Dense Matching of Multiple Wide-baseline Views

Christoph Strecha
KU Leuven
Belgium
firstname.surname@esat.kuleuven.ac.be

Tinne Tuytelaars
KU Leuven
Belgium

Luc Van Gool
KU Leuven / ETH Zürich
Belgium / Switzerland

Abstract

This paper describes a PDE-based method for dense depth extraction from multiple wide-baseline images. Emphasis lies on the usage of only a small amount of images. The integration of these multiple wide-baseline views is guided by the relative confidence that the system has in the matching to different views. This weighting is fine-grained in that it is determined for every pixel at every iteration. Reliable information spreads fast at the expense of less reliable data, both in terms of spatial communications within a view and in terms of information exchange between the views. Changes in intensity between images can be handled in a similar fine grained fashion.

1. Introduction

During the last few years more and more user-friendly solutions for 3D modelling have become available. Techniques have been developed [8] to reconstruct scenes in 3D from video or images as the only input. The strength of these so-called shape-from-video techniques lies in the flexibility of the recording, the wide variety of scenes that can be reconstructed and the ease of texture extraction.

Typical shape-from-video systems require large overlap between subsequent frames. This requirement is usually fulfilled for video sequences. Often, however, one would like to reconstruct from a small number of still images, taken from very different viewpoints. Indeed it might not always be possible to record video of the object of interest, e.g. due to obstacles or time pressure. Furthermore there is a wide class of applications where the images are not taken for the *purpose* of 3D modelling, but a reconstruction is desirable afterwards.

Recently, local, viewpoint invariant features [18, 3, 11, 12], have made wide-baseline matching possible, and hence the viewpoints can be further apart. However, these techniques yield only a sparse set of matching features. This paper focuses on the problem of dense matching under wide-

Figure 1. *3 wide-baseline views of the 'bookshelf' scene. Top: images 1 and 2 with matching invariant regions; bottom: the same for images 1 and 3.*

baseline conditions.

Wide-baseline matching tallies with the development of digital cameras. The low resolution cameras which were available in the past required the use of many images (video) to build accurate 3D models. The redundancy coming from their overlap also helped to overcome noise. Digital cameras today, on the other hand, have resolutions in the order of thousands of pixels and the images are less noisy. Wide-baseline reconstruction based on high-resolution stills may soon outperform small baseline, video-based reconstruction with its low-res input.

This said, stereo matching has been studied mainly in the context of small baseline stereo and for almost frontoparallel planes. Many such algorithms are available today, based on a diversity of concepts (e.g. minimal path search [19], graph cuts [9, 4, 22], etc.). Recently a comparative study has been published by Scharstein *et al.* [14]. There are also several approaches that combine many views, often taken from all around the object. Examples are voxel carving [10], photo hulls [15], and level sets [6]. They use
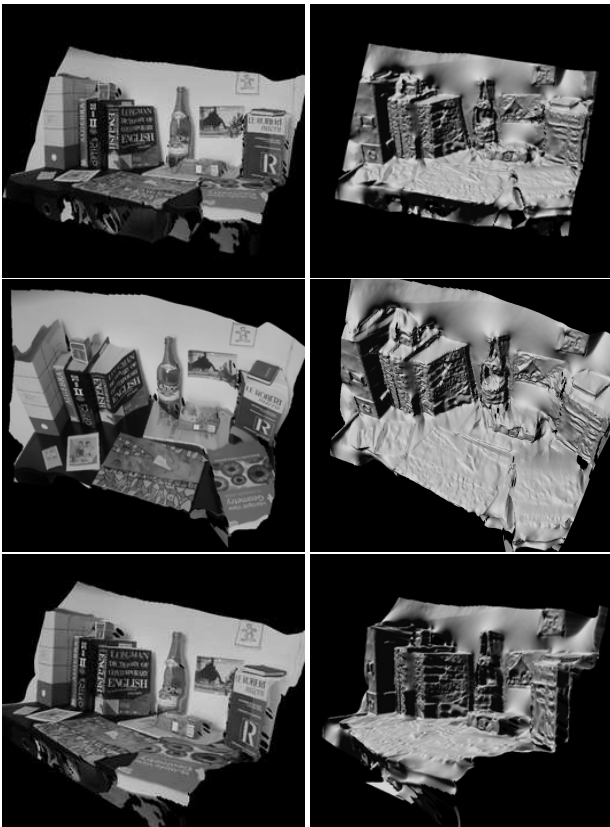
Figure 2. *Textured and untextured views of the reconstruction from the scene in fig. 1. Shown is the reconstruction from the inverse depth map d_i.*

a large amount of images integrated into a single scheme. Several of these approaches use a discretized volume and restrict possible depth values (3D points) to a predefined accuracy. This is not the case for pixel-based PDE approaches [1, 17, 16], which do not need 3D discretization and compute depth (disparity) with higher (machine) precision for every pixel. For large images with fine details, it is questionable whether volume based algorithms can combine reasonable speed and memory requirements with high accuracy.

Current image-based PDE solutions for 3D reconstruction have been proposed for stereo [1] and multi-view stereo [17]. They are faster than earlier PDE-based methods, due to efficient implicit discretization [2]. So far, these algorithms have only been evaluated for small baseline stereo. In the case of wide-baseline stereo, however, the following two constraints that exist in many stereo algorithms should be reduced:
-the uniqueness constraint (as in [9, 4]), as several pixels in one image should be allowed to correspond to a single pixel in another, e.g. due to large scale changes, and
-the ordering constraint along epipolar lines as in [4, 9, 19].

A wide-baseline system should also be able to cope with large occlusions and depth discontinuities, as well as with intensity changes in the images. All these effects will be more outspoken under wide-baseline conditions.

The proposed algorithm offers extensions to our earlier work [17] in that it includes a smoothness term based on a newly defined diffusion tensor. It is no longer restricted to precalibrated cameras and a diffusion process with variable time steps ('inhomogeneous time') guides the convergence. Our system is also simplified in the sense that the number of diffusion equations is reduced. We also extended the approach to vector valued images (color). The algorithm is also akin to the work of Alvarez *et al.* [1], but we do not use their assumption of small occlusions and handle more than two images.

The paper is organized as follows. Section 2 describes the built-in calibration step and introduces the depth parameterization of corresponding points as a key to integrate multiple stereo views. Section 3 discusses the algorithm using N stereo views in a single scheme to extract depth, light scaling between matching points, and their level of consistency. In section 4 we introduce the inhomogeneous time diffusion and the discretization of the PDE system. Section 5 shows experiments on real images and section 6 concludes the paper.

2. Preprocessing and parameterization

Starting point of the proposed wide-baseline system is the matching of affine invariant regions across all images. This replaces the typical matching of (Harris) corners in traditional, uncalibrated structure-from-motion. Here we use the affine invariant region implementation in [18] and use the method of [7] to further boost the number of multi-view matches. Fig. 1 shows the output of the affine invariant region matching for three wide-baseline images of a bookshelf.

Exploiting these matches, a self-calibration procedure similar to shape-from-video methods is set up, followed by a bundle adjustment, in order to reduce errors in the camera parameters and the 3D structure of the feature point cloud. Then follows the actual search for dense correspondences, just as in the case of video, but now for wide-baseline input. This search exploits both the knowledge of the camera parameters and the discrete set of matching features that both are available at that point.

The integration of multiple views is simplified through the use of a unified, physically relevant parameterization of corresponding points. We use inverse depth. Consider N images $I_i, i = 1..N$. We choose the center of projection of the first camera as the Euclidean coordinate center. A 3D point denoted by $\mathbf{X} = (X, Y, Z, 1)^T$ is projected to the

Figure 3. *Top: Four images used in the wide-baseline experiment. Bottom: depth maps for these images (dark pixels indicate low consistency regions).*

coordinates $\mathbf{x}_i = (x_i, y_i, 1)^T$ in image I_i through:

$$\lambda_i \mathbf{x}_i = \mathbf{K}_i [\mathbf{R}_i^T | - \mathbf{R}_i^T \mathbf{t}_i] \mathbf{X} \qquad (1)$$

where $\mathbf{K}_i$ is the usual camera calibration matrix, $\mathbf{R}_i$ is the 3×3 rotation matrix specifying the relative orientation of the camera and $\mathbf{t}_i = (t_x, t_y, t_z)^T$ is the translation vector between the first and the ith camera ($\mathbf{R}_1 = \mathbf{1}, \mathbf{t}_1 = \mathbf{0}$).

From eq. (1) the epipolar lines between all N views can be derived. It follows for corresponding image points $\mathbf{x}_i$ and $\mathbf{x}_j$ in the ith and jth image and for a *coordinate system that is attached to the i*th camera ($\mathbf{R}_i = \mathbf{1}$, $\mathbf{t}_i = \mathbf{0}$) that:

$$\frac{\lambda_j}{Z_i} \mathbf{x}_j = \mathbf{K}_j \mathbf{R}_j^T \mathbf{K}_i^{-1} \mathbf{x}_i + \frac{1}{Z_i} \mathbf{K}_j \tilde{\mathbf{t}}_{\mathbf{j}} \qquad (2)$$

with $\tilde{\mathbf{t}}_{\mathbf{j}} = -\mathbf{R}_j^T \mathbf{t}_j$. Note that Z_i is proportional to λ_i. The stereo correspondence is divided into a component that depends on the rotation and pixel coordinate (according to the homography $\mathbf{H}_{ij} = \mathbf{K}_j \mathbf{R}_j^T \mathbf{K}_i^{-1}$) and a depth dependent part that scales with the amount of translation between the cameras. The corresponding point in the jth image for a point $\vec{x}_i$[1] in the ith image will be written as [17]:

$$\vec{l}(\vec{x}_i, d_i) = \frac{\begin{pmatrix} \mathbf{H}_{ij}[1]\mathbf{x}_i \\ \mathbf{H}_{ij}[2]\mathbf{x}_i \end{pmatrix} + d_i \begin{pmatrix} \mathbf{K}_j[1]\tilde{\mathbf{t}}_j \\ \mathbf{K}_j[2]\tilde{\mathbf{t}}_j \end{pmatrix}}{\mathbf{H}_{ij}[3]\mathbf{x}_i + d_i \tilde{t}_{z_j}} \qquad (3)$$

with the parameter $d_i = \frac{1}{Z_i}$, i.e. a *depth related parameter*. $\mathbf{H}_{ij}[k]$ is the 3-vector for the k *th* row of the homography $\mathbf{H}_{ij}$ and similarly for $\mathbf{K}_j[k]$.

---

[1]In the following the vector sign describes non-homogeneous pixel coordinates $\vec{x}_i = (x_i, y_i)$

3. Integration of multiple views

Given N images $i = 1..N$, that have been calibrated as described, our approach is based on the minimization of the following cost functionals for the different cameras, and written here in terms of the i*th* camera:

$$\begin{aligned} E_i[d_i, \kappa_{ij}] = & \int \sum_{j \neq i}^{N} c_{ij} |\kappa_{ij} I_i(\vec{x}) - I_j(\vec{l}(\vec{x}, d_i))|^2 d\vec{x} \\ & + \lambda_1 \int (\nabla d_i)^T D(\nabla C_i) \nabla d_i d\vec{x} \\ & + \lambda_2 \int \sum_{j \neq i}^{N} |\nabla \kappa_{ij}|^2 d\vec{x}. \qquad (4) \end{aligned}$$

In fact the minimum of this energy for all cameras (usually not more than 10) will be estimated by systems of coupled PDE's, described in the sequel. This functional will now be explained term by term.

The first term quantifies the similarity of the intensity patterns in the matching points. Note that it replaces the traditional brightness constancy assumption by a local intensity scaling $\kappa_{ij}(\vec{x})$. This factor κ is introduced to account for the changes in the lighting conditions, which tend to be more severe under wide-baseline conditions. Note that the norm in this term can be defined as an appropriate distance for vector valued functions, like e.g. a distance in a color space or a distance between outputs of a filter bank. Typically a pixel $\vec{x}$ in the ith image can obtain a depth value $d_i(\vec{x})$ through matching with *all* remaining $N - 1$ views. This is true for all pixels that do not fall victim to occlusions. The local weight $c_{ij}[\vec{x}]$ takes care of such situations. This value will be close to one if the match has homed in on the correct solution and close to zero if no consistent match
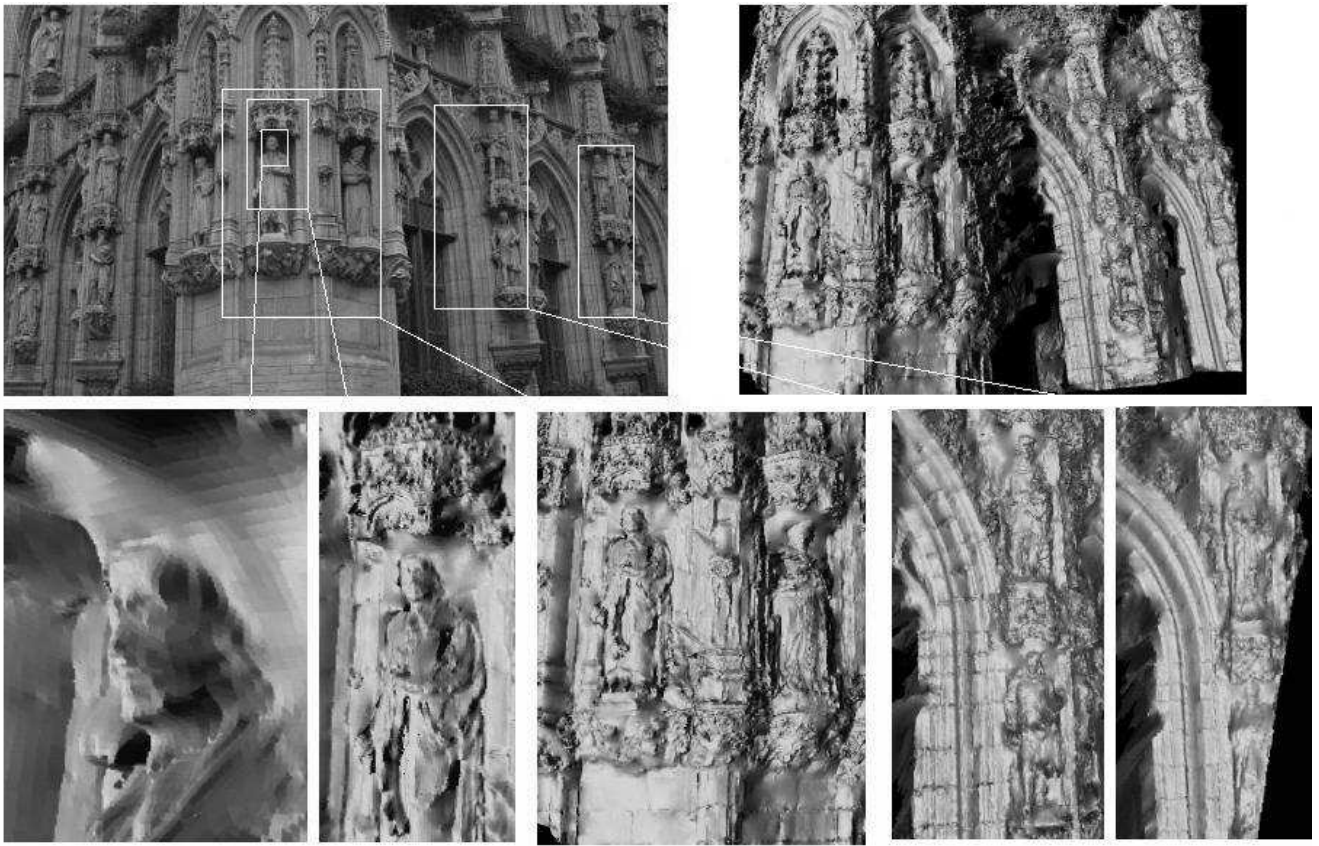
Figure 4. *Views of the untextured 3D reconstruction from the depth map d_1 of the left image in fig. 3.*

has been found (as in the case of occlusion). How c_{ij} is calculated is explained in more detail shortly.

Since we do not know the value of $I_j(\vec{l}(\vec{x}, d_i))$ in eq. 4, we split the inverse depth value $d_i(\vec{x})$ into a current $d_{i_0}(\vec{x})$ and a residual $d_{i_r}(\vec{x})$ estimate. This is similar to the bi-local strategy used by Proesmans *et al.* [13]. These two estimates sum to the solution d_i in each iteration. Taylor expansion around d_{i_r} and neglecting $O(d_{i_r}^2)$ and the higher order terms gives for this term in eq. 4:

$$
\begin{aligned}
I_j(\vec{l}(\vec{x}, d_{i_0} + d_{i_r})) &= I_j(\vec{l}(\vec{x}, d_{i_0})) + \frac{\partial I_j}{\partial \vec{x}} d_{i_r} \\
\frac{\partial I_j}{\partial \vec{x}} &= \left(\frac{\partial I_j(\vec{l}(\vec{x}, d_{i_0}))}{\partial x} \frac{t_1 h_3 - t_3 h_1}{(h_3 + d_o t_3)^2} \right. \\
&\quad \left. + \frac{\partial I_j(\vec{l}(\vec{x}, d_{i_0}))}{\partial y} \frac{t_2 h_3 - t_3 h_2}{(h_3 + d_o t_3)^2} \right),
\end{aligned}
$$

with

$$
\begin{aligned}
h_1 &= \mathbf{H}_{ij}[1]\mathbf{x}_i, \ h_2 = \mathbf{H}_{ij}[2]\mathbf{x}_i, \ h_3 = \mathbf{H}_{ij}[3]\mathbf{x}_i \\
k_1 &= \mathbf{K}_j[1]\tilde{\mathbf{t}}_j, \ k_2 = \mathbf{K}_j[2]\tilde{\mathbf{t}}_j.
\end{aligned}
$$

The second term in eq. (4) regularizes the inverse depths d_i. It forces the solutions to be smooth, while preserving depth discontinuities through anisotropic diffusion. Various anisotropic diffusion operators have been considered in the literature, mostly in the context of optical flow, disparity [13, 5, 2, 21] computation, and image enhancement [20]. We tested four operators. Following Weickert *et al.*'s taxonomy [21], the first is a 'nonlinear, anisotropic flow (consistency) driven diffusion operator', the second a 'linear, anisotropic image driven operator' (equivalent to [1]) and the third a 'nonlinear, anisotropic flow driven diffusion operator'. The 4*th* is the one proposed by Proesmans *et al.* [13]. All yield similar, satisfactory results and further studies are needed to evaluate the operators in detail. The implementations in this paper are based on the first, with as diffusion tensor:

$$
D(\nabla C_i) = \frac{1}{|\nabla C_i|^2 + 2\nu^2} \left\{ \begin{bmatrix} \frac{\partial C_i}{\partial y} \\ -\frac{\partial C_i}{\partial x} \end{bmatrix} \begin{bmatrix} \frac{\partial C_i}{\partial y} \\ -\frac{\partial C_i}{\partial x} \end{bmatrix}^T + \nu^2 \mathbf{1} \right\}.
$$

This tensor stops diffusion at pixels for which no convincing match has been found in any of the other images. This situation is signalled by systematically low 'consistencies' of the current matches with the other views. Before introducing this consistency concept, it has to be mentioned that correspondence search always occurs in two directions in our
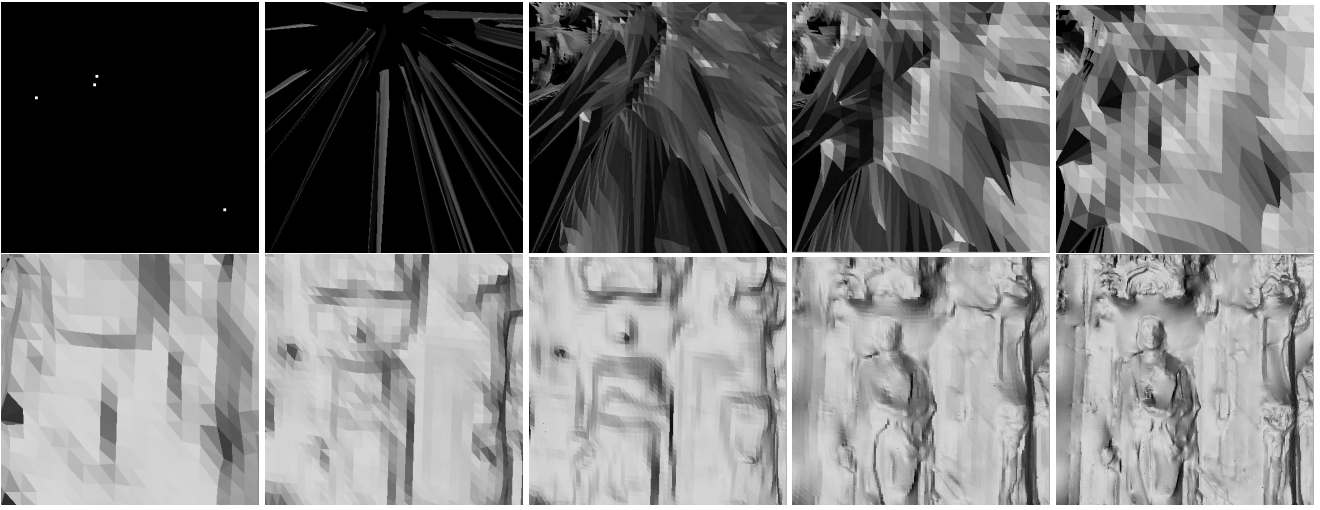
Figure 5. *Evolution of a part of the 3D mesh for the left image of the scene in fig: 3. Top: lowest scale, left four initial seed points, they stay more or less fixed during diffusion as described in the text (right images). Bottom: the evolution when going to still finer scales.*

overall scheme. If currently camera i is looking for matches in camera j, the reverse will happen when camera j is being considered during the same, global iteration. In terms of camera i the first search is called *forward matching*, the second *backward matching*. Ideally, matches in images i and j point at eachother, i.e. the displacement vectors from one to the other and v.v. should sum to zero. We put this in more formal terms now. Defining $\vec{p} = \vec{l}_{ij}(\vec{x}, d_i) \in I_j$ as the corresponding point to pixel $\vec{x} \in I_i$ and its backward match $\vec{q} = \vec{l}_{ji}(\vec{p}, d_j) \in I_i$, the confidence c_{ij} is a function of the forward-backward error $|\vec{q} - \vec{x}|$. The 'consistency' $c_{ij}(\vec{x})$ at $\vec{x}$ in image i with its aledged match in image j is computed as: $c_{ij} = 1/(1 + |\vec{q} - \vec{x}|/k)$. C_i – the overall consistency – is the maximum of all consistencies from image i to the others, i.e. $C_i = max(c_{ij}) \forall j \neq i$. Rather than letting diffusion be driven by all intensity edges, as done by Alvarez *et al.* [2], our scheme is less indiscriminate and targets its caution more specifically towards regions where consistent and inconsistent matches come close.

Finally, the third term forces the intensity scaling to vary smoothly across the image.

To minimize the cost functional eq.(4) we rather solve the corresponding gradient descent equations for each camera:

$$\partial_t d_i = \lambda_1 div(D(\nabla C_i)\nabla d_i) \tag{5}$$
$$- \sum_{j \neq i}^N \frac{c_{ij}\partial I_j^\sigma}{\partial \vec{x}}\left(\kappa_{ij}I_i^\sigma - I_j^\sigma + \frac{\partial I_j^\sigma}{\partial \vec{x}}(d_i - d_{i_0})\right)$$
$$\partial_t \kappa_{ij} = \lambda_2 \nabla^2 \kappa_{ij}$$

$$- c_{ij}I_i^\sigma \left(\kappa_{ij}I_i^\sigma - I_j^\sigma + \frac{\partial I_j^\sigma}{\partial \vec{x}}(d_i - d_{i_0})\right).$$

The superscript σ indicates the Gauss convolved version of the symbol. Indeed, the scheme is embedded in a multi-scale procedure to avoid convergence into local minima [17]. This set of coupled diffusion equations is solved in turn for each image i (first equation) and each image pair i, j (second equation). Hence, we have a single eq. for the depth parameter, but $N - 1$ eqs. for the intensity factors κ. As the eqs. are solved in turn for each camera, this yields N^2 eqs. for a single, global iteration in case one wants to deal with changing intensities, but only N equations if one adheres to the brightness constancy assumption. This preliminary strategy for κ extraction quickly becomes expensive and more efficient, transitivity exploiting schemes can be devised.

After each iteration the confidence values c_{ij} are updated according to the forward-backward mechanism similar to Proesmans *et al.* [13, 17]. Note that we don't regularize the c_{ij} as Proesmans does.

4. Discretization and initialization

Starting point for the diffusion process defined by eq. (5) is an initial value. This is zero ($d_i(\vec{x}) = 0$) for most pixels except for those that have been used in the calibration procedure and remained valid also after self-calibration and bundle adjustment. Using this information is especially useful when dealing with wide-baselines and even more if the scene contains ordering changes along epipolar lines.

Figure 6. *Intensity scaling experiment: from left to right (a)-(f) In each row: original image(a); inverse depth map(b); (c,d) the two images(a) from the other rows warped to the image(a) in the row taking into account their depth maps and the intensity scalings κ_{ij}; the intensity scalings κ_{ij} (e,f) that accounts for the light change from the two images(a) of the other rows to the image(a) in the row (κ_{ij} maps).*

Hence, matching invariant regions provide sparse, initial depth values. These values have themselves confidence values attached, that are related to the residual of the bundle. We here introduce a novel framework to deal with varying confidences in a diffusion-like process. Without this modification the very sparse set of consistent depth values would be pulled away by the majority of uninitialized depth values to fulfill the smoothness assumption of the underlying diffusion process. To prevent this we use an inhomogeneous time diffusion process. Clocks will not run any more with equal speeds throughout space as it is for a usual diffusion process. Different pixels diffuse at a different time scale, that is related to the pixel confidence. High confidence pixels diffuse much slower than low confidence pixels. After implicit discretization of eq. 5 this inhomogeneous time diffusion is realized by replacing the usually constant time step size τ by the local step size $\tau(\vec{x})$. We get for the first equation in eq. 5:

$$\frac{d_i^{t+1} - d_i^t}{\tau(\vec{x})} = div(D(\nabla C_i)\nabla d_i^{t+1}) \qquad (6)$$

$$- \quad \lambda \sum_{j \neq i}^{N} \frac{c_{ij}\partial I_j^\sigma}{\partial \vec{x}} \left(\kappa_{ij} I_i^\sigma - I_j^\sigma + \frac{\partial I_j^\sigma}{\partial \vec{x}}(d_i^{t+1} - d_i^t) \right).$$

Figure 5 shows the diffusion process over time for experiment 2 (see section 5) on fig. (3). In the top row four initialization points of a part of the first image are shown (left) for the computation at the lowest scale. During iterations these points remain almost fixed due to their slower time scale. Other pixels will be attracted much faster to these points (top right images). At the end (bottom row left to right) initial depth values get also consistent with the surrounding depth's and with the energy 4 - diffusion reaches the solution.

5. Results for real images

We tested the proposed algorithms on real data.

• Experiment 1 (strong wide-baseline): The three bookshelf images of fig. 1 are the input. This figure also shows
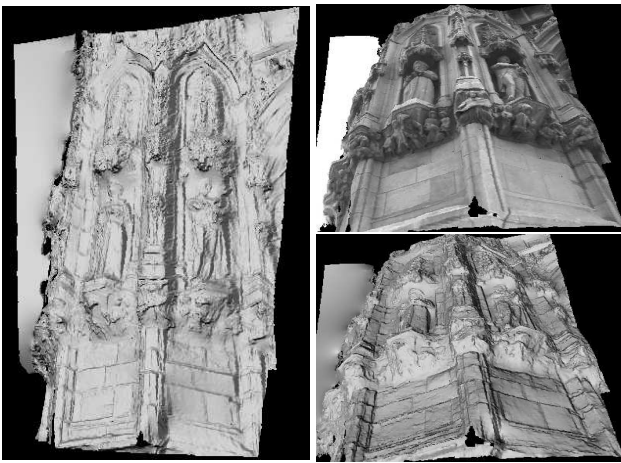
Figure 7. *Views of the 3D untextured (left, right bottom) and textured (right top) reconstruction from experiment 3 top left image in fig. 6*

the extracted affine invariant regions. The 3D reconstructions computed from the depth map of each of the three images are shown in fig. 2 In this experiment we did not use the intensity scale maps κ_{ij} as a variable (i.e. $\kappa_{ij} = 1$). The image size is 640×480 pixels and the number of initial 3D points that passed the calibration procedure was 74, 65, and 73, resp., for the three images. We applied the minimization based on the Euclidean distance in RGB color space.

• Experiment 2 (strong scale changes, large image size): Fig. 3 shows the four images with the inverse depth maps d_i for each image underneath. Dark pixels in these depth maps indicate regions of low overall consistency C_i. There no consistent match could be established in any of the other images. The 3D reconstruction of the left image of fig. 3 is shown in fig. 4. We used 1536×1024 gray value images without κ (intensity scale) updating. The number of initial depth values was 33, 30, 17, and 14, respectively for the images from left to right in fig. 3.

• Experiment 3 (intensity scaling). The images in fig. 6 (left) have been taken at three different days. A clear change in lighting conditions is visible. The inverse depth maps are shown next to the original images. Then follow the two texture mapped, κ-corrected views of the other camera-centered reconstructions seen from the viewpoint of the initial camera. Ideally, these images should be very similar for the overlapping part, which can be seen to be the case. This shows the correctness of the κ maps seen in the right two images. Gray value images of size 720×576 have been used with 64, 54, and 64 initial depth values, resp. Fig. 7 shows the reconstruction from the first image's depth map d_1.

• Experiment 4 (few, high-res stills). This experiment shows the capability of the algorithm to get good 3D from a small amount of images. Three still 1536×1024 images in a rather small baseline situation have been used. These are the two left images of fig. 3 together with a third one not shown. The result can be seen in fig. 8.

6. Summary and conclusions

We have proposed a multi-view wide-baseline stereo system for the reconstruction of precise 3D models. From our wide-baseline stereo pipeline, - affine invariant feature matching - selfcalibration - bundle adjustment - dense reconstruction, this paper has focused on the dense reconstruction. Armed with a very sparse set of initial depth estimates we developed an efficient algorithm to propagate these by an inhomogeneous time diffusion process, that is guided by a properly weighted matching energy that takes into account the matching to all views. The algorithm can deal with occlusions, light changes, ordering changes along epipolar lines and extensive changes in scale. Good quality reconstructions can be obtained from a very small set of images. The method can handle large image sizes due to its comparatively small memory usage. The computation time is reasonable, e.g. 15 min. in experiment 2 using four 1536x1024 images (on a Athlon 2200 based computer), incl. the triangulation of all four images.

References

[1] L. Alvarez, R. Deriche, J. Sánchez, and J. Weickert. Dense disparity map estimation respecting image derivatives: a pde and scale-space based approach. *Journal of Visual Communication and Image Representation*, 13(1/2):3–21, 2002.

[2] L. Alvarez, J. Weickert, and J. Sánchez. Reliable estimation of dense optical flow fields with large displacements. *IJCV*, pages 41–56, 2000.

[3] A. Baumberg. Reliable feature matching across widely separated views. *Proc. CVPR*, pages 774–781, 2000.

[4] C. Buehler, S. Gortler, M. Cohen, and L. McMillan. Minimal surfaces for stereo. *Proc. ECCV*, 3:885–899, 2002.

[5] R. Deriche, P. Kornprobst, and G. Aubert. Optical flow estimation while preserving its discontinuities: A variational approach. *Proc. 2nd Asian Conf. on Computer Vision*, 2:71–80, DEC 1995.

[6] O. Faugeras and R. Keriven. Complete dense stereovision using level set methods. *Proc. ECCV*, pages 379–393, 1998.

[7] V. Ferrari, T. Tuytelaars, and L. Van Gool. Wide-baseline muliple-view correspondences. *Proc. CVPR*, pages 718–725, 2003.

[8] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 1998.

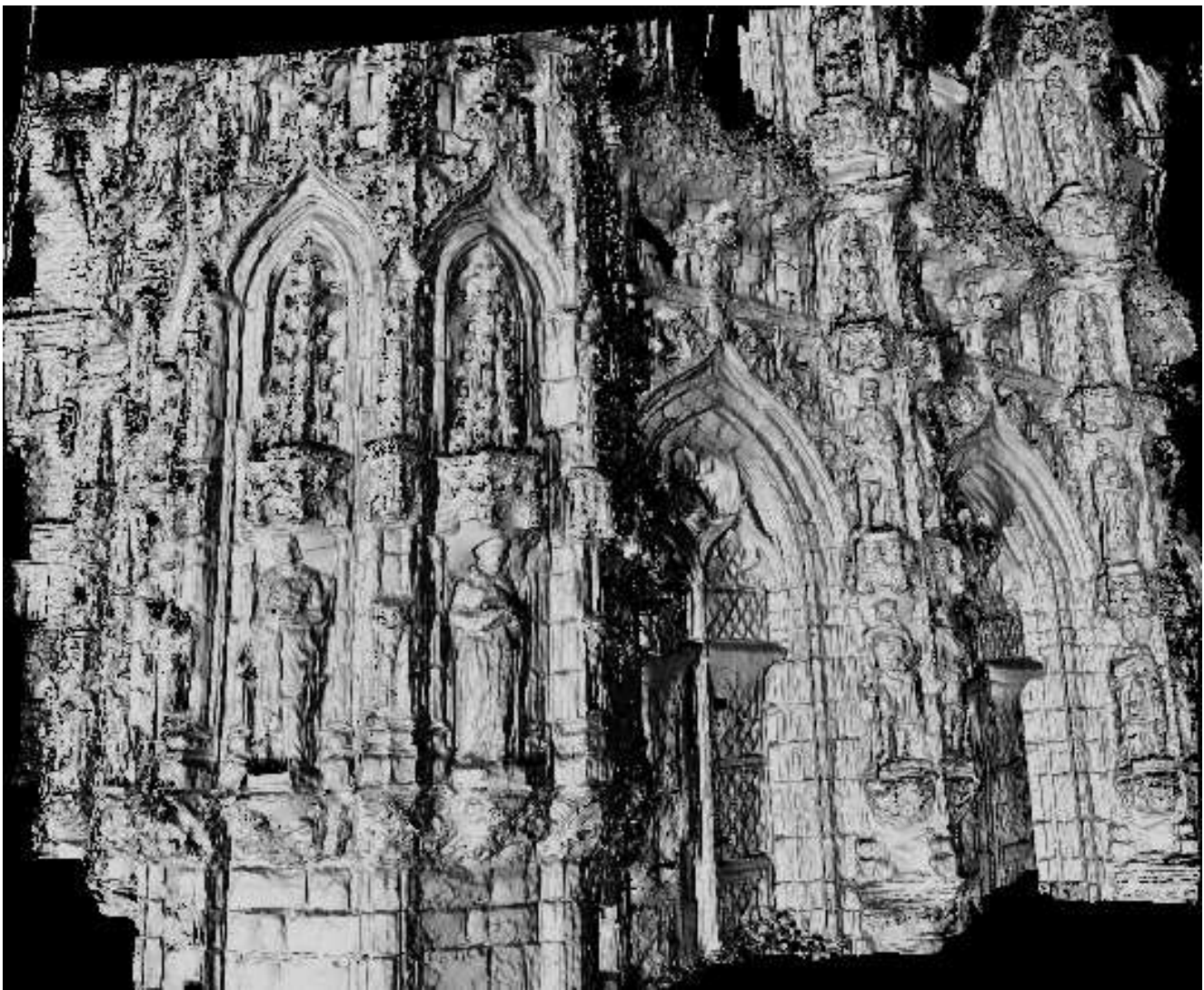[9] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. *Proc. ECCV*, 3:82–96, 2002.

Figure 8. *View of the untextured 3D reconstruction of experiment 4.*

[10] K. Kutulakos and S. Seitz. A theory of shape by space carving. *IJCV*, 38(3):197–216, 2000.

[11] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo for maxinally stable external regions. *Proc. BMVC*, pages 414–431, 2002.

[12] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. *Proc. ECCV*, 1:128–142, 2002.

[13] M. Proesmans, L. Van Gool, E. Pauwels, and A. Oosterlinck. Determination of optical flow and its discontinuities using non-linear diffusion. *Proc. ECCV*, 2:295–304, 1994.

[14] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1/2/3):7–42, 2002.

[15] G. Slabaugh, R. Schafer, and M. Hans. Image-based photo hulls. *1st Int. Symp. of 3D Data Processing Visualization and Transmission*, pages 704–707, 2002.

[16] C. Strecha and L. Van Gool. Motion-stereo integration for depth estimation. *Proc. ECCV*, 2:170–185, 2002.

[17] C. Strecha and L. Van Gool. Pde-based multi-view depth estimation. *1st Int. Symp. of 3D Data Processing Visualization and Transmission*, pages 416–425, 2002.

[18] T. Tuytelaars and L. Van Gool. Wide baseline stereo matching based on local, affinely invariant regions. *Proc. BMVC*, pages 412–422, 2000.

[19] G. Van Meerbergen, M. Vergauwen, M. Pollefeys, and L. Van Gool. A hierarchical stereo algorithm using dynamic programming. *IJCV - Stereo and multi-baseline vision*, 47(1-3):275–285, 2002.

[20] J. Weickert. Theoretical foundations of anisotropic diffusion in image processing. *Computing, Suppl.*, 11:221–236, 1996.

[21] J. Weickert and T. Brox. Diffusion and regularization of vector- and matrix-valued images. *Inverse Problems, Image Analysis, and Medical Imaging. Contemporary Mathematics, AMS, Providence*, 313:251–268, 2002.

[22] Y.Boykov, O.Veksler, and R.Zabih. Fast approximate energy minimization via graph cuts. *Proc. ICCV*, 1:377–384, 1999.