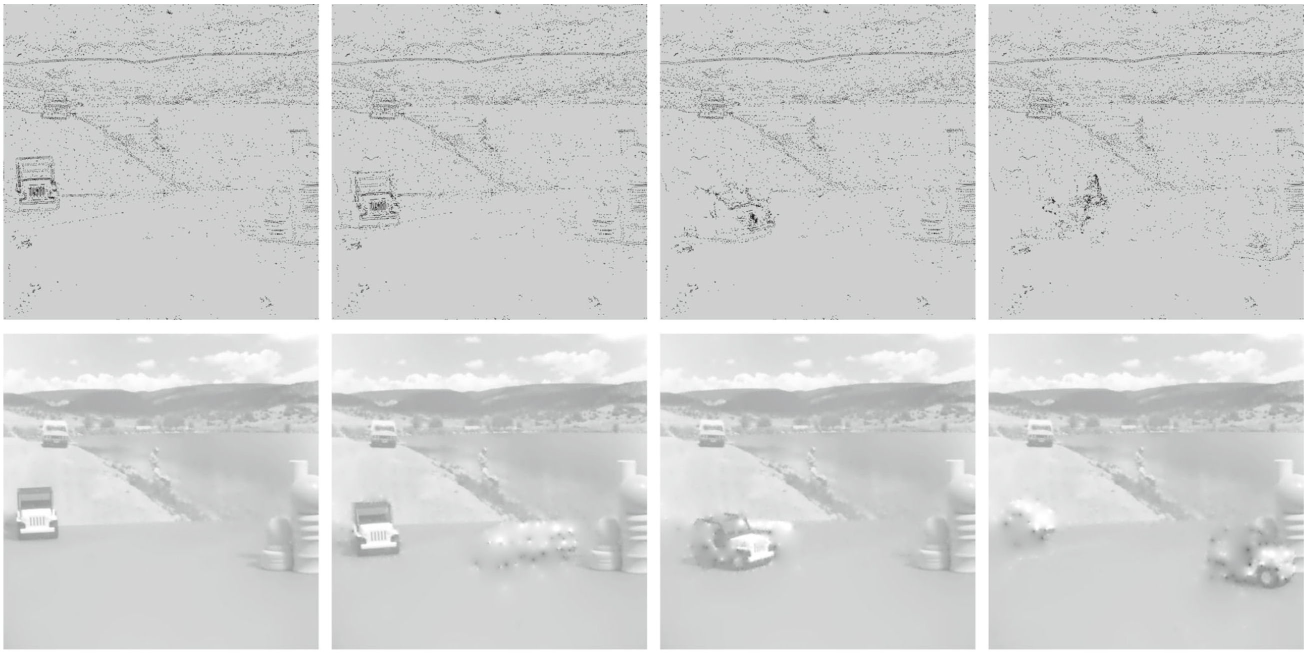


Fig. 9 Toy Vehicle sequence, frames *(left-to-right)* 1, 4, 7 and 10. Displayed are *(top-to-bottom)* shifted masks with densities of 2.86% to 3.00%, images reconstructed with shifted masks. Optical flow was computed according to [40] with $\alpha = 100$ in (6). The train model is not present in the first frame, hence it has too few mask points allocated

4.5 Evaluation of the Reconstruction Error

Overall, the error evolution, as observed in the Yosemite sequence, is rather steady and predictable, even though such a behaviour can only be expected in well-behaved sequences. The Toy Vehicle sequence from [37] exhibits strong occlusions and non-monotonic behaviour of the error, see Table 2. Nevertheless, the behaviour of the error evolution could be used to automatically detect frames after which a full mask optimisation becomes again necessary.

Figure 7 presents an optimal mask for the last frame of the Yosemite sequence as well as the shifted mask. The corresponding reconstructions are also depicted. Fine details are lost with the reconstruction from the shifted mask, e.g. some of the object boundaries are blurred in comparison with the optimal mask. However, the overall structure of the scene remains preserved. We remark that the bright spots appear due to our choice of the inpainting operator, see [13].

In Fig. 8, we examine the Toy Vehicle sequence, which contains large occlusions as well as large motions. When using the method from [40] with $\alpha = 10$ in (6), the magnitudes of the estimated flow field are too small. For this parameter choice, the highest magnitudes of the flow fields in each frame are between 1.8 and 8.8. Consequently, the mask locations for the car do not move anymore after the first few frames. With a regularisation parameter of $\alpha = 100$, the highest magnitudes are between 21 and 79, resulting in more accurate mask movement and reconstructions, as can be seen in Fig. 9. This highlights the dependence on a somewhat reliable flow field.

Finally, Table 2 contains further evaluations of the MSE as well as the SSIM for the image sequences from [37]. Both measures show a similar behaviour. Denser masks have higher SSIM (resp. lower MSE), and the SSIM decreases (resp. MSE increases) with the number of considered frames. The error evolution is usually monotone. However, if occlusions occur, then important mask pixels may be badly positioned or even completely absent. In that case, notable fluctuations in the error will occur. This is especially visible in the "Toy Vehicle" sequence where the maximal error is not the error in the last frame.

For almost all sequences, Algorithm 2 leads to better reconstructions than the previously in [19] proposed Algorithm 1. Therefore, we omit the results for Algorithm 1 in Table 2 and refer to [19] for the corresponding error values. For the Toy Vehicle sequence, the error measures are very similar, due to the usage of inaccurate flow fields in both algorithms.

5 Summary and Conclusion

Our work shows that it is possible to replace the expensive frame-wise computation of optimal inpainting data with the simple computation of a displacement field. Since run times to compute the latter are almost negligible when compared to the former, we gain a significant increase in performance. Our experiments demonstrate that simple and fast optic flow methods are sufficient for the task at hand, yet one may spend higher attention to movement of object boundaries.

In addition, the loss in accuracy along the temporal axis can easily be predicted. We may decide automatically when it becomes necessary to recompute an optimal mask while traversing the individual frames. We conjecture that the presented insights and the documented computational aspects on possible design choices will be helpful in the future development of PDE-based video compression techniques.

Acknowledgements Open Access funding provided by Projekt DEAL.

Funding The work was supported by Deutsche Forschungsgemeinschaft (Grant No. GZ: BR 2245/4-1), Graduate Research School of the Brandenburg Technical University Cottbus-Senftenberg (Grant No. StochMethod TP 1).

References

1. Andris, S., Peter, P., Weickert, J.: A proof-of-concept framework for PDE-based video compression. In: Proceedings of 32nd Picture Coding Symposium. IEEE (2016)
2. Belhachmi, Z., Bucur, D., Burgeth, B., Weickert, J.: How to choose interpolation data in images. SIAM J. Appl. Math. **70**(1), 333–352 (2009)
3. Bertalmío, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: Proceedings of 27th Annual Conference on Computer Graphics and Interactive Techniques, pp. 417–424. ACM Press/Addison-Wesley Publishing Company (2000)
4. Black, M.J.: Image sequences (2018). http://cs.brown.edu/people/mjblack/images.html
5. Bloor, M., Wilson, M.: Generating blend surfaces using partial differential equations. Comput. Aided Des. **21**(3), 165–171 (1989)
6. Brinkmann, E.M., Burger, M., Grah, J.: Regularization with sparse vector fields: From image compression to TV-type reconstruction. In: Aujol, J., Nikolova, M., Papadakis, N. (eds.) Scale Space and Variational Methods in Computer Vision. LNCS, vol. 9087, pp. 191–202. Springer, Berlin (2015)
7. Bruhn, A., Weickert, J., Feddern, C., Kohlberger, T., Schnörr, C.: Variational optical flow computation in real time. IEEE Trans. Image Process. **14**(5), 608–615 (2003)
8. Carlsson, S.: Sketch based coding of grey level images. Signal Process. **15**, 57–83 (1988)
9. Chambolle, A., Pock, T.: A first order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. **40**(1), 120–145 (2011)
10. Chen, Y., Ranftl, R., Pock, T.: A bi-level view of inpainting-based image compression. In: Z. Kúkelová, J. Heller (eds.) Computer Vision Winter Workshop (2014)
11. Demaret, L., Iske, A., Khachabi, W.: Contextual image compression from adaptive sparse data representations. In: R. Gribonval (ed.) Proceedings of SPARS'09 (Signal Processing with Adaptive Sparse Structured Representations Workshop) (2009)
12. Facciolo, G., Arias, P., Caselles, V., Sapiro, G.: Exemplar-based interpolation of sparsely sampled images. In: D. Cremers, Y. Boykov, A. Blake, F.R. Schmidt (eds.) Proceedings of the 7th International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition, *LNCS*, vol. 5681, pp. 331–344. Springer (2009)
13. Galić, I., Weickert, J., Welk, M., Bruhn, A., Belyaev, A., Seidel, H.P.: Towards PDE-based image compression. In: Paragios, N., Faugeras, O., Chan, T., Schnörr, C. (eds.) Variational, Geometric and Level-Set Methods in Computer Vision. LNCS, vol. 3752, pp. 37–48. Springer, Berlin (2005)
14. Guillemot, C., Meur, O.L.: Image inpainting: overview and recent advances. IEEE Signal Process. Mag. **31**(1), 127–144 (2014)
15. Hoeltgen, L., Mainberger, M., Hoffmann, S., Weickert, J., Tang, C.H., Setzer, S., Johannsen, D., Neumann, F., Doerr, B.: Optimising spatial and tonal data for PDE-based inpainting. In: M. Bergounioux, G. Peyré, C. Schnörr, J.B. Caillau, T. Haberkorn (eds.) Variational Methods, no. 18 in Radon Series on Computational and Applied Mathematics, pp. 35–83. De Gruyter (2016)
16. Hoeltgen, L., Weickert, J.: Why does Non-binary Mask Optimisation Work for Diffusion-Based Image Compression?, pp. 85–98. Springer, Berlin (2015)
17. Hoeltgen, L., Setzer, S., Weickert, J.: An Optimal Control Approach to Find sparse Data for Laplace Interpolation. Energy Minimization Methods in Computer Vision and Pattern Recognition. LNCS, vol. 8081, pp. 151–164. Springer, Berlin (2013)
18. Hoeltgen, L., Peter, P., Breuß, M.: Clustering-based quantisation for PDE-based image compression. Signal Image Video Process. **12**(3), 411–419 (2018)
19. Hoeltgen, L., Breuß, M., Radow, G.: Towards PDE-based video compression with optimal masks and optic flow. In: Lellmann, J., Burger, M., Modersitzki, J. (eds.) Scale Space and Variational Methods in Computer Vision. Lecture Notes in Computer Science, pp. 79–91. Springer, Berlin (2019)
20. Hoeltgen, L., Kleefeld, A., Harris, I., Breuß, M.: Theoretical foundation of the weighted laplace inpainting problem. Appl. Math. **64**(3), 281–300 (2019)
21. Horn, B.K., Schunck, B.G.: Determining optical flow. Artif. Intell. **17**(1–3), 185–203 (1981)
22. Johanson, P., Skelboe, S., Grue, K., Andersen, J.D.: Representing signals by their toppoints in scale space. In: Proceedings of 8th IARP 1986, pp. 215–217 (1986)
23. Mainberger, M., Bruhn, A., Weickert, J., Forchhammer, S.: Edge-based compression of cartoon-like images with homogeneous diffusion. Pattern Recognit. **44**(9), 1859–1873 (2011)
24. Mainberger, M., Hoffmann, S., Weickert, J., Tang, C.H., Johannsen, D., Neumann, F., Doerr, B.: Optimising spatial and tonal data for homogeneous diffusion inpainting. In: Bruckstein, A.M., Haar ter Romeny, B.M., Bronstein, A.M., Bronstein, M.M. (eds.) Scale Space and Variational Methods in Computer Vision. LNCS, vol. 6667, pp. 26–37. Springer, Berlin (2012)
25. Masnou, S., Morel, J.M.: Level lines based disocclusion. In: Proceedings of1998 IEEE International Conference on Image Processing, vol. 3, pp. 259–263. IEEE (1998)

26. Ochs, P., Chen, Y., Brox, T., Pock, T.: iPiano: inertial proximal algorithm for nonconvex optimization. SIAM J. Imaging Sci. **7**(2), 1388–1419 (2014)
27. Papenberg, N., Bruhn, A., Brox, T., Didas, S., Weickert, J.: Highly accurate optic flow computation with theoretically justified warping. Int. J. Comput. Vis. **67**(2), 141–158 (2006)
28. Peter, P., Hoffmann, S., Nedwed, F., Hoeltgen, L., Weickert, J.: From optimised inpainting with linear PDEs towards competitive image compression codecs. In: Bräunl, T., McCane, B., Rivers, M., Yu, X. (eds.) Image and Video Technology, LNCS, vol. 9431, pp. 63–74. Springer, Basel (2016)
29. Radow, G., Breuß, M.: Variational optical flow: warping and interpolation revisited. In: Vento, M., Percannella, G. (eds.) Computer Analysis of Images and Patterns. LNCS, pp. 409–420. Springer, Berlin (2019)
30. Schmaltz, C., Peter, P., Mainberger, M., Ebel, F., Weickert, J., Bruhn, A.: Understanding, optimising, and extending data compression with anisotropic diffusion. Int. J. Comput. Vis. **108**(3), 222–240 (2014)
31. Schönlieb, C.B.: Partial Differential Equation Methods for Image Inpainting, Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge (2015)
32. Shen, J., Chan, T.F.: Mathematical models for local nontexture inpaintings. SIAM J. Appl. Math. **62**(3), 1019–1043 (2002)
33. Strutz, T.: Bilddatenkompression. Vieweg (2002)
34. Sullivan, G.J., Wiegand, T.: Video compression—from concepts to the H. 264/AVC standard. In: Proceedings of the IEEE, vol. 93, pp. 18–31 (2005)
35. Sun, D.: (2018). http://research.nvidia.com/person/deqing-sun
36. The Mathworks Inc.: Compute optical flow using Horn-Schunck method (2018). https://de.mathworks.com/help/vision/ug/compute-optical-flow-using-horn-schunck-method.html
37. The USC-SIPI image database (2014). http://sipi.usc.edu/database/
38. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
39. Weinzaepfel, P., Jégou, H., Pérez, P.: Reconstructing an image from its local descriptors. In: Proceedings of 2011 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 337–344. IEEE Computer Society Press (2011)
40. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime TV-L^1 optical flow. In: Hamprecht, F.A., Schnörr, C., Jähne, B. (eds.) Pattern Recognition, LNCS, pp. 214–223. Springer, Berlin (2007)

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

Michael Breuß received the doctorate degree in mathematics from the University of Hamburg in 2001, and the habilitation in mathematics from the Technical University in Brunswick in 2006. Since 2016 he is professor for applied mathematics at the Brandenburg Technical University in Cottbus, Germany. His research interests are mainly in mathematical image processing and 3D vision, and include in particular numerical methods.

Laurent Hoeltgen is a postdoctoral researcher at the Brandenburg Technical University in Cottbus, Germany. He received his Ph.D. in mathematics from Saarland University, in 2015. His research interests include image processing with partial differential equations, mathematical optimisation and scientific computing.

Georg Radow is a doctoral researcher at the Brandenburg Technical University in Cottbus, Germany. He received his M.Sc. in applied mathematics from Brandenburg Technical University Cottbus-Senftenberg, in 2015. His research interests are optimisation methods in image processing, in particular correspondence problems.