Optical flow based interpolation of temporal image sequences

Jan Ehrhardt[a], Dennis Säring[a] and Heinz Handels[a]

[a]Department of Medical Informatics, University Medical Center Hamburg-Eppendorf,
Germany

ABSTRACT

Modern tomographic imaging devices enable the acquisition of temporal image sequences. In our project, we study cine MRI sequences of patients with myocardial infarction. Because the sequences are acquired with different temporal resolutions, a temporal interpolation is necessary to compare images at predefined phases of the cardiac cycle.

This paper presents an interpolation method for temporal image sequences. We derive our interpolation scheme from the optical flow equation. The spatiotemporal velocity field between the images is determined using an optical flow based registration method. Here, an iterative algorithm is applied, using the spatial and temporal image derivatives and a spatiotemporal smoothing step. Afterwards, the calculated velocity field is used to generate an interpolated image at the desired time by averaging intensities between corresponding points.

The behavior and capability of the algorithm is demonstrated by synthetic image examples. Furthermore, quantitative measures are calculated to compare this optical flow based interpolation method to linear interpolation and shape–based interpolation in 5 cine MRI data sets. Results indicate that the presented method outperforms both linear and shape–based interpolation significantly.

Keywords: interpolation, optical flow, image registration, temporal images, cine MRI, cardiac imaging

1. INTRODUCTION

Modern tomographic imaging devices enable the acquisition of temporal image sequences and the study of organ motion becomes more and more important. For example cine MRI is used for the functional analysis of the heart[1] and 4D–CT data sets are used for modelling organ motion during the respiratory cycle.[2] But, in general, the spatial and temporal resolution of imaging devices is limited and a compromise between spatial resolution, temporal resolution, acquisition time and signal to noise ratio must be found. Therefore, in a number of image processing tasks a spatial and temporal interpolation of data sets is necessary to calculate dense motion models for instance. In our project, we compare cardiac cine MRI sequences of different patients, acquired with different temporal resolutions. A temporal interpolation of the image data is necessary to generate images at predefined phases of the cardiac cycle.

Interpolation is commonly used in medical image processing and is required, whenever the acquired image data is not at the same level of discretization as desired or whenever geometric transformations of the image data are necessary. There are numerous techniques for the spatial interpolation of images. Grevera and Udupa (1998)[3] divide spatial interpolation methods into two groups: intensity–based methods and shape–based methods. See Lehmann et al. (1999)[4] and Meijeriing et al. (2001)[5] for a comparison of different intensity–based methods and Grevera and Udupa (1998)[3] for a comparison of intensity– and shape–based interpolation methods.

Any spatial interpolation algorithm can be used to generate image information for arbitrary time points, if the temporal $n\mathrm{D}+t$ image sequence is considered as a spatial image data set in $n+1$ dimensions. But, temporal dependence between objects in successive frames is lost and this conversion may cause problems, since the units of measurement of the variable in time–direction are different from those of the other (spatial) variables.

This paper describes a method for the interpolation between frames of a temporal image sequence based on non–linear registration. The approach consist of two steps. First, the optical flow between the temporal images

is determined using an optical flow based registration algorithm. Afterwards the calculated velocity field is used to generate an interpolated image at the desired time. The algorithm regards the spatiotemporal nature of the image sequence. So a conversion into a spatial data volume is not necessary.

Other registration–based interpolation methods for spatial 3D image volumes were already presented.[6,7] Goshtasby et al. (1992)[6] presented a registration–based method to interpolate between neighbouring slices in tomographic image data sets. The slices were registered using intensity and gradient features. To calculate the intensity values of the interpolated slice from the resulting displacement field, "interpolation lines" were calculated to identify corresponding points in two adjacent original slices. However, the paper is more than ten years old and the registration method used is not state of the art. Penney et al. (2004)[7] applied voxel–based registration using B–spline regularization and the normalized mutual information similarity measure to calculate correspondences between neighbouring slices. The interpolation step is similar to the method used by Goshtasby et al. (1992). Both methods are limited to consecutive slices and no theoretical motivations for the registration algorithm used and for the concept of "interpolation lines" were given. In particular, the use of the mutual information similarity measure was not motivated by Penney et al., in spite of existing faster and more reliable registration methods for monomodal data.

The theoretical motivation of our registration method as well as for the interpolation step is the optical flow equation. In contrast to the other approaches, the optical flow equation makes the use of more than two consecutive images for the calculation of the velocity field possible and the concept of the "interpolation lines" is no longer needed. First, the behavior and capability of the algorithm was demonstrated on synthetic example images. Furthermore, cine MRI data sets were used to perform a quantitative analysis and statistical measures were calculated to compare our interpolation algorithm to linear and shape–based interpolation.

2. METHODS AND MATERIALS

The presented interpolation method consists of two steps: First the time–dependend optical flow field is determined using a non–linear registration method. Following the calculated optical flow field is used to generate interpolated images for arbitrary time points. In section 2.3 the evaluation methods are explained in detail.

2.1. Determining the optical flow

The initial hypothesis of optical flow based methods is that pixel intensities of time varying image regions remain constant. The conservation of the intensity of points under motion is formulated in the expression that the total derivative of the image function is zero[8]:

$$\frac{dI(\boldsymbol{x}(t),t)}{dt} = 0, \tag{1}$$

where $I : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}$ is the image intensity function*. From the intensity conservation assumption (1) follows for the image $I(\boldsymbol{x}(t),t)$ at time $t = t_0 + \delta t$:

$$I(\boldsymbol{x}(t),t) = I(\boldsymbol{x}(t - \delta t), t - \delta t) = I(\boldsymbol{x}(t_0), t_0). \tag{2}$$

For a small time step δt we can approximate

$$x(t - \delta t) = x(t) - \delta t \frac{\partial x}{\partial t} + O^2,$$

where O^2, the 2nd and higher order terms, are assumed neglible. Eq. (2) becomes

$$I(\boldsymbol{x}(t),t) = I(\boldsymbol{x}(t) - \delta t \cdot \boldsymbol{v}, t - \delta t), \tag{3}$$

where $\boldsymbol{v} = \left(\frac{\partial x}{\partial t}, \frac{\partial y}{\partial t}\right)^T$ is the (two–dimensional) velocity field. Thus, if the velocity field $\boldsymbol{v}$ is known, we can interpolate the image at time t from an image at time $t_0 = t - \delta t$.

---

*Here, we restrict to the two–dimensional case, but the extension to other dimensions is straight forward.

A number of different optical flow methods have been proposed to estimate the velocity field from a set of images (see Barron et al. (1994)[9] for a survey and comparison). From the optical flow equation (1) we obtain

$$\boldsymbol{v} = -\nabla I \frac{\partial_t I}{\|\nabla I\|^2}. \tag{4}$$

where ∇I is the spatial image gradient. However, equation (4) is ill–posed and not sufficient to compute both components of the velocity vector.[8,9] Only the motion component in the direction of the local brightness gradient ∇I of the image intensity function may be estimated. As a consequence, the flow velocity cannot be computed locally without introducing additional constrains. In our implementation the necessary regularization is done by a spatiotemporal Gaussian smoothing of the velocity field. The smoothing step limits the possible differences between magnitude and direction of spatially and temporally neighbouring velocity vectors. The temporal derivative $\partial_t I$ can be computed by finite differences or by a convolution with a Gaussian derivative in time direction to take more than two consecutive frames into account. The spatial derivative can be approximated by averaging the gradients of neighbouring image frames $\nabla I = (I(\boldsymbol{x}, t_i) + \nabla I(\boldsymbol{x}, t_{i+1}))/2$. The resulting optical flow equation

$$\boldsymbol{v} = -\frac{2(\nabla I(\boldsymbol{x}, t_i) + \nabla I(\boldsymbol{x}, t_{i+1}))\ \partial_t I}{\|\nabla I(\boldsymbol{x}, t_i) + \nabla I(\boldsymbol{x}, t_{i+1})\|^2 + k} \tag{5}$$

has the disadvantage that the gradient has to be recomputed at each iteration. But, for the interpolation step the inverse velocity field is needed and thus the symmetric behaviour of eq. (5) is advantageous. The constant k is estimated from local image properties and is used to stabilize the calculation.

The registration algorithm has two parameters: the variance of the spatiotemporal Gaussian smoothing function and the stop criterion.

2.2. Optical flow based interpolation of temporal images

From the intensity conservation assumption (eq. (3)) follows for the image $I(\boldsymbol{x}, t)$ at time $t = t_0 + \delta t$:

$$I(\boldsymbol{x}(t), t) \approx I(\boldsymbol{x}(t) - \delta t \cdot \boldsymbol{v}, t_0). \tag{6}$$

Thus, if the velocity field $\boldsymbol{v}$ is known, we can interpolate the image at time t from an image at time t_0. But in general the intensity conservation assumption might not be fulfilled and structures may appear or disappear between two time steps. Therefore, we use a weighted average between corresponding voxels in the adjacent time frames $I(\boldsymbol{x}, t_i)$ and $I(\boldsymbol{x}, t_{i+1})$:

$$I(\boldsymbol{x}(t), t) = (1 - \delta t) \cdot I\left(\boldsymbol{x}(t) - \delta t \boldsymbol{v}, t_i\right) + \delta t \cdot I\left(\boldsymbol{x}(t) - (1 - \delta t)\boldsymbol{v}^{-1}, t_{i+1}\right), \tag{7}$$

with $t_i < t < t_{i+1}$, $\delta t = t - t_i$ and a normalized time step $t_{i+1} - t_i = 1$. In general, the inverse velocity field $\boldsymbol{v}^{-1}$ can't be computed directly. In our interpolation scheme an iterative Newton–Raphson method is used to calculate the inverse velocity for each grid point.[10]

Problems can arise near image borders, if $\boldsymbol{x}(t) - \delta t \boldsymbol{v}$ or $\boldsymbol{x}(t) - (1 - \delta t)\boldsymbol{v}^{-1}$ lie outside the frame. In these cases the position in the interpolated image is marked with an out of data flag.

2.3. Evaluation methods

Our interpolation method relies on two assumptions: the intensity conservation assumption as formalized in eq. (1) and that the algorithm described in section 2.1 is capable to calculate the correct velocity field $\boldsymbol{v}$.

In a first evaluation study we try to gain insight into how the algorithm works. Therefore we generated two synthetic phantoms.

Phantom I – moving disk: Three images of a white disk at different locations were generated (see fig. 1). The disks were translated by $(5, 15)$ and $(10, 30)$ pixels in (x, y)–direction. Our aim was to estimate the middle disk image by the interpolation methods and to compare these to the phantom.
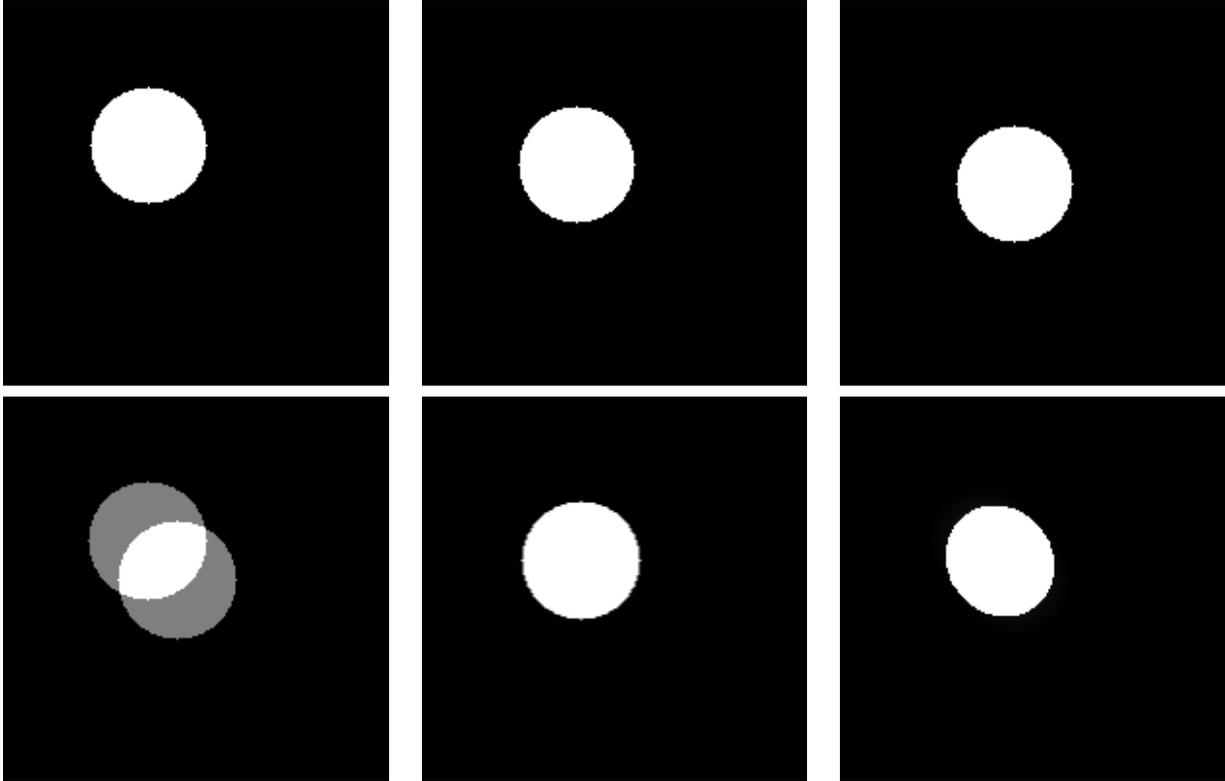
Figure 1. Top row: Three frames of a moving disk (phantom I). Bottom row: Estimates of the middle image of the phantom by linear interpolation (left), optical flow based interpolation (middle) and shape–based interpolation (right). The optical flow–based method produces the most accurate result.

Phantom II – disk and disk with hole: A disk image and a corresponding disk image with a varying radius and a hole in the middle were generated (see fig. 2). For this image sequence the intensity conservation assumption is violated. The aim was to evaluate the behavior of the algorithm in this case.

In a second evaluation procedure we calculated quantitative measures to compare our interpolation method with two other methods: linear and shape–based interpolation.[11] Linear interpolation is the most frequently used interpolation technique and was chosen as a baseline reference. The shape–based interpolation algorithm was chosen since it was shown to have the best performance in a comparison of interpolation methods.[3] The shape based interpolation was implemented according to.[11]

For the quantitative evaluation five cardiac MRI datasets (ECG-triggered true FISP sequences, between 13 and 21 time frames, 224×256 pixels) were used. One step of the shape–based interpolation is to convert the n–D grey value image into a $(n + 1)$–D binary image (the so–called lifting). Afterwards a $(n + 1)$–D distance transform is performed. For images with a large number of grey values these steps can create computational problems, both in CPU time and memory requirements. Therefore, the image intensities were limited to 256 grey values.

For evaluation each frame of the temporal image sequence (apart from the first and the last slice) was removed one at a time and the three interpolation methods were used to interpolate between the neighboring image frames. Finally, the interpolated images were compared to the original removed images. In conformity with the paradigm proposed by Grevera and Udupa[3] three error measures were used:

Mean difference (MD): For one sequence, $I_\tau^{int}(\boldsymbol{x})$ and $I_\tau^{orig}(\boldsymbol{x})$ represent the intensity value at pixel position $\boldsymbol{x}$ in the τ–th frame of the interpolated and the original image. N_τ denotes the number of interpolated
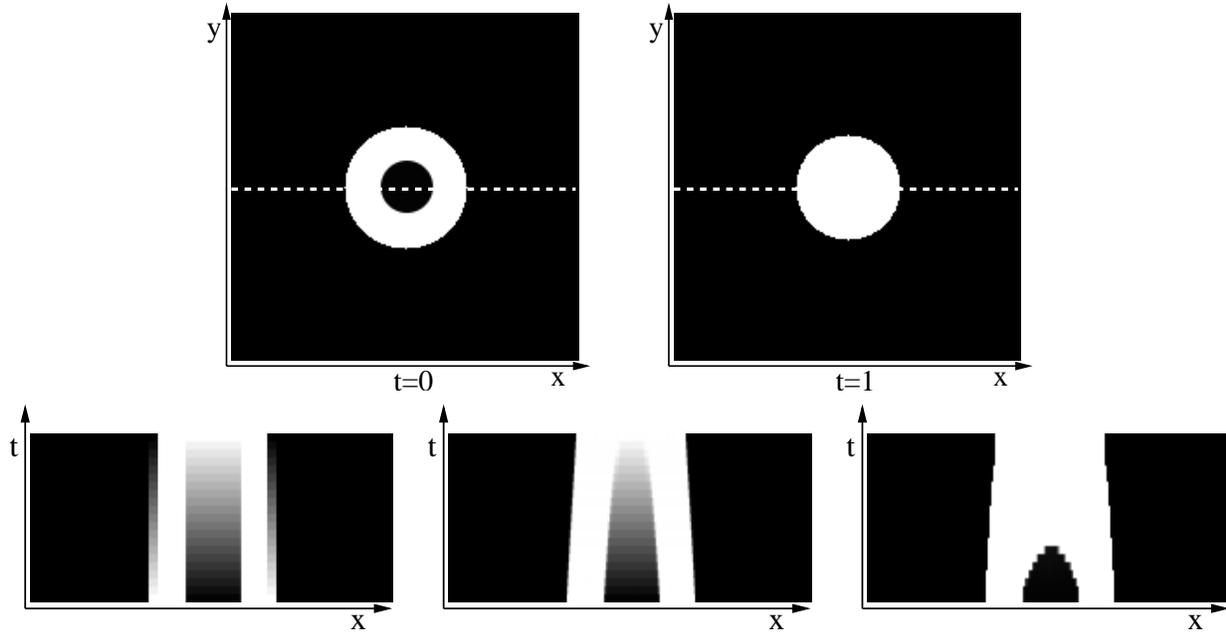
Figure 2. Top row: The two images of phantom II. Bottom row: Interpolations between the phantom images with varying time steps. Each row shows one line in the interpolated image for a given time step. The position of the line is indicated by the dotted line in the top images. Bottom left: linear interpolation, bottom middle: optical flow based interpolation and bottom right: shape–based interpolation. Although the intensity conservation assumption is violated the optical flow–based method produces satisfactory results. In this case a compromise between linear and shape–based interpolation was found.

images and Ω_τ the set of pixels in frame τ. The MD is defined by:

$$MD = \frac{1}{N_\tau} \sum_{\tau=1}^{N_\tau} \frac{1}{|\Omega_\tau|} \sum_{\boldsymbol{x} \in \Omega_\tau} |I_\tau^{int}(\boldsymbol{x}) - I_\tau^{orig}(\boldsymbol{x})| \tag{8}$$

Number of sites of disagreement (NSD): This measure is the number of pixels where the difference between the $I_\tau^{int}(\boldsymbol{x})$ and $I_\tau^{orig}(\boldsymbol{x})$ is greater than a threshold θ:

$$NSD = \sum_{\tau=1}^{N_\tau} \sum_{\boldsymbol{x} \in \Omega_\tau} \delta(|I_\tau^{int}(\boldsymbol{x}) - I_\tau^{orig}(\boldsymbol{x})|), \text{ where } \delta(z) = \left\{ \begin{array}{ll} 0, & \text{if } z < \theta \\ 1, & \text{otherwise} \end{array} \right. \tag{9}$$

In our evaluation a threshold of 5% of the maximum intensity value in the image was chosen.

Largest difference per frame (LD): This represents the maximum difference between corresponding pixel values:

$$LD(\tau) = \max_{\boldsymbol{x} \in \Omega_\tau} \left(|I_\tau^{int}(\boldsymbol{x}) - I_\tau^{orig}(\boldsymbol{x})| \right). \tag{10}$$

For all error measures, pixels containing out of data flags were excluded from the calculation.

To compare two interpolation methods a measure called statistical relevance was used. This measure expresses the degree of importance of the observed difference between the methods, e.g. the statistical relevance between the linear MD and the optical flow based MD is given by:

$$r_{flow/lin}^{MD} = \left\{ \begin{array}{ll} +100 \cdot \left(1 - \frac{MD_{flow}}{MD_{lin}}\right), & \text{if } MD_{lin} > MD_{flow} \\ -100 \cdot \left(1 - \frac{MD_{flow}}{MD_{lin}}\right), & \text{otherwise} \end{array} \right. . \tag{11}$$

data set	statistical relevance					
	flow/lin			flow/shape		
	r^{MD}	r^{NSD}	$\overline{r^{LD}}$	r^{MD}	r^{NSD}	$\overline{r^{LD}}$
MRI 01	3.35	12.11	14.46	3.4	19.91	20.72
MRI 02	–	13.33	21.95	1.47	24.65	26.3
MRI 03	5.49	9.35	9.97	3.05	8.19	16.07
MRI 04	3.69	8.63	–	2.84	9.18	17.68
MRI 05	–	3.07	7.12	3.02	14.24	12.75

Table 1. Statistical relevance values of *mean difference (MD)*, *number of sites of disagreement (NSD)* and *largest difference per frame (LD)* to compare the optical flow–based interpolation method with linear and shape–based interpolation. Positive values indicate the optical flow method performed better. A dash in the table indicates that the difference between the two methods was not statistically significant (paired student's t–test, $p \leq 0.05$).

3. RESULTS

Fig. 1 shows the three frames of phantom I and the results of interpolation of the middle image. The linear method shows a shadowing effect and the shape–based interpolation induces a slight deformation of the disk. The optical flow based method produces the most accurate result. Because large displacements were necessary, while only sparse gradient information is available, we used $\sigma^2 = 5$ for the spatial Gaussian smoothing of the deformation field in this case.

In a second qualitative evaluation we interpolated 20 frames between the original images of phantom II at varying time steps $\delta t \in [0, 1]$. The bottom row of fig. 2 gives a comprehensive sketch of the behavior of the different interpolation methods. Each row of the images show one line of the interpolated image for a given time step. Although the intensity constrain is violated the optical flow based method produced satisfactory results. In this case a compromise between intensity–based and shape–based interpolation was found.

Thirdly, 5 cine MRI sequences were considered. In all cases, a factor of $\sigma^2 = 1.5$ was used for spatial and temporal smoothing. Due to the cyclic behavior of the cardiac image sequences periodic boundary conditions are used for the Gaussian convolution in time direction. The velocity field was calculated according to eq. (5), where the temporal derivative $\partial_t I$ was approximated by finite differences. More complex approaches for the computation of the temporal derivative led to substantially longer computation times but not to substantially improved interpolation results.

Table 1 shows the statistical relevance of the error measures MD and NSD and the mean statistical relevance of LD (averaged over the frames) to compare the optical flow based interpolation*(flow)* with the linear*(lin)* and shape–based*(shape)* interpolation methods. Positive values indicate the first method performed better. A dash in the table indicates that the difference between the two methods was not statistically significant (paired student's t–test, $p \leq 0.05$). The results in table 1 show that the optical flow based interpolation outperformed linear and shape–based interpolation in most cases, significantly. In contrast to NSD and LD only a slight improvement of the mean difference (MD) is indicated. Since a large part of the displayed structures does not change over the cardiac cycle the mean difference is strongly influenced by noise.

The most noticeable improvements by our method were observed, where the image structures change between adjacent frames considerably. In fig. 3 sample images estimated by the three interpolation methods and corresponding difference images were shown. The linear interpolated image appears blurred and large differences can be observed. The shape–based interpolation conserves edges of image structures but small details are lost. The optical flow based interpolation performs more accurately and only few differences are shown in the difference image.

4. DISCUSSION AND CONCLUSION

We presented a method for interpolating temporal image sequences. The interpolation algorithm was theoretically derived from the optical flow equation and the performance of the algorithm was evaluated qualitatively and quantitatively. The quantitative results show that the optical flow based method clearly outperforms the linear
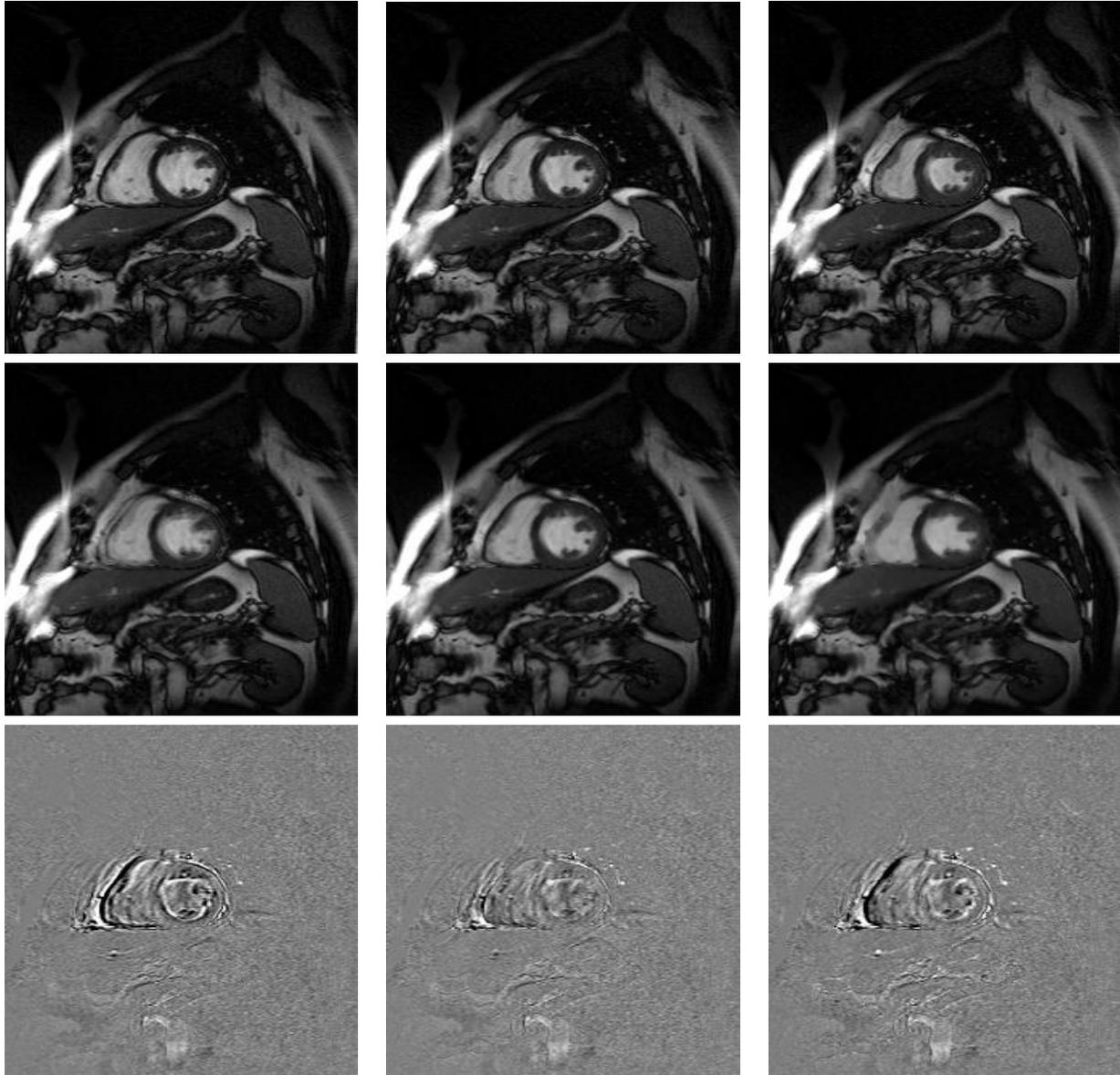
Figure 3. Top row: Three consecutive slices of a cine MR sequence. The interpolation results for the second slice are shown in the middle row. Middle row: Sample slice estimated by linear (left), optical flow based (middle) and shape–based (right) interpolation. Bottom row: Corresponding difference image $I^{int}(\boldsymbol{x}) - I^{orig}(\boldsymbol{x})$ between the interpolated and original slice. The linear interpolated image (left) appears blurred and large differences can be observed. The shape-based interpolation (right) conserves edges of image structures but small details are lost. The optical flow–based interpolation (middle) performs more accurately and only few differences are shown in the difference image.

and shape–based interpolation. The presented method is also applicable to interpolate between neighboring slices in spatial tomographic data sets. An evaluation of this approach is accomplished at present.

Furthermore, in our experiments the optical flow based method was computational less expensive than the shape–based interpolation. The performance of the shape–based method strongly depends on the implementation of the distance transformation. We used the algorithm of Danielsson (1980)[12] but faster algorithms exist.[13] However, in contrast to the shape–based method the performance of the optical flow based interpolation is

independent of the dynamic range of the images.

In contrast to the registration methods used by Goshtasby et al. (1992) and Penney et al. (2004), our algorithm shows a symmetric behavior. A drawback of our method is, that the Gaussian regularization does not guaranty that the optical flow field is a homeomorphism and that the inverse exists for all points. Therefore, further research will adress the integration of other regularization techniques. Furthermore, an evaluation of different strategies for the calculation of the optical flow field is planned.

REFERENCES

1. A. Frangi, W. Niessen, and M. A. Viergever, "Three–dimensional modeling for functional analysis of cardiac images: a review," *IEEE Trans. Medical Imaging* **20**(1), 2001.
2. D.-A. Low, M. Nystrom, E. Kalinin, P. Parikh, J.-F. Dempsey, J.-D. Bradley, S. Mutic, S.-H. Wahab, T. Islam, G. Christensen, D.-G. Politte, and B.-R. Whiting, "A method for the reconstruction of four-dimensional synchronized CT scans acquired during free breathing," *Med.-Phys.* **30**(6), pp. 1254–63, 2003.
3. G. J. Grevera and J. K. Udupa, "An objective comparison of 3–D image interpolation methods," *IEEE Trans. Medical Imaging* **17**(4), pp. 642–652, 1998.
4. T. M. Lehmann, C. Goenner, and K. Spitzer, "Survey: Interpolation methods in medical image processing," *IEEE Trans. Medical Imaging* **18**(11), 1999.
5. E. H. W. Meijering, W. J. Niessen, and M. A. Viergever, "Quantitative evaluation of convolution-based methods for medical image interpolation," *Med Image Anal* **5**, pp. 111–126, 2001.
6. A. Goshtasby, D. A. Turner, and L. V. Ackerman, "Matching of tomographic slices for interpolation," *IEEE Trans. Medical Imaging* **11**(4), 1992.
7. G. P. Penney, A. Schnabel, D. Rueckert, and M. A. Viergever, "Registration–based interpolation," *IEEE Trans. Medical Imaging* **23**(7), 2004.
8. B. K. P. Horn and B. G. Schunck, "Determining optical flow.," *Artificial Intelligence* **17**, pp. 185–203, Aug. 1981.
9. J. L. Barron, D. J. Fleet, and S. S. Beauchemin, "Performance of optical flow techniques," *Int. J. Comp. Vision* **12**(1), pp. 43–77, 1994.
10. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical recipes in C*, Cambridge University Press, 1992.
11. G. J. Grevera and J. K. Udupa, "Shape–based interpolation of multidimensional grey–level images," *IEEE Trans. Medical Imaging* **15**(6), pp. 881–892, 1996.
12. P.-E. Danielsson, "Euclidean distance mapping," *Comp Graph Img Proc* **14**, pp. 227–248, 1980.
13. C. Maurer, R. Qi, and V. Raghavan, "A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions," *IEEE Trans PAMI* **25**, pp. 265–270, 2003.