Spatial and Temporal Interpolation of Multi-View Image Sequences

Tobias Gurdan[1,2], Martin R. Oswald[1], Daniel Gurdan[2], Daniel Cremers[1]

[1]Department of Computer Science, Technische Universität München*
[2]Ascending Technologies GmbH, Krailing, Germany

Abstract. We propose a simple and effective framework for multi-view image sequence interpolation in space and time. For spatial view point interpolation we present a robust feature-based matching algorithm that allows for wide-baseline camera configurations. To this end, we introduce two novel filtering approaches for outlier elimination and a robust approach for match extrapolations at the image boundaries. For small-baseline and temporal interpolations we rely on an established optical flow based approach. We perform a quantitative and qualitative evaluation of our framework and present applications and results. Our method has a low runtime and results can compete with state-of-the-art methods.

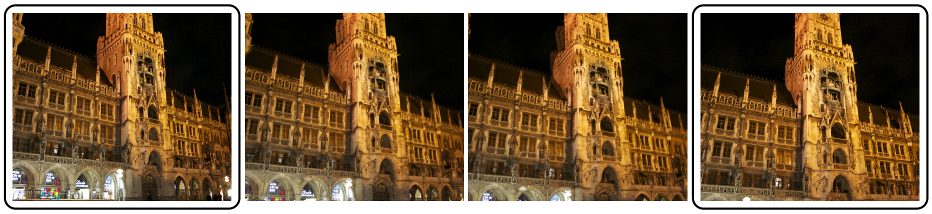


Fig. 1: Spatial view point interpolation between two images. Our approach robustly handles wide-baseline view point interpolations. *From left to right*: source image, interpolation at $t = 1/3$, interpolation at $t = 2/3$, target image.

1 Introduction

Image interpolation is one of the vital tools in image processing for the creation of visual effects, as can for example be seen in the movie "The Matrix" [3]. While it has been very tedious and costly to create such effects in the past, the rapid increase in computational power has reformed the movie industry substantially. Over the last few years, slow motion and immersive 3d effects had an enormous increase in both quality and quantity. However, there are still many open problems, including for instance free viewpoint video, which becomes more and more popular with the rise of 3d televisions [17]. In this paper, we tackle the challenge of spatiotemporal interpolation of image sequences from multiple, synchronised cameras. The main problem is to robustly estimate pixel correspondences between two given camera frames and subsequently interpolate the pixel motion

* This work was supported by the ERC Starting Grant "Convex Vision".

properly in order to synthesise novel in-between images convincingly. The result is a slow-motion effect by interpolating consecutive frames or a smooth transition from one camera to another as depicted in Fig. 1.

1.1 Related Work

There exist a variety of applications and respective approaches to spatial view point interpolation. Phototourism [21] is a prominent example. Goesele *et al.* [8] used point clouds to register camera positions and transition between views. In the field of sports, a promising technique was introduced by Germann *et al.* [7]. They make use of billboards and foreground-background separation to create a plausible illusion of depth while transitioning from one view to another. Inamato and Saito [10] as well as Replay Technologies Inc. [17] went one step further and applied 3d-registration and projection on top of layer segmentation. For casually captured multi-view video collections, Ballan *et al.* [1] combined a previously reconstructed background geometry and billboards of a foreground object to spatially navigate between camera views. In contrast to our approach, view interpolation is only done for the background geometry and concealed with a strong motion blur. Werlberger *et al.* [26] and Chen and Lorenz [2] addressed temporal image interpolation and suggested the use of optical flow correspondences. Mahajan *et al.* [14] estimated gradient rather than pixel correspondences.

The early years of feature based image morphing, on which our work is based, are covered by Wolberg [27] and more currently by Vlad [24]. In their pioneering work on spatial image interpolation Seitz and Dyer [20] proposed a pre- and postwarping scheme, which was later improved by Fragneto *et al.* [6]. It allows for physically plausible transitions, whereas former methods could lead to unrealistic deformations. One of the first free viewpoint videos filmed with consumer cameras was then presented by Zitnick *et al.* [29], who used layers, matting and a color segmentation-based stereo algorithm to estimate dense depth maps. Vedula *et al.* [23] estimated scene flow on a voxel model for spatiotemporal interpolation. The volume discretisation makes this approach costly to recover fine object details. Lipski *et al.* [12,11] recently proposed a dense matching scheme based on combined edgelet matching and color segmentation that allows free-viewpoint navigation for handheld multi-view video footage in both space and time. Their work is currently considered state of the art for dense image interpolation.

So far, no other work on spatiotemporal image interpolation can simultaneously cope with casual, wide baseline camera setups and preserve image quality in the interpolated results. Or they are not able to navigate in both space and time or need extensive computations. Our framework is designed to combine all aspects in a simple and effective manner.

1.2 Contribution

We present a framework for temporal and spatial multi-view image sequence interpolation. We introduce a novel pipeline for sparse feature matching, filtering and extrapolation that significantly improves correspondence quality compared to common matching techniques. Due to its robustness, our approach is suited even for wide baseline setups, where dense stereo matching approaches reach their limits. To the best of our knowledge feature-based spatiotemporal view point interpolation has not been explored in literature yet. In contrast to works like [1,23,14,26] we do not impose strong assumptions on the scene structure or the camera movement, allowing for arbitrary, uncalibrated input. In several experiments we study the strengths and weaknesses of sparse matching techniques for image interpolation on a variety of sequences.

2 Image Interpolation

In our approach the synthesis of novel views or in-between frames of image-sequences consists of three steps: matching, warping and blending. We consider two cases: spatial interpolation of camera motion between different camera views and temporal interpolation of similar images with dynamic content. The following subsections detail our approach to both cases with respect to the three processing steps, using either sparse or dense correspondences.

2.1 Spatial Interpolation

For spatial interpolation one needs to model camera motion in order to obtain plausible view point interpolations, especially in wide-baseline setups. We rely on established feature matching techniques to get an implicit, locally approximated motion representation. Such sparse matching techniques perform more robustly in wide-baseline setups compared to stereo or optical flow methods.

Instead of exhaustively searching the feature space with subsequent outlier filtering, we follow the ideas of Zhang *et al.* [28] who use the epipolar constraint to establish matches in the first place, limiting the search space of potential matches from two dimensions to a relaxed one dimensional space. Additionally, we propose two filtering techniques to robustly detect and eliminate false positive matches. We finally propose algorithms to extrapolate missing features and matches on the image borders.

Epipolar Guided Matcher To get an initial estimate of a scene's epipolar geometry, we rely on SIFT features and descriptors [13]. After descriptor matching using fast approx. nearest neighbours (FLANN) [16], we filter outliers by thresholding descriptor distances, discard non-symmetric matches and finally applying the RANSAC algorithm [5], which simultaneously detects outliers and estimates the fundamental matrix. Fig. 2 (top) illustrates the matching pipeline. The so obtained matchings are not suitable for image morphing. We aim for even *distribution* to avoid degenerate meshes, meaningful *coverage* including most to all relevant image parts and *reliability*, as false positive matches highly impair the final morphing quality. In the following, we evaluated various feature detectors and descriptors and found, that Harris corners with SIFT descriptors fit our requirements best. We use these features as basis for guided matching using the

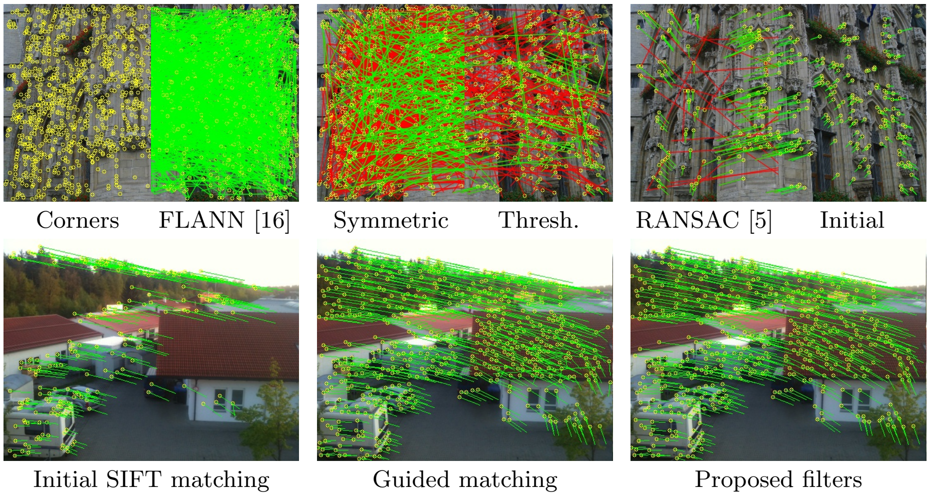


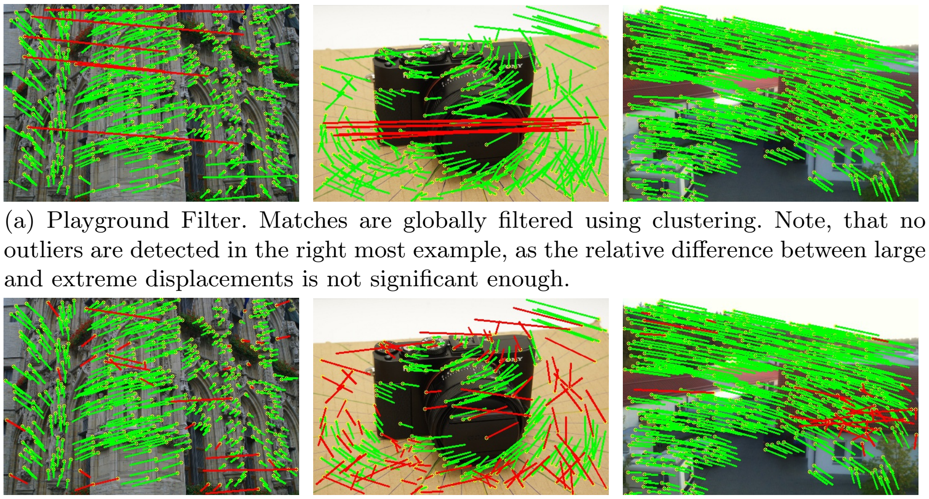
Fig. 2: Our matching pipeline on the Cityhall [22] and KimWest datasets. *Top*: Initial step by step epipolar geometry estimation. *Bottom*: Comparison of SIFT matching, our guided matching and the result after applying our proposed set of outlier filters. Notice the insufficient and poorly distributed keypoints in the initial SIFT matching, whereas our guided matching approach yields good coverage and robust correspondences.

epipolar constraint, effectively reducing the search space beforehand from two dimensions to a relaxed one dimensional space.

We evaluate our approach on different datasets. Fig. 2 (bottom) shows exemplary results on the KimWest dataset. As especially on real world scenes false positive matches occur, we present two novel filtering techniques for robust outlier elimination in the following subsections.

Playground Filter We developed a global filtering procedure, that effectively eliminates obvious false positive matches, i.e. matches with significant displacement deviations. Using k-Means clustering, we group matches into three clusters of keypoint-pairs. They represent small, large and extreme displacements, classifying keypoints based on their relative motion into a foreground, background or an outlier cluster, respectively. To prevent the loss of good matches by discarding a cluster that does not contain actual outliers, we introduce a lower bound for the separation of background and outlier clusters (cf. Fig. 3a). Note that our assumption on scene motion holds especially for camera rotations around a point of interest, but can easily be extended to arbitrary motions by according cluster initialisation. Fig. 3a shows some examples of applied playground filter results.

Council Filter To address outliers that can not be distinguished by global motion models, we propose a novel, local filtering scheme. We compare subsets of matches in small regions around each keypoint, which we call *council*. We employ a voting scheme to rate the quality of a keypoint. A match is accepted if and only



(a) Playground Filter. Matches are globally filtered using clustering. Note, that no outliers are detected in the right most example, as the relative difference between large and extreme displacements is not significant enough.

(b) Council Filter. Matches are locally filtered based on neighbourhood comparison.

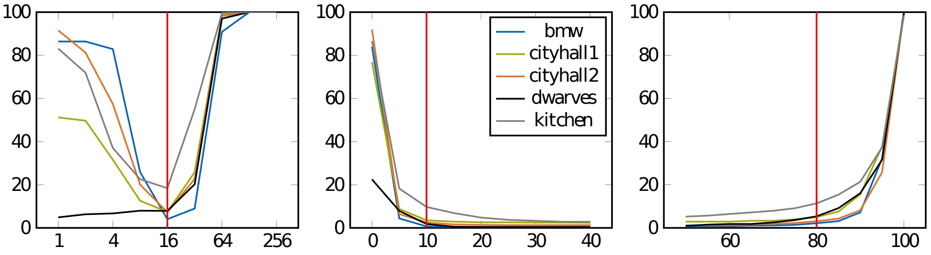
Fig. 3: Outlier elimination using our proposed algorithms on the Cityhall [22], Camera and KimWest datasets. *Green lines*: Inlier matches. *Red lines*: Classified outliers.

if sufficient nearby matches show similar behaviour. This implies consistency and symmetry among matchings. Tab. 1 lists the set of free parameters, by which a council votes.

Table 1: Council Filter Parameters

Variable	Value	Explanation
`maxDistanceRatio`	1/16	neighbourhood radius relative to image width
`maxAngle`	10	maximal angular difference in degree
`minNormRatio`	0.8	minimal relative displacement conformance
`minNeighbours`	3	minimal council size
`minVotes`	2	minimal number of attained votes
`minVoteRatio`	0.15	minimal ratio of attained votes

We evaluate the parameters by comparing the number of outliers relative to the total number of matches. Fixing all but one parameters to initial default values and iteratively adapting the free parameter yields the plots in Fig. 4. For each parameter, we choose the value, for which most outliers are identified while simultaneously the amount of misclassifications is minimised. These are optima in either the first or second derivative of the attained measurements. If the results vary among different scenes, we choose conservatively according to the worst performing example, in order to prevent false positive matches. This might lead to discarding true positive matches in other examples, which we tolerate in



(a) maxDistanceRatio (b) maxAngle (c) minNormRatio

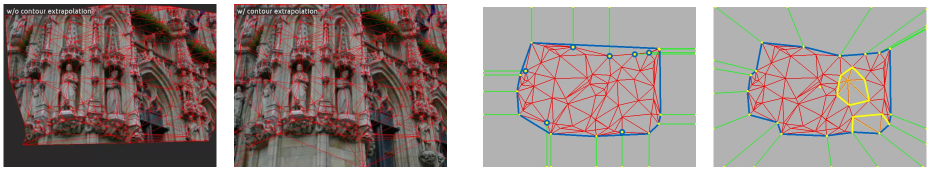
Fig. 4: Exemplary council filter parameter evaluation. The plots show the progression of false positive matches for a single, variable parameter. Red lines mark the selected values. The units on the x-axis are in *relative image width* (a), *degree* (b) and *percent* (c) respectively. *Y-axis*: outlier ratio in *percent*. *X-axis*: free parameter value.

order to prevent unpleasant distortions in the final interpolations. We make use of quadtrees to allow for efficient search queries in keypoint neighbourhoods.

Affine-SIFT SIFT descriptors, which are robust against small viewpoint changes, fail to cover all six parameters of affine transformations. Morel *et al.* [15] present an algorithm that compensates for this lack of dimensionality. We extend their approach by discarding inconclusive matches beforehand using relative descriptor distance thresholding and additionally apply our proposed Playground and Council filters.

ASIFT works well for image pairs with high transition tilts, but are often inaccurate in regions of small movements due to the very dense feature space. Thus, we propose to use ASIFT only as an alternative in those situations in which epipolar guided matching fails to establish reliable matchings. Our framework automatically detects these situations by correspondence quality and quantity evaluation and subsequently uses ASIFT matching instead.

Triangulation and Extrapolation One challenge in sparse image warping is the lack of matches in the boundary regions (cf. Fig. 5a). In the following, we propose a local extrapolation scheme which yields consistent results. New boundary points are attained by projecting outer contour points onto the image boundaries. Let $P \subset \mathbb{R}^2$ be a set of feature points and B be the set of the four image corners. Let further T be the Delaunay triangulation of $P \cup B$. We then introduce the contour of P as the set of points $C \subseteq P$, for which there exists an edge (b, c) in the set of triangles T with $c \in C, b \in B$. In contrast to the convex hull, this definition includes even concave sections (cf. Fig. 5b), which leads to a denser and more accurate extrapolation. We find the contour using neighbourhood cycle closures in undirected, planar graphs. Each of the contour points is then projected onto the image boundary using centric radial projection, yielding an extended set of image boundary points B'. In contrast, orthogonal projection leads to misordered triangle vertices and causes unpleasant artefacts



(a) Triangulation of matches for the City-hall dataset. *Left*: Without border extrapolation. *Right*: Proposed extrapolation.

(b) Match extrapolation on the image border using contour projection. Notice our proposed contour on the right (blue).

Fig. 5: Illustration of the proposed mesh extrapolation.

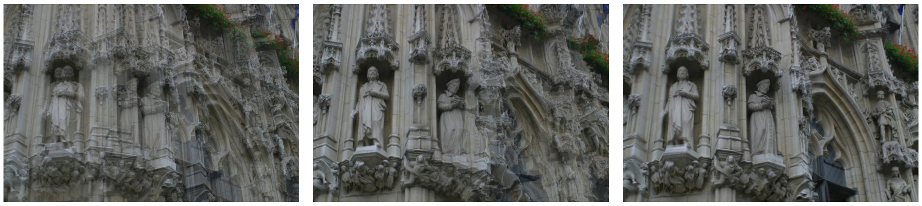
in the final interpolation. The final triangulation is obtained by re-triangulating the set of points $P \cup B'$. In the end, missing displacement information for $B \cup B'$ is attained by assigning each boundary point the motion of its corresponding contour point's match as a local approximation.

2.2 Temporal Interpolation

For temporal interpolation, we incorporate optical flow as presented by Werlberger *et al.* [26] and Chen and Lorenz [2] as a simple and effective tool for generating high quality correspondences between images with small variations (e.g. consecutive video frames or small baseline images). Let $\mathcal{F} : \Omega \subset \mathbb{R}^2 \rightarrow \Omega$ be the flow from source image $\mathcal{I}_0 : \Omega \rightarrow \mathbb{R}^3$ to target image $\mathcal{I}_1 : \Omega \rightarrow \mathbb{R}^3$. A linear dense warping from source to target image is then given by $\mathcal{I}_t(x, y) = \mathcal{I}_1(x + t \cdot \mathcal{F}_x^{-1}(x, y),\ y + t \cdot \mathcal{F}_y^{-1}(x, y))$, $t \in [0, 1]$. We apply this backward-warping to synthesise novel in-between frames. In order to generate spatiotemporal interpolations we first synthesise images in the temporal direction and subsequently perform spatial interpolation as described in the previous section on the temporal in-betweens. For a visual demonstration of interactive results, we refer to the supplemental material, as temporal variations are difficult to show on print.

2.3 Image Warping and Blending

In order to synthesise novel views between two input images $\mathcal{I}_0, \mathcal{I}_1$ at $t \in [0, 1]$, we first warp both images to $\mathcal{I}_0(t)$ and $\mathcal{I}_1(1 - t)$ respectively. In the dense setting, we use pixel-wise backward warping. For sparse matchings, we apply mesh warping and view morphing, where the vertex positions of a textured triangle mesh covering all keypoints are linearly interpolated in rectified image space (cf. [19]). We then blend the intermediate images by adaptive, pixel-wise interpolation, weighted by a logistic sigmoid function $s : [0, 1] \mapsto [0, 1]$ into the blended image $\mathcal{I}(t) = s(1 - t; a) \cdot \mathcal{I}_0(t) + s(t; a) \cdot \mathcal{I}_1(1 - t)$, where $s(x; a) = \frac{h(-2a(x-0.5)) - h(a)}{h(-a) - h(a)}$, with $h(x) = (1 + \exp(x))^{-1}$ and parameter $a \in (0, 1)$ steering the steepness of the sigmoid function. For sparse matchings, we backproject the result using homography interpolation as presented in [6].



Cross-dissolve DualTV-L1 dense flow Proposed sparse approach

Fig. 6: Interpolation on the Cityhall dataset [22] at $t = 0.5$. Our approach yields authentic transformations for such affine camera motions. Notice the poor quality of naive image blending (left) and a flow-based approach (middle), where our sparse approach leads to artefact-free interpolations (right).

3 Results

Implementation We implemented all algorithms in C++ and the GLSL shader language. The software is available for the public on the authors web page[1]. All experiments were run on an Intel Quad-Core i7 3.4 GHz, 16 GB RAM machine with an Intel HD 4000 graphics chip. Establishing and storing dense matches between two images of width 480 px using the TV-L1 dense approach takes about one second. Sparse matches for the same input usually take about half a second. Large speed up factors are expected for a GPU-based implementation. After precomputing all matching data, the actual image interpolation can be explored in real time and Full HD, due to GPU accelerated routines.

Evaluation Our framework can be used for a wide variety of applications, including the synthesis of novel views from small and wide baseline images and videos, the generation of slow-motion effects, virtual avatars, scene exploration

[1] `http://www.tobiasgurdan.de/research/`

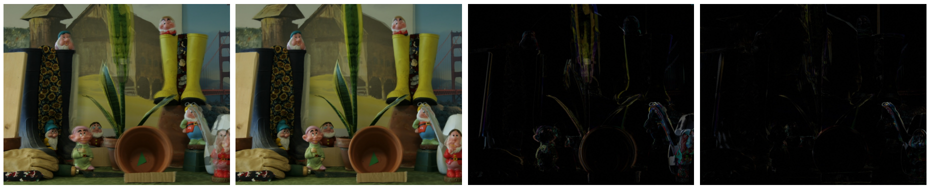
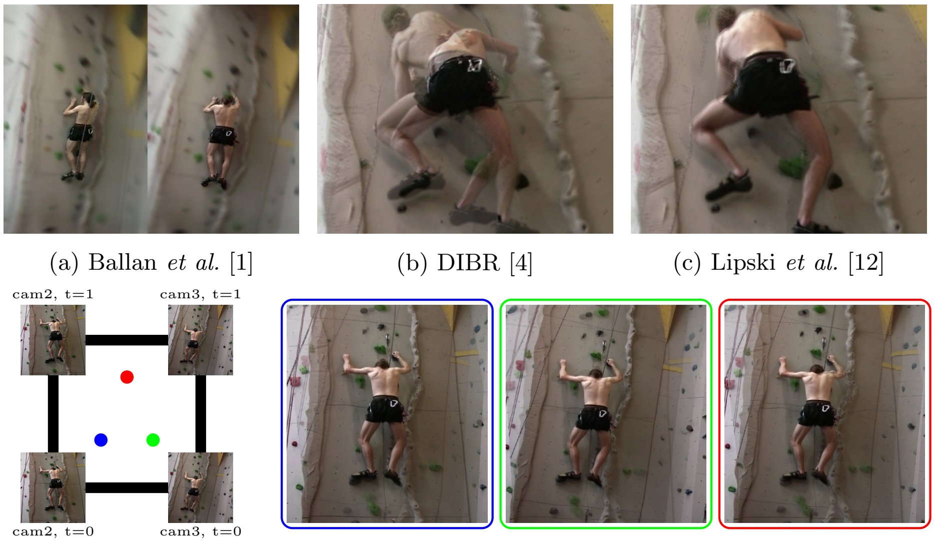


Fig. 7: Comparison of dense and sparse image interpolation on the Middlebury "Dwarves" dataset [18]. *Left*: Dense, TV-L1 flow, $e_{RMSE} = 11.19$, Sec. 2.2. *Right*: Sparse, guided matching, $e_{RMSE} = 6.08$, Sec. 2.1. The left two images show interpolations for one skipped frame, which we take as ground truth. The right two images show difference images of the algorithms' results and the ground truth. Despite the large camera motion, our sparse approach still manages to establish reliable correspondences.


(a) Ballan *et al.* [1]

(b) DIBR [4]

(c) Lipski *et al.* [12]

Fig. 8: *Top*: Comparison to other methods on the Climbing dataset [9]. *Bottom*: Spatiotemporal interpolation using our approach. Notice in (a) the sudden switch of perspective on the climber and the sharpness contrast. (b) shows strong artifacts and ghosting. (c) achieve higher quality results, though there still occur ghosting and warping artifacts. In contrary, notice the high quality of the climber in our approach (bottom). Surrounding artefacts are negligible, as the viewers focus lies on the image centre.

and many more (c.f. supplemental material). We evaluated our approach on several challenging data sets. Fig. 6 shows a qualitative comparison of simple image blending and the methods provided by our framework on a wide-baseline scene (spatial interpolation). Fig. 7 shows a quantitative and qualitative comparison of the flow based approach and our sparse approach on the Middlebury "Dwarves" sequence [18]. We used three consecutive images and interpolated the centre frame from the first and last image and compared the result to the original centre frame as the ground truth by computing the pixel- and channel-wise root mean square error e_{RMSE} for each method. Fig. 7 demonstrates that our sparse approach is better suited for larger camera motions. In Fig. 1 we illustrate that our sparse approach also handles wide-baseline scenarios while still preserving image quality in the interpolated images. Fig. 8 and Fig. 9 give elaborate comparisons to state-of-the-art methods, classifying our approach as a compatible alternative for a wide variety of scenes and use-cases.

Runtime Our framework has low computation times. While Lipski *et al.* need several seconds to compute the correspondence map for an image pair [12], our approach does so in under a second in both spatial and temporal domain. However, our employed subroutines are currently implemented on a single CPU, but well suited for parallelisation on GPUs. Especially, the most computational part,

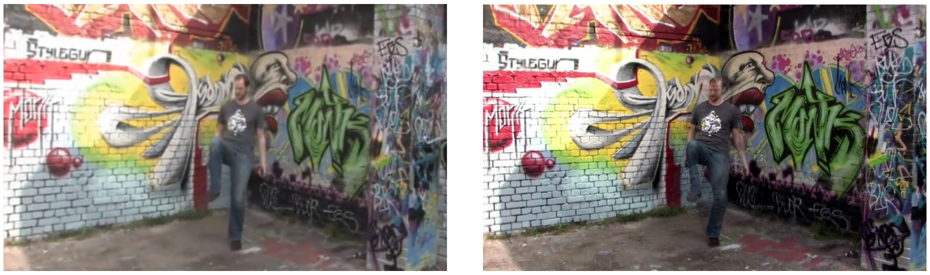


Fig. 9: Comparison of spatiotemporal interpolation on the Graffiti dataset at $s = 0.5, t = 0.5$. *Left*: Lipski *et al.* [12]. *Right*: Our sparse approach. While the juggler shows noticeable artefacts, the wall in the background is perfectly matched and leads to an overall pleasing transition between views.

the SIFT feature extraction, could be parallelised, e.g.[25]. This would further decrease computation time significantly, making the approach real-time capable.

Limitations Although sparse features are robust to large camera movements, with their triangulation we assume a linear distribution of depth values between the features. If the distribution of the feature does not cover depth discontinuities, interpolation artefacts become noticeable as for example can be seen for the person in Fig. 9. In contrast, the piecewise flat background is perfectly interpolated. Similar artefacts are also visible in several frames of the climbing sequence Fig. 8. Moreover, the heuristic interpolation of extrapolated points on the image boundary can lead to distortions, but exact matching information is not available at these locations. Nevertheless, the approach works well in many scenarios and outperforms many related works in terms of speed and simplicity.

4 Conclusion

We presented a framework for spatial and temporal image interpolation. We showed that sparse correspondence based interpolation poses a qualified alternative to state-of-the-art, dense 3d reconstruction or billboard approaches. Establishing robust correspondences among image pairs is still one of the most challenging problems in computer vision. We tackled this problem by introducing a novel matching and outlier filtering pipeline which makes the approach robust even in wide-baseline camera setups. We further proposed an extrapolation method for absent correspondences on image boundaries. For temporal interpolation, a dense correspondence is more suited and can be efficiently solved using established optical flow based methods. We combine the strengths of these approaches in a simple and effective spatiotemporal interpolation method that has a low computation time. We demonstrated competitive results on a variety of sequences with different scene structure and camera baselines.

References

1. Ballan, L., Brostow, G.J., Puwein, J., Pollefeys, M.: Unstructured video-based rendering: interactive exploration of casually captured videos. ACM Trans. Graph. 29(4) (2010), http://dblp.uni-trier.de/db/journals/tog/tog29.html#BallanBPP10
2. Chen, K., Lorenz, D.A.: Image sequence interpolation based on optical flow, segmentation, and optimal control. IEEE Transactions on Image Processing 21(3), 1020–1030 (2012), http://dblp.uni-trier.de/db/journals/tip/tip21.html#ChenL12
3. Debevec, P.: The Campanile Movie. In: SIGGRAPH 97 Electronic Theater (1997), http://www.debevec.org/Campanile/, (visited: May 2014)
4. Fehn, C.: Depth-Image-Based Rendering (DIBR), Compression and Transmission for a New Approach on 3D-TV. In: Proceedings of SPIE Stereoscopic Displays and Virtual Reality Systems XI. pp. 93–104 (2004)
5. Fischler, M., Bolles, R.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24(6), 381–395 (1981)
6. Fragneto, P., Fusiello, A., Rossi, B., Magri, L., Ruffini, M.: Uncalibrated view synthesis with homography interpolation. In: 3DIMPVT. pp. 270–277. IEEE (2012), http://dblp.uni-trier.de/db/conf/3dim/3dimpvt2012.html#FragnetoFRMR12
7. Germann, M., Hornung, A., Keiser, R., Ziegler, R., Würmlin, S., Gross, M.: Articulated billboards for video-based rendering. Comput. Graphics Forum (Proc. Eurographics) 29(2), 585–594 (2010)
8. Goesele, M., Ackermann, J., Fuhrmann, S., Haubold, C., Klowsky, R., Steedly, D., Szeliski, R.: Ambient point clouds for view interpolation. ACM Trans. Graph. 29(4), 95:1–95:6 (Jul 2010), http://doi.acm.org/10.1145/1778765.1778832
9. Hasler, N., Rosenhahn, B., Thormählen, T., Wand, M., Gall, J., Seidel, H.P.: Markerless motion capture with unsynchronized moving cameras. In: CVPR. pp. 224–231 (2009)
10. Inamoto, N., Saito, H.: Free viewpoint video synthesis and presentation from multiple sporting videos. In: ICME. pp. 322–325. IEEE (2005), http://dblp.uni-trier.de/db/conf/icmcs/icme2005.html#InamotoS05
11. Lipski, C.: Virtual Video Camera: a System for Free Viewpoint Video of Arbitrary Dynamic Scenes. Ph.D. thesis, TU Braunschweig (Jun 2013)
12. Lipski, C., Linz, C., Berger, K., Magnor, M.A.: Virtual video camera: image-based viewpoint navigation through space and time. In: SIGGRAPH Posters. ACM (2009), http://dblp.uni-trier.de/db/conf/siggraph/siggraph2009posters.html#LipskiLBM09
13. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. In: International Journal of Computer Vision. vol. 20 (2003)
14. Mahajan, D., Huang, F.C., Matusik, W., Ramamoorthi, R., Belhumeur, P.N.: Moving gradients: a path-based method for plausible image interpolation. ACM Trans. Graph. 28(3) (2009)
15. Morel, J.M., Yu, G.: Asift: A new framework for fully affine invariant image comparison. SIAM J. Imaging Sciences 2(2), 438–469 (2009), http://dblp.uni-trier.de/db/journals/siamis/siamis2.html#MorelY09
16. Muja, M., Lowe, D.G.: Fast approximate nearest neighbors with automatic algorithm configuration. In: Ranchordas, A., Arajo, H. (eds.) VISAPP (1). pp. 331–340. INSTICC Press (2009), http://dblp.uni-trier.de/db/conf/visapp/visapp2009-1.html#MujaL09

17. Replay Technologies Inc.: freeD^{TM} technology (2013), http://replay-technologies.com/, (visited: May 2014)
18. Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: CVPR (2007)
19. Seitz, S.M., Dyer, C.R.: Physically-valid view synthesis by image interpolation. In: In Proc. IEEE Workshop on Representations of Visual Scenes. pp. 18–25 (1995)
20. Seitz, S.M., Dyer, C.R.: View morphing. In: SIGGRAPH. pp. 21–30 (1996), http://dblp.uni-trier.de/db/conf/siggraph/siggraph1996.html#SeitzD96
21. Snavely, N., Garg, R., Seitz, S.M., Szeliski, R.: Finding paths through the world's photos. ACM Transactions on Graphics (Proceedings of SIGGRAPH 2008) 27(3), 11–21 (2008)
22. Strecha, C., Tuytelaars, T., Gool, L.J.V.: Dense matching of multiple wide-baseline views. In: ICCV. pp. 1194–1201 (2003)
23. Vedula, S., Baker, S., Kanade, T.: Image-based spatio-temporal modeling and view interpolation of dynamic events. ACM Trans. Graph. 24(2), 240–261 (2005)
24. Vlad, A.: Image morphing techniques. JIDEG 5(1) (JUNE 2010), http://www.sorging.ro/ro/member/serveFile/format/pdf/slug/image-morphing-techniques
25. Warn, S., Apon, A., Cothren, J.: Accelerating sift on hybrid clusters. In: Proceedings of the ACM SIGSPATIAL Second International Workshop on High Performance and Distributed Geographic Information Systems. pp. 2–9. HPDGIS '11, ACM, New York, NY, USA (2011), http://doi.acm.org/10.1145/2070770.2070771
26. Werlberger, M., Pock, T., Unger, M., Bischof, H.: Optical flow guided tv-l1 video interpolation and restoration. In: Energy Minimization Methods in Computer Vision and Pattern Recognition (2011)
27. Wolberg, G.: Image morphing : A survey. Visual Computer 14, 360–372 (1998), http://ci.nii.ac.jp/naid/80010827845/en/
28. Zhang, Z., Deriche, R., Faugeras, O.D., Luong, Q.T.: A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. Artif. Intell. 78(1-2), 87–119 (1995), http://dblp.uni-trier.de/db/journals/ai/ai78.html#ZhangDFL95
29. Zitnick, C.L., Kang, S.B., Uyttendaele, M., Winder, S.A.J., Szeliski, R.: High-quality video view interpolation using a layered representation. ACM Trans. Graph. 23(3), 600–608 (2004), http://dblp.uni-trier.de/db/journals/tog/tog23.html#ZitnickKUWS04