Spatiotemporal Descriptor for Wide-Baseline Stereo Reconstruction of Non-Rigid and Ambiguous Scenes

Eduard Trulls, Alberto Sanfeliu, and Francesc Moreno-Noguer

Institut de Robòtica i Informàtica Industrial (CSIC/UPC)
C/ Llorens i Artigas 4-6, 08028 Barcelona, Spain
{etrulls, sanfeliu, fmoreno}@iri.upc.edu

Abstract. This paper studies the use of temporal consistency to match appearance descriptors and handle complex ambiguities when computing dynamic depth maps from stereo. Previous attempts have designed 3D descriptors over the spacetime volume and have been mostly used for monocular action recognition, as they cannot deal with perspective changes. Our approach is based on a state-of-the-art 2D dense appearance descriptor which we extend in time by means of optical flow priors, and can be applied to wide-baseline stereo setups. The basic idea behind our approach is to capture the changes around a feature point in time instead of trying to describe the spatiotemporal volume. We demonstrate its effectiveness on very ambiguous synthetic video sequences with ground truth data, as well as real sequences.

Key words: stereo, spatiotemporal, appearance descriptors

1 Introduction

Appearance descriptors have been used to successfully address both low- and high-level computer vision problems by describing in a succinct and informative manner the neighborhood around an image point. They have proved robust in many applications such as 3D reconstruction, object classification or gesture recognition, and remain a major topic of research in computer vision.

Stereo reconstruction is often performed matching descriptors from two or more different images to determine the quality of each correspondence, and applying a global optimization scheme to enforce spatial consistency. This approach fails on scenes with poor texture or repetitive patterns, regardless of the descriptor or the underlying optimization scheme. In these situations we can attempt to solve the correspondence problem by incorporating dynamic information. Many descriptors are based on histograms of gradient orientations [1–3], which can be extended to the spacetime volume. Since the local temporal structure depends strongly on the camera view, most efforts in this direction have focused on monocular action recognition [4–7]. For stereo reconstruction, spatiotemporal descriptors computed in this manner should be oriented according

Work supported by the Spanish Ministry of Science and Innovation under projects RobTaskCoop (DPI2010-17112), PAU+ (DPI2011-27510) and MIPRCV (Consolider-Ingenio 2010)(CSD2007-00018), and the EU ARCAS Project FP7-ICT-2011-287617. E. Trulls is supported by scholarship from Universitat Politècnica de Catalunya.

to the geometry of the setup. This approach was applied in [8] to disparity estimation for narrow-baseline stereo, but its application to wide-baseline scenarios remains unexplored.

We face the problem from a different angle. Our main contribution is a spatiotemporal approach to 3D stereo reconstruction applicable to wide-baseline stereo, augmenting the descriptor with optical flow priors instead of computing 3D gradients or similar primitives. We compute, for each camera, dense Daisy descriptors [3] for a frame over the spatial domain, and then extend them over time with optical flow priors while discarding incorrect matches, effectively obtaining a concatenation of spatial descriptors for every pixel. After matching, global optimization algorithm is applied over the spatiotemporal domain to enforce spatial and temporal consistency. The core idea is to develop a primitive that captures the evolution of the spatial structure around a feature point in time, instead of trying to describe the spatiotemporal volume, which for matching requires a knowledge of the geometry of the scene. We apply this approach to dynamic sequences of non-rigid objects with a high number of ambiguous correspondences and evaluate it on both synthetic and real sequences. We show that its reconstructions are more accurate and stable than those obtained with state-of-the-art descriptors. In addition, we demonstrate that our approach can be applied to wide-baseline setups with occlusions, and that it performs very strongly against image noise.

2 Related Work

The main reference among keypoint descriptors is still the SIFT descriptor [1], which has shown great resilience against affine deformations on both the spatial and intensity domains. Given its computational cost, subsequent work has often focused on developing more efficient descriptors, such as PCA-SIFT [9], GLOH [2] or SURF [10], and recent efforts such as Daisy veer towards dense computation [3]—i.e. computing a descriptor for every pixel. Open problems include the treatment of non-rigid deformations, scale, and occlusions. Current techniques accommodate scale changes by limiting their application to singular points [1, 11, 10], where scale can be reliably estimated. A recent approach [12] exploits a log-polar transformation to achieve scale invariance without detection. Non-rigid deformations have been seldom addressed with region descriptors, but recent advances have shown that kernels based on heat diffusion geometry can effectively describe local features of deforming surfaces [13]. Regarding occlusions, [3] demonstrated performance improvements in multi-view stereo from the treatment of occlusion as a latent variable and enforcing spatial consistency with graph cuts [14].

Although regularization schemes such as graph cuts may improve the spatial consistency when matching pairs of images, in scenes with little texture or highly repetitive patterns the problem can be too challenging. Dynamic information can then be used to further discriminate amongst possible matches. In controlled settings, the correspondence problem can be further relaxed via structured-light patterns [15, 16]. Otherwise, more sophisticated descriptors need to be designed. For this purpose, SIFT and its many variants have been extended to 3D data, although mostly for monocular cameras, as the local temporal structure varies strongly with large viewpoint changes. For instance, this has been applied to volumetric images on clinical data, and to video sequences for action recognition [6, 5, 7, 17]. There are exceptions in which spatiotemporal descriptors have also been applied to disparity estimation for narrow-baseline stereo [8], designing primitives (Stequels) that can be reoriented and matched from slightly different viewpoints. The application of this approach to wide-baseline scenarios remains unexplored.

Our approach is also related to scene flow—i.e. the simultaneous recovery of 3D flow and geometry. We have not attempted to compute scene flow, as these approaches usually require strong assumptions such as known reflectance models or relatively small deformations to simplify the problem, and often use simple pixel-wise matching strategies that cannot be applied to ambiguous scenes or wide baselines. A reflectance model is exploited in [18] to estimate the scene flow under known illumination conditions. Zhang and Kambhamettu [19] estimate an initial disparity map and compute the scene flow iteratively, exploiting image segmentation to maintain discontinuities. A more recent approach is that of [20], which couples dense stereo matching with optical flow estimation—this involves a set of partial differential equations which are solved numerically. Of particular interest to us is the work of [21], which decouples stereo from motion while enforcing a scene flow consistent across different views—this approach is agnostic to the algorithm used for stereo.

3 Spatiotemporal Descriptor

Our approach could be applied, in principle, to any spatial appearance descriptor described in the Section 2. We pick Daisy [3] for the following reasons: (1) it is designed for dense computation, which has been proven useful for wide-baseline stereo, and will particularly help in ambiguous scenes without differentiated salient points; (2) it is efficient to compute, and more so for our purposes, as we can store the convolved orientation maps for the frames that make up a descriptor without recomputing them; and (3) unlike similar descriptors like SIFT [1] or GLOH [2], Daisy is computed over a discrete grid, which can be warped in time to follow the evolution of the structure around a point, allowing us to validate the optical flow priors and determine the stability of the descriptor in time. The last point will help us particularly when dealing with occlusions. This section describes the computation of the descriptor set for a single camera, and the following section explains how to match descriptor sets for stereo reconstruction.

We extend the Daisy descriptor in the following way. For every new (gray-scale) frame I_k, we compute the optical flow priors for each consecutive pair of frames from the same camera, in both the forward and backward directions: F^{+}_{k-1} (from I_{k-1} to I_k) and F^{-}_{k} (viceversa). To compute a full set of spatiotemporal descriptors for frame k using $T = 2 \cdot B + 1$ frames we require frames I_{k-B} to I_{k+B} and their respective flow priors. To compute the flow priors we use [22]. Since the size of the descriptor grows linearly with T we use small values, of about 3 to 11.

We then compute, for each frame, H gradient maps, defined as the gradient norms at each location if they are greater than zero, to preserve polarity. Each orientation map is convolved with multiple gaussian kernels of different values to obtain the convolved orientation maps, so that the size of the kernel determines the size of the region. The gaussian filters are separable, and the convolution for larger kernels is obtained from consecutive convolutions with smaller kernels, for increased performance. The result is a series of convolved orientation maps, which contain a weighted sum of gradient norms around each pixel. Each map describes regions of a different size for a given orientation. A Daisy descriptor for a given feature point would then be computed as a concatenation of these values over certain pixel coordinates defined by a grid. The grid is defined as equally spaced points over equally spaced concentric circles around the feature point coordinates (see Fig. 1). The number of circles Q and the number of grid points on each circle P, along with the number of gradient maps H, determine the size of the descriptor, $S = (1 + P \cdot Q) \cdot H$. The number of circles also determines the number of kernels to compute, as outer rings use convolution maps computed over larger regions for some invariance against rotation. The histograms are normalized separately for each grid point, for robustness against partial occlusions. We can compute a score for the match between a pair of Daisy descriptors as:

$$D(\mathbf{D_1}(\mathbf{x}), \mathbf{D_2}(\mathbf{x})) = \frac{1}{S} \sum_{g=1}^{S} \|\mathbf{D_1^{[g]}}(\mathbf{x}) - \mathbf{D_2^{[g]}}(\mathbf{x})\|, \tag{1}$$

where $\mathbf{D_i^{[g]}}$ denotes the histogram at grid point $g = 1 \ldots G$. To enforce spatial coherence in dealing with occlusions we apply binary masks over each grid point, obscuring parts of its neighborhood and thus removing them from the matching process, following [3]. The choice of masks will be explained in the following section. The matching score (cost) can then be computed in the following way:

$$D' = \frac{1}{\sum_{q=1}^{S} \mathcal{M}^{[q]}} \sum_{g=1}^{S} \mathcal{M}^{[g]} \|\mathbf{D_1^{[g]}}(\mathbf{x}) - \mathbf{D_2^{[g]}}(\mathbf{x})\|, \tag{2}$$

where $\mathcal{M}^{[q]}$ is the mask for grid point g. Note that we do not use masks to build the spatiotemporal descriptor, only to match two spatiotemporal descriptors for stereo reconstruction.

For our descriptor, we use the grid defined above to compute the sub-descriptor over the feature point on the central frame. We then warp the grid through time by means of the optical flow priors, translating each grid point independently. In addition to warping the grid, we average the angular displacement of each grid point over the center of the grid to estimate the change in rotation over the patch between the frames, and compute a new sub-descriptor using the warped grid and the new orientation.

An example of this procedure is depicted in Fig. 1. We then match each sub-descriptor computed with the warped grid against the sub-descriptor for the central frame, and discard it if the matching score falls above a certain threshold. The matching score is typically smaller than the occlusion cost in the regularization process (see section 4). With this procedure we can validate the optical flow and discard a match if the patch suffers significant transformations such as large distortions, lighting changes or, in particular, partial occlusions. Note that we do not need to recompute the convolved orientation maps for each frame, as we can store them in memory. Computing the descriptors is then reduced to sampling the convolved orientation maps at the appropriate grid points on each frame. To compute descriptors over different orientations we simply rotate the grid and shift the histograms circularly. We also apply interpolation over both

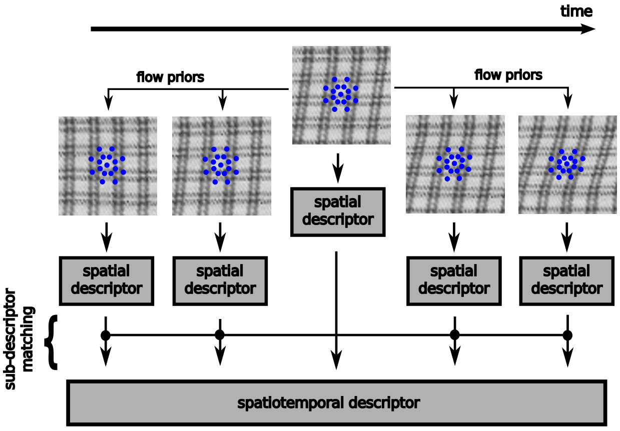

Fig. 1. Computation of the spatiotemporal descriptor with flow priors. Sub-descriptors outside the central frame are computed with a warped grid and matched against the sub-descriptor for the central frame. Valid sub-descriptors are then concatenated to create the spatiotemporal descriptor.

the spatial domain and the gradient orientations. The resulting descriptor is assembled by concatenating the sub-descriptors in time $\widetilde{\mathbf{D}}(\mathbf{x}) = \{D^{k-B} \ldots D^{k+B}\}$, and its size is at most $S' = T \cdot S$, where S is the size of a single Daisy descriptor.

To compute the distance between two spatiotemporal descriptors $\mathbf{D_1^k}$ we average the distance between valid pairs of sub-descriptors:

$$\widetilde{D} = \frac{1}{V} \sum_{l=k-B}^{k+B} v_l \cdot D'(\mathbf{D_1^{\{l\}}}(\mathbf{x}), \mathbf{D_2^{\{l\}}}(\mathbf{x})) \ , \tag{3}$$

where $\mathbf{D_i^{\{l\}}}$ is the sub-descriptor for frame l, v_l are a set of binary flags that determine valid matching sub-descriptor pairs (where both sub-descriptors pass the validation process), and $V = \sum_{l=k-B}^{k+B} v_l$. Note that in a worst-case scenario we will always match the sub-descriptors for frame k.

4 Depth Estimation

For stereo reconstruction we use a pair of calibrated monocular cameras. Since Daisy is not rotation-invariant we require the calibration data to compute the descriptors along the epipolar lines, rotating the grid. We do this operation for the central frame, and for frames forwards and backwards we use the flow priors to warp the grid and to estimate the patch rotation between the frames. We discretize the 3D space from a given perspective, then compute the depth for every possible match of descriptors and store it if the match is the best for a given depth bin, building a cube of matching scores of size $W \times H \times L$, where W and H are the width and height of the image and L is the number of layers we use to discretize 3D space. We can think of these values as costs, and then apply global optimization techniques such as graph cuts [14] to enforce piecewise smoothness, computing a good estimate that balances the matching costs with a smoothness cost that penalizes the use of different labels on neighboring pixels. To deal with occlusions we incorporate an occlusion node in the graph structure with a constant cost. We use the same value, 20% of the maximum cost, for all experiments.

We can use two strategies for global optimization: enforcing spatial consistency or enforcing both spatial and temporal consistency. To enforce *spatial consistency*, we can match two sets of spatiotemporal descriptors and run the optimization algorithm for a single frame. The smoothness function for the optimization process will consider the 4 pixels adjacent to the feature point, and its cost is binary: a constant penalty for differing labels, and 0 for matching labels.

To enforce *spatial and temporal consistency*, we match two sets of multiple spatiotemporal descriptors. To do this we use a hypercube of matching costs of size $W \times H \times L \times M$, where M is the number of frames used in the optimization process. Note that M does not need to match T, the number of frames used for the computation of the descriptors. We then set 6 neighboring pixels for the smoothing function, so that each pixel (x, y, t) is linked to its 4 adjacent neighbors over the spatial domain and the two pixels that share its spatial coordinates on two adjacent frames, $(x, y, t-1)$ and $(x, y, t+1)$. We then run the optimization algorithm over the spatiotemporal volume and obtain M separate depth maps. The value for M is constrained by the amount of memory available, and we use $M = 5$ for all the experiments in this paper.

This strategy produces better, more stable results on dynamic scenes, but introduces one issue. The smoothness function operates over both the spatial and the temporal domain in the same manner, but we have a much shorter buffer in the temporal domain (M, versus the image size). This results in over-penalizing label discontinuities over the time domain, and in practice smooth gradients on the temporal dimension are often lost in favor of two frames with the same depth values. Additionally, the frames at either end of the spatiotemporal volume are linked to a single frame, as opposed to two, and the problem becomes more noticeable. For 2D stereo we apply the Potts model, $F(\alpha, \beta) = S \cdot T(\alpha \neq \beta)$, where α and β are discretized depth values for adjacent pixels, S is the smoothness cost, and $T(\cdot)$ is 1 if the argument is true and 0 otherwise. For spatiotemporal stereo we use instead a truncated linear model:

$$F'(\alpha, \beta) = \begin{cases} \frac{S \cdot |\alpha - \beta|}{L} & \text{if } |\alpha - \beta| \leq L \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Notice that $F = F'$ for $L = 1$. This effectively relaxes the smoothness constraint, but the end results are better due to the integration of spatiotemporal hypotheses. The frames at either end of the spatiotemporal buffer, which are linked to one single neighbor, can still present this behaviour. We solve this problem discarding their depth maps, which are reestimated in the following iteration. Note that we do not need to recompute the cost cubes for the discarded frames: we just store the two cubes on the end of the buffer, which become the first ones on the next buffer, and then compute the following $M - 2$ cubes ($M \geq 3$). This procedure is depicted in Fig. 2. This strategy introduces some overhead in the regularization process, but its computational cost is negligible

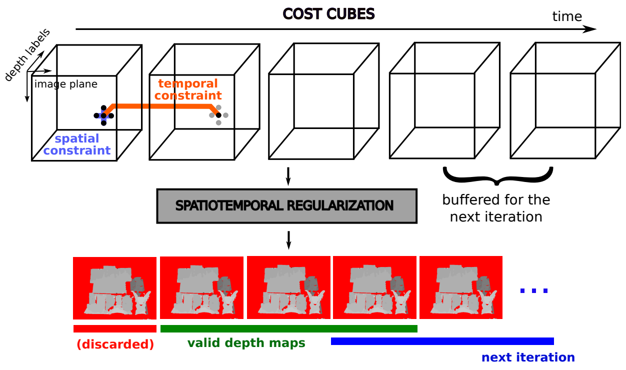

Fig. 2. To perform spatiotemporal regularization, we apply a graph cuts algorithm over the spatiotemporal hypercube of costs (here displayed as a series of cost cubes). The smoothness function connects pixels on a spatial and temporal neighborhood. To avoid singularities on the frames at either end of the buffer, we discard their resulting depth maps and obtain them from the next iteration, as pictured. We do not need to recompute the cost cubes.

with respect to the cost of matching dense descriptors. We use the spatiotemporal optimization strategy for most of the experiments on this paper. We refer to each strategy as GC-2D and GC-3D.

4.1 Masks for occlusions

To deal with occlusions we formalize binary masks over the descriptor, as in [3].We use a set of $P + 1$ binary masks, where P is the number of grid points on each ring. To enforce spatial coherence the masks are predefined as follows: one mask enables all grid points, and the rest enable the grid points in a half moon and disable the rest, for different orientations of the half moon (see Fig. 3). After an initial iteration, we use the first depth estimation to compute a score for each mask, using the formulation proposed by [3], which penalizes differing depth values on enabled areas. We then compute a new depth estimation, using for each pixel the mask with the highest score computed from the previous depth map. Two or three iterations are enough to refine the areas around occlusions.

As we explained in Section 1 our goal is to design a spatiotemporal descriptor that can be applied to wide-baseline scenarios without prior knowledge of their geometry, in order to deal with highly ambiguous video sequences. As such we are not focused on dealing with occlusions, and in fact do not use masks for most experiments in this paper. Masks only help around occlusions, and will be counter-productive if the first depth estimation contains errors due to ambiguous matches, as they will incorrectly reject valid data in the following iterations. Nevertheless, we extend this approach to our descriptor and apply it to a standard narrow-baseline video sequence with ground truth data (see Section 5.4), to demonstrate its viability for spatiotemporal stereo. To apply a mask over the spatio-temporal descriptor, we apply it separately to each of its sub-descriptors for

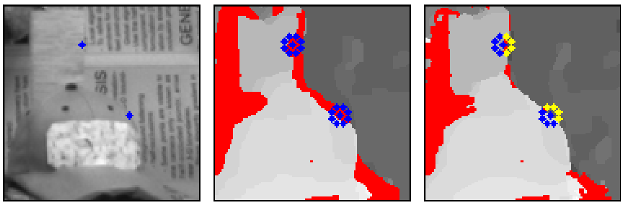

Fig. 3. Application of masks to scenes with occlusions. Left to right: reference image, a depth map after one iteration, and a posterior refinement using masks. Enabled grid points are shown in blue, and disabled grid points are drawn in black. We show a simplified grid of 9 points, for simplicity.

different frames. For simplicity, we use a GC-2D optimization for the first iterations, and a GC-3D optimization on the last iteration. The masks are recomputed at each step. For reference, a round of spatiotemporal stereo for two 480x360 images takes about 24 minutes on a 2 Ghz dual-core computer (without parallelization), up from about 11 minutes for Daisy. This does not include optical flow estimation (approximately 1 minute per image), for which we use a Matlab off-the-shelf implementation—the rest of the code is C++. The bottleneck is on descriptor matching—note that for these experiments we allow matches along most of the epipolar lines.

5 Experiments and results

Wide-baseline datasets are scarce, and we know of none for dynamic scenes, so we create a series of synthetic sequences with ground truth depth data to evaluate our descriptor. We compute a dense mesh over 3D space, texture it with a repetitive pattern, and use it to generate images and ground truth depth maps from different camera poses. We generate three datasets: (1) A dataset with 5×5 different mesh deformations: sinusoidal oscillations, and random noise over the mesh coordinates. (2) A wide-baseline dataset, with camera poses along five angular increments of $\pi/16$ rad (for a largest baseline of $\pi/4$). And (3) a narrow-baseline dataset with gaussian image noise.

We use the first dataset to gain an intuitive understanding of what kind of dynamic information our descriptor can take advantage of. The second and third datasets are used to benchmark our approach against state-of-the-art descriptors: SIFT [1], Daisy [3] and the spatiotemporal Stequel descriptor [8]. Additionally, we demonstrate the application of masks on a real dataset with occlusions. We also apply it to highly ambiguous real sequences without ground truth data.

Each of our synthetic datasets contains 30 small (480×480) images from each camera pose. All the experiments are performed over the 15 middle frames, and the spatiotemporal descriptors use the additional frames as required. We refer to our descriptor as Daisy-3D, and to the spatial Daisy descriptor as Daisy-2D. If necessary, we use Daisy-3D-2D and Daisy-3D-3D to distinguish between the spatial and spatiotemporal regularization schemes presented in Section 4. For Daisy-2D we use the configuration suggested in [3], with a grid radius of 15 pixels, 25 grid points and 8 gradient orientations, resulting in a descriptor size of 200. For Daisy-3D we use smaller spatial sub-descriptors, concatenated in time, e.g. 136×7 means (at most) 7 sub-descriptors of size 136. For Daisy-2D and SIFT, we apply a graph-cuts spatial regularization scheme. To provide a fair comparison with SIFT we do not use feature points, but instead compute the descriptors densely at a constant scale, over a patch rotated along the epipolar lines, as we do for Daisy-2D and -3D. For the stequel algorithm we rectify the images with [23]. We do not compute nor match descriptors for background pixels. We choose a wide scene range, as the actual range of our synthetic scenes is very narrow and the correspondence problem would be too simple. To evaluate and plot the results, we prune them to the actual scene range (i.e. white or black pixels in the depth map plots are outside the actual scene range). We follow this approach for every experiment except for those on occlusions (5.4), which already feature very richly textured scenes.

5.1 Parameter selection

A preliminary study with the synthetic datasets shows that, as expected, noise on the mesh coordinates is very discriminating, for spatial descriptors but more so for spatiotemporal descriptors. Affine and non-rigid transformations are also informative, but to a lesser degree. The optimal number of frames used to build the descriptor is often small, between 5 and 9. Larger values require a smaller footprint for the spatial descriptor, which reduces the overall performance. For the Daisy parameters, we prefer to reduce the number of histograms (grid points) rather than bins (oriented gradients), as the latter prove more discriminant. For Daisy-3D we use descriptors of 17 histograms of 8 bins (size 136) and up to 7 frames in time for most of the experiments in this paper.

5.2 Wide baseline experiments

For this experiment we compare our descriptor with Daisy-2D, SIFT and Stequel. We match video sequences of the same mesh from five different camera positions, each rotated $\pi/16$ with respect to the mesh (up to $\pi/4$), computing depth maps from the rightmost viewpoint. Figs. 4 and 6 show the results for two video sequences using different image patterns. One contains a highly textured scene, while the other shows a sparse, repetitive pattern. On the configuration with the narrowest baseline Stequel performs better than any of the other algorithms, due to sub-pixel accuracy from a Lucas-Kanade-like refinement [8], while the rest of the algorithms suffer from the discretization process in our regularization framework. Stequel still performs well on the following setup, despite slight systematic errors. For the wide baseline setups the other algorithms perform much better in comparison, with the Daisy-based descriptors edging out SIFT. In fact, large baselines prove very informative without occlusions. Note that we do not apply masks to this experiment, since the scene contains no occlusions.

5.3 Image noise experiments

For these experiments we have 11 video sequences with different levels of image noise. Despite the distortion over the flow estimates, our descriptor significantly outperforms the others (see Figs. 5 and 7). For this experiment we also run our spatiotemporal descriptor with spatial regularization (Daisy-3D-2D) and with spatiotemporal regularization (Daisy-3D-3D), to demonstrate that the descriptor itself is robust to image noise. The Stequel descriptor proves very vulnerable to noise, and SIFT performs slightly better than Daisy.

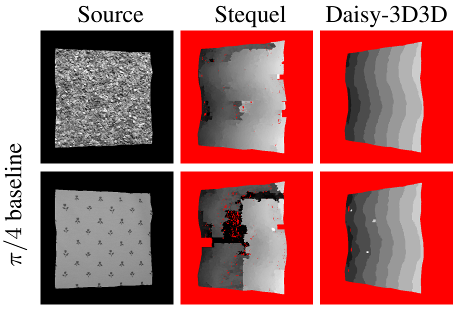

Fig. 4. Baseline examples.

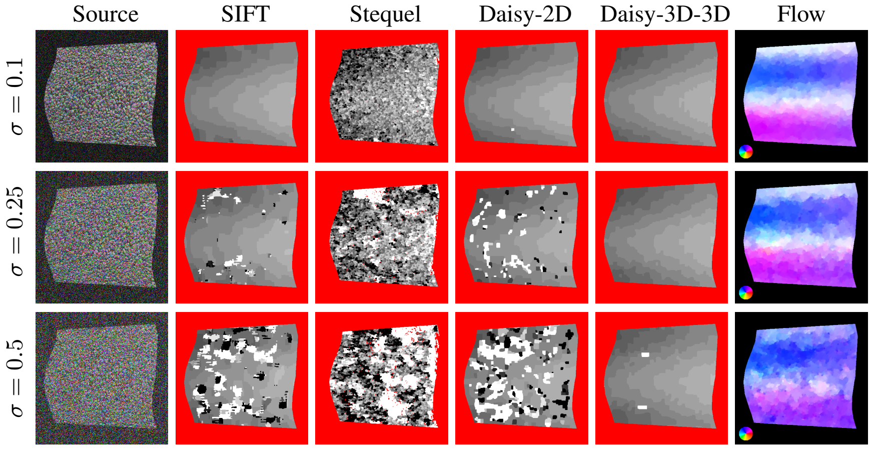

Fig. 5. Samples from the image noise experiment.

5.4 Experiments with masks

To assess the behavior of our descriptor against occlusions we have used a narrow-baseline stereo dataset from [8]. It contains a very richly textured scene, and we use a very small descriptor, with 9 grid points and 4 gradients (size 36). Note that Daisy-2D stereo already performs very well on these settings, and we do not expect significant improvements with the spatiotemporal approach. Instead, we use it to demonstrate that

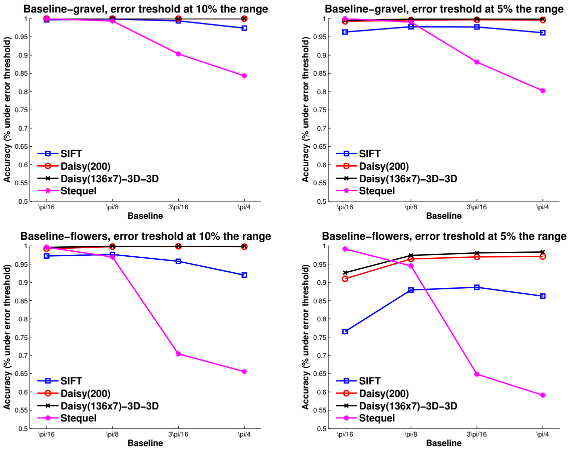

Fig. 6. Baseline results.

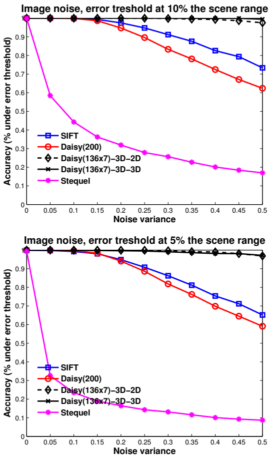

Fig. 7. Image noise results.

propagating sub-descriptors in time around occlusion areas does not impair the reconstruction process if they are validated with the mechanism presented in Section 3. Fig. 8 shows that the reconstruction does not change significantly as the number of frames used to build the spatiotemporal descriptor increases. This is to be expected, as the scene contains few ambiguities and dynamic content. But we also observe that using a high number of frames does not result in singularities around occlusions.

5.5 Experiments with real sequences

In addition to the synthetic sequences, we apply the algorithm to highly ambiguous real sequences. We do so qualitatively, as we cannot compute ground truth data at high rates through traditional means: structured light would alter the scene, most 3D laser range finders are not fast enough, and available time-of-flight cameras or motion sensor devices such as the Kinect are not accurate enough to capture small variations in depth. We require high rates to extract an accurate flow, which would otherwise fail on sequences of this complexity. In fact for this experiment we compute the optical flow at a higher frame-rate, approximately 100 Hz. We then interpolate the flow data and run the stereo algorithm at a third of that rate. We do so because even though we have demonstrated that our spatiotemporal descriptor is very robust against perturbations of the flow data (see Section 5.3), video sequences of this nature may suffer from the aperture problem—we believe that in practice this constraint can be relaxed. Even without ground truth data, simple visual inspection shows that the depth estimate is generally much more stable in time and follows the expected motion (see Fig. 9).

6 Conclusions and future work

We have proposed a solution to stereo reconstruction that can handle very challenging situations, including highly repetitive patterns, noisy images, occlusions, non-rigid

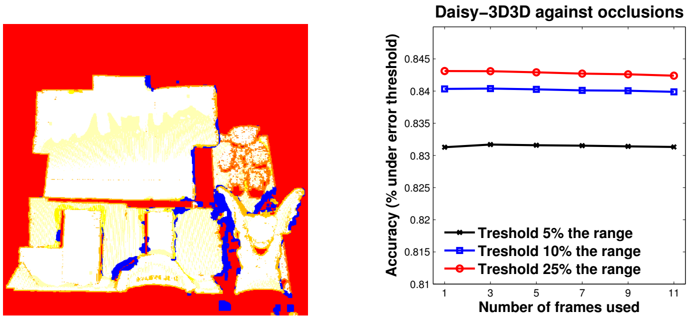

Fig. 8. Benchmarking the spatiotemporal descriptor against occlusions. The image displays the depth error after three iterations with masks, using the Daisy-3D3D approach with 11 frames. The error is expressed in terms of layers: white indicates a correct estimation, light yellow indicates an error of one layer, which is often due to quantization errors, and further deviations are drawn in yellow to red. Red also indicates occlusions. Blue indicates a foreground pixel incorrectly labeled as an occlusion. The plot shows accuracy values at different error thresholds for an increasing number of frames used to compute the spatiotemporal descriptor.

deformations, and large baselines. We have shown that these artifacts can only be addressed by appropriately exploiting temporal information. Even so, we have not attempted to describe cubic-shaped volumes of data in space and time, as is typically done in current spatiotemporal stereo methods. In contrast, we have used a state-of-the-art appearance descriptor to represent all pixels and their neighborhoods at a specific moment in time, and have propagated this representation through time using optical flow information. The resulting descriptor is able to capture the non-linear dynamics that individual pixels undergo over time. Our approach is very robust against optical flow noise and other artifacts such as mislabeling due to aperture noise—the spatiotemporal regularization can cope with these errors. Additionally, objects that suddenly pop on-screen or show parts that were occluded will be considered for matching across frames where they may not exist—we address this problem matching warped descriptors across time.

One of the main limitations of the current descriptor is its high computational cost and dimensionality. Thus, as future work, we will investigate techniques such as vector quantization or dimensionality reduction to compact our representation. In addition, we believe that monocular approaches to non-rigid 3D reconstruction and action recognition [24, 25] may benefit from our methodology, specially when the scene or the person is observed from very different points of view. We will also research along these lines.

We would also like to investigate the application of our approach to scene flow estimation. We currently do not check the flow estimates for consistency—note that we perform alternative tests which are also effective: we do check the warped descriptors for temporal consistency and discard them when they match poorly, and we apply a global regularization to enforce spatiotemporal consistency. Given our computational concerns, we could decouple stereo from motion as in [21], while enforcing joint consistency.

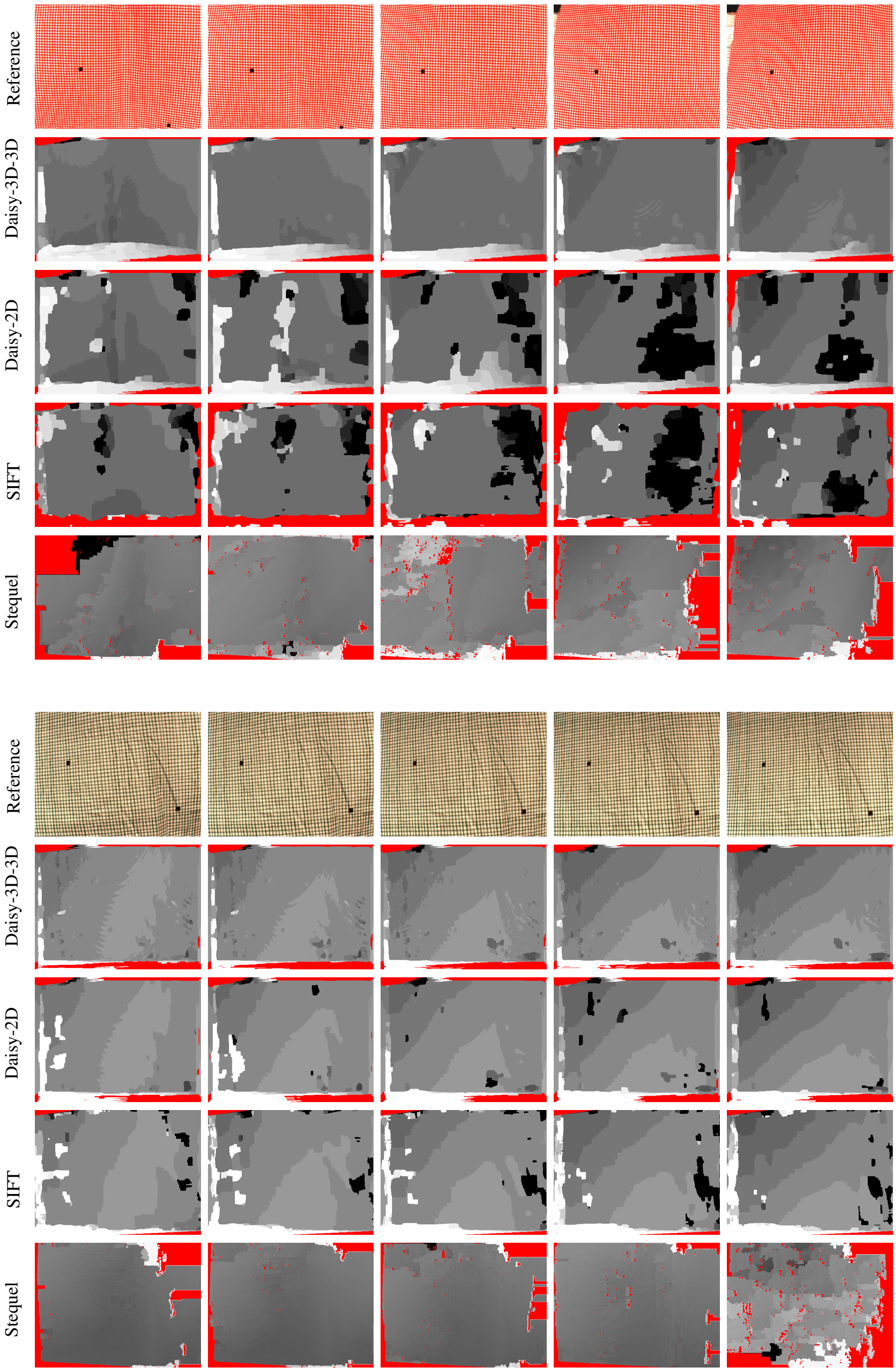

Fig. 9. Depth reconstruction for five consecutive frames of two very challenging stereo sequences.

References

1. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV (2004)
2. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. T.PAMI **27** (2005)
3. Tola, E., Lepetit, V., Fua, P.: Daisy: An efficient dense descriptor applied to wide-baseline stereo. T.PAMI **32** (2010)
4. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional SIFT descriptor and its application to action recognition. In: Int. Conf. on Multimedia. (2007)
5. Derpanis, K., Sizintsev, M., Cannons, K., Wildes, R.: Efficient action spotting based on a spacetime oriented structure representation. In: CVPR. (2010)
6. Kläser, A., Marszałek, M., Schmid, C.: A spatio-temporal descriptor based on 3D-gradients. In: BMVC. (2008)
7. Laptev, I., Lindeberg, T.: Space-time interest points. In: ICCV. (2003)
8. Sizintsev, M., Wildes, R.: Spatiotemporal stereo via spatiotemporal quadric element (stequel) matching. In: CVPR. (2009)
9. Ke, Y., Sukthankar, R.: PCA-SIFT: A more distinctive representation for local image descriptors. In: CVPR, Washington, USA (2004) 511–517
10. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: SURF: Speeded up robust features. CVIU (2008)
11. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. T.PAMI (2002)
12. Kokkinos, I., Yuille, A.: Scale invariance without scale selection. In: CVPR. (2008)
13. Moreno-Noguer, F.: Deformation and illumination invariant feature point descriptor. In: CVPR. (2011)
14. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. T.PAMI **23** (2001) 1222–1239
15. Zhang, L., Curless, B., Seitz, S.M.: Spacetime stereo: Shape recovery for dynamic scenes. In: CVPR. (2003)
16. Davis, J., Ramamoothi, R., Rusinkiewicz, S.: Spacetime stereo: A unifying framework for depth from triangulation. In: CVPR. (2003)
17. Rodriguez, M., Ahmed, J., Shah, M.: Action MACH: A spatio-temporal maximum average correlation height filter for action recognition. In: CVPR. (2008)
18. Carceroni, R., Kutulakos, K.: Multi-view scene capture by surfel sampling: From video streams to non-rigid 3D motion, shape reflectance. In: ICCV. (2001) 60–67
19. Zhang, Y., Kambhamettu, C., Kambhamettu, R.: On 3D scene flow and structure estimation. In: CVPR. (2001) 778–785
20. Huguet, F., Devernay, F.: A variational method for scene flow estimation from stereo sequences. In: ICCV, Rio de Janeiro, Brasil (2007)
21. Wedel, A., Rabe, C., Vaudrey, T., Brox, T., Franke, U., Cremers, D.: Efficient dense scene flow from sparse or dense stereo data. In: ECCV, Marseille, France (2008)
22. Liu, C.: Beyond pixels: Exploring new representations and applications for motion analysis. In: PhD Thesis, MIT. (2009)
23. Fusiello, A., Trucco, E., Verri, A.: A compact algorithm for rectification of stereo pairs. Machine Vision and Applications **12** (2000) 16–22
24. Moreno-Noguer, F., Porta, J., Fua, P.: Exploring ambiguities for monocular non-rigid shape estimation. In: ECCV. (2010) 370–383
25. Simo-Serra, E., Ramisa, A., Alenya, G., Torras, C., Moreno-Noguer, F.: Single image 3D human pose estimation from noisy observations. In: CVPR. (2012)