Multi Viewpoint Stereo from Uncalibrated Video Sequences

Reinhard Koch, Marc Pollefeys, and Luc Van Gool

Katholieke Universiteit Leuven,
Kardinaal Mercierlaan 94, B-3001 Leuven, Belgium
Email: firstname.lastname@esat.kuleuven.ac.be
http://www.esat.kuleuven.ac.be/psi/visics.html

Abstract. This contribution describes an automatic 3D surface modeling system that extracts dense metric 3D surfaces from an uncalibrated video sequence. A static 3D scene is observed from multiple viewpoints by freely moving a video camera around the object. No restrictions on camera movement and internal camera parameters like zoom are imposed, as the camera pose and intrinsic parameters are calibrated from the sequence.

Dense surface reconstructions are obtained by first treating consecutive images of the sequence as stereoscopic pairs and computing dense disparity maps for all image pairs. All viewpoints are then linked by controlled correspondence linking for each image pixel. The correspondence linking algorithm allows for accurate depth estimation as well as image texture fusion from all viewpoints simultaneously. By keeping track of surface visibility and measurement uncertainty it can cope with occlusions and measurement outliers. The correspondence linking is applied to increase the robustness and geometrical resolution of surface depth as well as to remove highlights and specular reflections, and to create super-resolution texture maps for increased realism.

The major impact of this work is the ability to automatically generate geometrically correct and visually pleasing 3D surface models from image sequences alone, which allows the economic model generation for a wide range of applications. The resulting textured 3D surface model are highly realistic VRML representations of the scene.

1 Introduction

3D surface reconstruction from image sequences is an ongoing research topic in the computer vision society. For calibrated sequences good results were obtained by structure-from-motion algorithms or stereoscopic image sequences (see [10, 26, 16]). Contributions were made for multi baseline stereo [19], incremental depth estimation [18] and multi viewpoint analysis [9] from known camera positions.

Metric reconstruction from uncalibrated sequences is still under investigation. In the uncalibrated case all parameters - camera pose and intrinsic calibration as well as the 3D scene structure - have to be estimated from the 2D image

sequence alone. Faugeras and Hartley first demonstrated how to obtain projective reconstructions from such image sequences [6, 11]. Since then, researchers tried to find ways to upgrade these reconstructions to metric (i.e. Euclidean but unknown scale, see for example [7, 22, 25]).

They mostly restricted themselves to constant intrinsic parameters. Recently, extensions of this work to varying intrinsic camera parameters were proposed [21, 12, 23].

To employ these self-calibration methods for sequence analysis they must be embedded in a complete scene reconstruction system. Beardsley et al. [1] proposed a scheme to obtain projective calibration and 3D structure by robustly tracking salient scene feature points throughout an image sequence. This sparse object representation outlines the object shape, but gives insufficient surface detail for visual reconstruction. Highly realistic 3D surface models need dense depth maps and can not rely on relatively few feature points.

In [23] the method of Beardsley et al. [1] was extended in two directions: On the one hand the projective reconstruction was updated to metric even for varying internal camera parameters. On the other hand a dense stereo matching technique [4] was applied between two selected images of the sequence to obtain a dense depth map for a single viewpoint. From this depth map a triangular surface wire-frame was constructed and texture mapping from one image was applied to obtain realistic surface models.

In this contribution we extend the dense surface reconstruction as proposed in [23] to sequence analysis. An algorithm is proposed that links image correspondences over the image sequence and allows to integrate both depth and image texture. Only weak restrictions are imposed on the camera motion and on scene geometry, and the approach is embedded in a completely uncalibrated and automatic framework. It will be shown how the analysis of surface depth and surface texture from the image sequence will improve the accuracy of the surface model. At the same time texture integration opens new means to enhance the object texture quality. The creation of super-resolution texture maps and removal of imaging artifacts or specular reflections are additional features of the proposed algorithm.

1.1 Overview of 3-D Reconstruction System

The complete 3-D surface reconstruction system consists of several modules to be executed in a processing pipeline. Some modules were discussed in earlier publications and will be sketched only. The system structure can be summarized as follows:

1. Sparse Point Tracking and Calibration. Section 2 reviews the feature point tracking algorithm. It uses a 2 step approach:
   (a) obtain projective calibration and 3D point reconstruction,
   (b) upgrade to metric calibration and structure through self-calibration based on constraints applied to the *absolute quadric*.

2. Pairwise Dense Disparity Matching. Section 3 deals with dense disparity measurements from image pairs. Adjacent images of the sequence are treated as stereo image pairs and dense correspondence maps are computed between these pairs [4].
3. Correspondence Linking Algorithm. In Sect. 4 the correspondence linking algorithm is developed which links all possible image point correspondences over the sequence, guided by a depth verification that allows to test for outliers and occlusions. Depth and texture fusion are derived from the basic linking scheme.
4. 3-D Surface Reconstruction. Textured triangular surface meshes are then generated from refined depth and texture maps to build highly realistic scene models, as described in Sect. 5.

2 Camera Calibration through Feature Point Tracking

Two things are needed to build a 3D model from an image sequence: (1) the calibration[1] of the sequence and (2) the correspondences between the images. Starting from an image sequence acquired by an uncalibrated video camera, both these prerequisites are unknown and therefore have to be retrieved from image data. Only a few but very reliable image correspondences are needed to retrieve the calibration of the camera setup. Salient feature points like strong intensity corners are robustly tracked throughout the image sequence for that purpose. In a two-step approach a projective calibration and feature point reconstruction is recovered from the image sequence which is then upgraded to metric calibration with a self-calibration approach.

2.1 Retrieving the Projective Framework

At first, feature correspondences are found by extracting intensity corners in different images and matching them using a robust corner matcher [24]. In conjunction with the matching of the corners a restricted calibration of the setup is calculated (i.e. only determined up to an arbitrary projective transformation). This allows to eliminate matches which are inconsistent with the calibration. The 3D position of a point is restricted to the line passing through its image point and the camera projection center. Therefore the corresponding point is restricted to the projection of this line in the other image. Using this constraint, more matches can easily be found and used to refine this calibration. The principle is explained in Fig. 1.

The matching is first carried out on the first two images. This defines a projective framework in which the projection matrices of the other views are retrieved one by one. In this approach we follow the procedure proposed by Beardsley et al. [1]. We therefore obtain projection matrices (3×4) of the form

[1] By *calibration* we mean the actual internal calibration of the camera as well as the relative position and orientation of the camera for the different viewpoints.

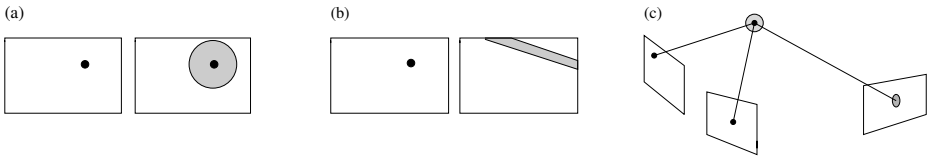

Fig. 1. Images with (a) a priori search region, (b) search region based on the epipolar constraint, (c) prediction of search region in the sequence after projective reconstruction of the point (used for refinement).

$$\mathbf{P}_1 = [\mathbf{I}|0] \text{ and } \mathbf{P}_i = [\mathbf{H}_{1i}|e_{1i}] \tag{1}$$

with $\mathbf{H}_{1i}$ the homography for some reference plane from view 1 to view i and e_{1i} the corresponding epipole.

2.2 Retrieving the Metric Framework

Such a projective calibration is certainly not satisfactory for the purpose of 3D modeling. A reconstruction obtained up to a projective transformation can differ very much from the original scene according to human perception: orthogonality and parallelism are in general not preserved, part of the scene can be warped to infinity, etc. To obtain a better calibration, constraints on the internal camera parameters can be imposed (e.g. absence of skew, known aspect ratio, ...). By exploiting these constraints, the projective reconstruction can be upgraded to metric (Euclidean up to scale).

In that case the camera projection matrices should have the following form:

$$\mathbf{P}_i = \mathbf{K}_i \left[\mathbf{R}_i|\text{-}\mathbf{R}_i\mathbf{t}_i\right] \text{ with } \mathbf{K}_i = \begin{bmatrix} f_x & s & u_x \\ & f_y & u_y \\ & & 1 \end{bmatrix} \tag{2}$$

where $\mathbf{R}_i$ and $\mathbf{t}_i$ indicate the orientation and position of the camera for view i, $\mathbf{K}_i$ contains the internal camera parameters, f_x and f_y stand for the horizontal and vertical focal length (in pixels), $\mathbf{u} = (u_x, u_y)$ is the principal point and s is a measure of the skew.

A practical way to obtain the calibration parameters from constraints on the internal camera parameters is through application of the concept of the absolute quadric [25, 23]. In space, exactly one degenerate quadric of planes exists which has the property to be invariant under all rigid transformations. In a metric frame it is represented by the following 4×4 symmetric rank 3 matrix $\Omega = \begin{bmatrix} \mathbf{I} & 0 \\ 0 & 0 \end{bmatrix}$. If $\mathbf{T}$ transforms points $M \rightarrow \mathbf{T}M$ (and thus $\mathbf{P} \rightarrow \mathbf{P}\mathbf{T}^{-1}$), then it transforms $\Omega \rightarrow \mathbf{T}\Omega\mathbf{T}^\top$ (which can be verified to yield Ω when $\mathbf{T}$ is a similarity transformation)[2]. The projection of the absolute quadric onto the image yields

[2] Using (2) this can be verified for a metric basis. Transforming $\mathbf{P} \rightarrow \mathbf{P}\mathbf{T}^{-1}$ and $\Omega \rightarrow \mathbf{T}\Omega\mathbf{T}^\top$ will not change the projection.

the intrinsic camera parameters independent of the chosen projective basis:

$$\mathbf{K}_i\mathbf{K}_i^\top \propto \mathbf{P}_i\Omega\mathbf{P}_i^\top \tag{3}$$

where $\propto$ means equal up to an arbitrary non-zero scale factor. Therefore constraints on the internal camera parameters in $\mathbf{K}_i$ can be translated to constraints on the absolute quadric. If enough constraints are at hand, only one quadric will satisfy them all, i.e. the *absolute quadric*. At that point the scene can be transformed to the metric frame (which brings Ω to its canonical form).

3 Dense Stereo Pair Matching

With the camera calibration given for all viewpoints of the sequence, we can proceed with methods developed for calibrated structure from motion algorithms. The feature tracking algorithm already delivers a sparse surface model based on distinct feature points. This however is not sufficient to reconstruct geometrically correct and visually pleasing surface models. This task is accomplished by a dense disparity matching that estimates correspondences from the grey level images directly by exploiting additional geometrical constraints.

3.1 Exploiting Scene Constraints

The epipolar constraint obtained from calibration restricts corresponding image points to lie in the epipolar plane[3] which also cuts a 3D profile out of the surface of the scene objects. The profile projects onto the corresponding epipolar lines in the images $\mathbf{I_i}$ and $\mathbf{I_k}$ where it forms an ordered set of neighboring correspondences (see left of Fig. 2).

For well behaved surfaces this ordering is preserved and delivers an additional constraint, known as *ordering constraint*. Scene constraints like this can be applied by making weak assumptions about the object geometry. In many real applications the observed objects will be opaque and composed out of piecewise continuous surfaces. If this restriction holds then additional constraints can be imposed on the correspondence estimation. Kochan [17] listed as many as 12 different constraints for correspondence estimation in stereo pairs. Of them, the two most important apart from the epipolar constraint are:

1. *Ordering Constraint:* For opaque surfaces the order of neighboring correspondences on the corresponding epipolar lines is always preserved. This ordering allows the construction of a dynamic programming scheme which is employed by many dense disparity estimation algorithms [3, 4, 8].
2. *Uniqueness Constraint:* The correspondence between any two corresponding points is bidirectional as long as there is no occlusion in one of the images. A correspondence vector pointing from an image point to its corresponding point in the other image always has a corresponding reverse vector pointing back. This test is used to to detect outliers and occlusions.

[3] The epipolar plane is the plane defined by the the image point and the camera projection centers.

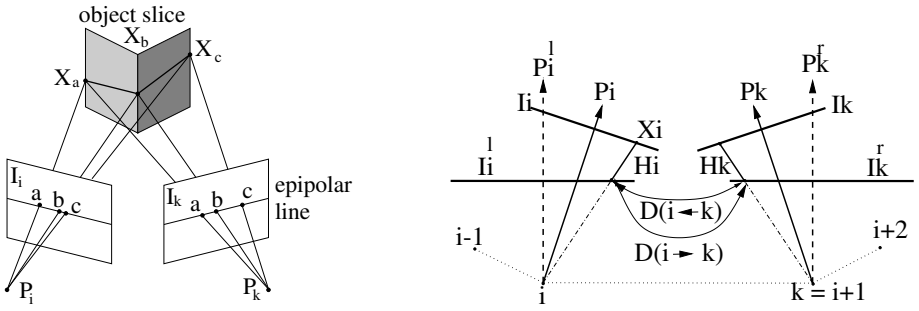

Fig. 2. Left: Object profile triangulation from ordered neighboring correspondences.Right: Rectification and correspondence between viewpoints i and k

3.2 Image Rectification

All above mentioned constraints operate along the epipolar lines which may have an arbitrary orientation in the image planes. The matching procedure is greatly simplified if the image pair is rectified to a standard geometry. In standard geometry both image planes are coplanar and the epipoles are projected to infinity. The rectified image planes can further be oriented such that the epipolar lines coincide with the image scan lines. Image rectification then involves a planar projective mapping from the original towards the rectified image planes. Note that the rectification process is under-determined and there is an additional degree of freedom to be set. Conveniently the projective mapping transformation $\mathbf{H}$ is chosen such that the resulting image distortions are minimized. Possible criteria can be to minimize the change of camera orientation [16] or the deviation of the image corner angles from right angles [5].

Figure 2 (right) clarifies the relation between the original and rectified images and cameras, as well as the correspondence matching process that links the viewpoints i and k. Viewpoint i is treated as the left member of the pair, k as right member. For identification purposes they are indexed with superscript (l,r), respectively. The camera and image $(\mathbf{I_i}, \mathbf{P_i})$ is then rectified to $(\mathbf{I_i^l}, \mathbf{P_i^l})$, while $(\mathbf{I_k}, \mathbf{P_k})$ is rectified to $(\mathbf{I_k^r}, \mathbf{P_k^r})$.

3.3 Constrained Matching

For dense correspondence matching a disparity estimator based on the dynamic programming scheme of Cox et al. [3], is employed that incorporates the above mentioned constraints. It operates on rectified image pairs where the epipolar lines coincide with image scan lines. The matcher searches at each pixel in image $\mathbf{I_i^r}$ for maximum normalized cross correlation in $\mathbf{I_k^l}$ by shifting a small measurement window (kernel size 5x5 or 7x7) along the corresponding scan line. The selected search step-size ΔD (usually 1 pixel) determines the search resolution. Matching ambiguities are resolved by exploiting the ordering constraint in the

dynamic programming approach [16]. The algorithm was further adapted to employ extended neighborhood relationships and a pyramidal estimation scheme to reliably deal with very large disparity ranges of over 50% of the image size [4]. The estimate is stored in a disparity map $D_{(i,k)}$ with one of the following values:

– a valid correspondence $\mathbf{x}^{\mathbf{l}}_{\mathbf{k}} = D_{(i,k)}[\mathbf{x}^{\mathbf{r}}_{\mathbf{i}}]$,
– an undetected search failure which leads to an outlier,
– a detected search failure with no correspondence.

The matching algorithm was tested extensively on different scenes under laboratory and real outdoor conditions. To verify ground truth the system estimates were compared with synthetic data of known ground truth and with a high resolution active scanning system that projects coded light stripe patterns on the scene. It was found that the relative depth error of the matching system ranged between 0.3% to 1%. For further details see [16].

4 Multi Viewpoint Linking

The pairwise disparity estimation allows to compute image to image correspondence between adjacent rectified image pairs, and independent depth estimates for each camera viewpoint. An optimal joint estimate will be achieved by fusing all independent estimates into a common 3D model. The fusion can be performed in an economical way through controlled correspondence linking as described in this contribution. The approach utilizes a flexible multi viewpoint scheme by combining the advantages of small baseline and wide baseline stereo.

As *small baseline stereo* we define viewpoints where the baseline is much smaller than the observed average scene depth. This configuration is usually valid for image sequences were the images are taken as a spatial sequence from many slightly varying view points. The advantages (+) and disadvantages (–) are

+ easy correspondence estimation, since the views are similar,
+ small regions of viewpoint related occlusions[4],
– small triangulation angle, hence small depth accuracy.

The *wide baseline stereo* in contrast is used mostly with still image photographs of a scene where few images are taken from a very different viewpoint. Here the depth accuracy is better but correspondence and occlusion problems appear

– difficult correspondence estimation, since the views are not similar,
– large regions of viewpoint related occlusions,
+ big triangulation angle, hence high depth accuracy.

The *multi viewpoint linking* combines the virtues of both approaches. In addition it will produce denser depth maps than either of the other techniques,

[4] As view point related occlusions we consider those parts of the object that are visible in one image only, due to object self-occlusion.

and allows additional features for depth and texture fusion. Advantages of *multi viewpoint linking* are

- very dense depth maps for each viewpoint,
- no viewpoint dependent occlusions,
- high depth accuracy through viewpoint fusion,
- texture enhancement through texture fusion.

4.1 Correspondence Linking Algorithm

The correspondence linking is described in the following section. It concatenates corresponding image points over multiple viewpoints by correspondence tracking over adjacent image pairs.

Consider an image sequence taken from $i = [1, N]$ viewpoints with camera projection matrices calibrated as described in Sect. 2. Assume that the sequence is taken by a camera moving sideways while keeping the object in view. For any view point i let us consider the image triple $[I_{i-1}, I_i, I_{i+1}]$. The image pairs (I_{i-1}, I_i) and (I_i, I_{i+1}) form two stereoscopic image pairs with correspondence estimates as described above.

We can now create two chains of correspondence links for an image point $\mathbf{x_i}$, one up and one down the image index i.

Upwards linking: $\mathbf{x_{i+1}} = (\mathbf{H^r_{i+1}})^{-1} D_{(i,i+1)}[\mathbf{H^l_i x_i}]$,
Downwards linking: $\mathbf{x_{i-1}} = (\mathbf{H^l_{i-1}})^{-1} D_{(i,i-1)}[\mathbf{H^r_i x_i}]$.

$\mathbf{H}$ denotes the transformation for image rectification and $D_{(i,k)}$ the correspondence between the rectified images. The linking process is repeated along the image sequence to create a chain of correspondences upwards and downwards. Every correspondence link requires 2 mappings and 1 disparity lookup. Throughout the sequence of N images, $2(N-1)$ disparity maps are computed. The multi viewpoint linking is then performed efficiently via fast lookup functions on the pre-computed estimates.

Occlusions and Visibility. In a triangulation sensor with two viewpoints i and k two types of occlusion occur. If parts of the object are hidden in both viewpoints due to object self-occlusion, then we speak of *object occlusions* which cannot be resolved from this viewpoint. If a surface region is visible in viewpoint i but not in k, we speak of a *shadow occlusion*. The regions have a shadow-like appearance of undefined disparity values since the occlusions at view k cast a shadow on the object as seen from view i. Shadow occlusions are in fact detected by the uniqueness constraint discussed in Sect. 3. A solution to avoid shadow occlusions is to incorporate a symmetrical multi viewpoint matcher as proposed in this contribution. Points that are shadowed in the (right) view $i+1$ are normally visible in the (left) view $i-1$ and vice versa. The exploitation of up-and down-links will resolve for the shadow occlusions. A helpful measure in this context is the visibility V that defines for a pixel in view i the maximum number of possible correspondences in the sequence. $V = 1$ is caused by a shadow occlusion, $V >= 2$ allows a depth estimate.

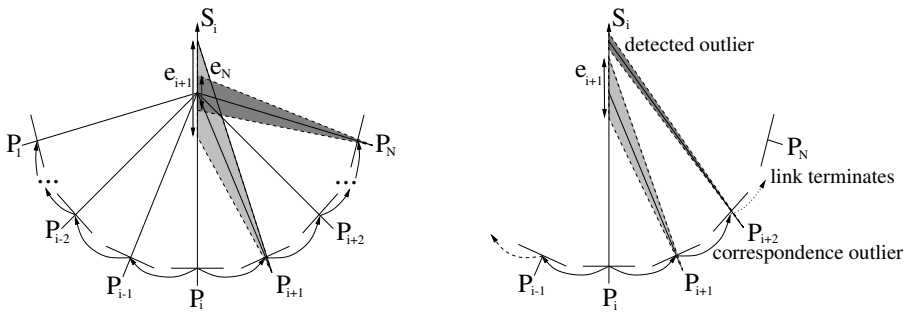

Fig. 3. Left: Depth fusion and uncertainty reduction from correspondence linking. Right: detection of correspondence outliers by depth interval testing.

Depth Estimation and Outlier Detection. Care must be taken to exclude invalid disparity values or outliers from the chain. If an invalid disparity value is encountered, the chain is terminated immediately. Outliers are detected by the statistics of the depth estimate computed from the correspondences. Inliers will update the depth estimate using a 1-D Kalman filter.

Depth and Uncertainty. Consider a 3D surface point $\mathbf{X}$ that is projected onto its corresponding image points $\mathbf{x_i} = \mathbf{P_i X}, \mathbf{x_k} = \mathbf{P_k X}$. The inverse process holds for triangulating $\mathbf{X}$ from the corresponding point pair $(\mathbf{x_i}, \mathbf{x_k})$. We can in fact exploit the calibrated camera geometry and express the 3D point $\mathbf{X}$ as a depth value d_x along the known line of sight $\mathbf{S_{xi}}$ that extends from the camera projection center through the image correspondence $\mathbf{x_i}$. Triangulation computes the depth as the length of $\mathbf{S_{xi}}$ between the camera projection center and the locus of minimum distance between the corresponding lines of sight. The triangulation is computed for each image point and stored in a depth map associated with the viewpoint.

The depth for each reference image point $\mathbf{x_i}$ is improved by the correspondence linking that delivers two lists of image correspondences relative to the reference, one linking down from $i \to 1$ and one linking up from $i \to N$. For each valid corresponding point pair $(\mathbf{x_i}, \mathbf{x_k})$ we can triangulate a consistent depth estimate $d(x_i, x_k)$ along $\mathbf{S_{xi}}$ with e_k representing the depth uncertainty. Figure 3(left) visualizes the decreasing uncertainty interval during linking. While the disparity measurement resolution ΔD in the image is kept constant (at 1 pixel), the reprojected depth error e_k decreases with the baseline.

Outlier Detection and Inlier Fusion. As measurement noise we assume a contaminated Gaussian distribution with a main peak within a small interval (of 1 pixel) and a small percentage of outliers. Inlier noise is caused by the limited resolution of the disparity matcher. Outliers are undetected correspondence failures and may be arbitrarily large. As threshold to detect the outliers we utilize the depth uncertainty interval e_k. The detection of an outlier at k terminates the linking at $k-1$. All depth values $[d_i, d_{i+1}, ..., d_{k-1}]$ are inlier depth values

that fall within the uncertainty interval around the mean depth estimate. They are fused by a simple 1-D kalman filter to obtain a mean depth estimate.

Figure 3 (right) explains the outlier selection and link termination. The outlier detection scheme is not optimal since it relies on the position of the outlier in the chain. Valid correspondences behind the outlier are not considered any more. It will, however, always be as good as a single estimate and in general superior to it. In addition, since we process bidirectionally up- and down-link, we always have two correspondence chains to fuse which allows for one outlier per chain.

4.2 Texture Enhancement

The correspondence linking builds a controlled chain of correspondences that can be used for texture enhancement as well. At each reference pixel one may collect a sorted list of image color values from the corresponding image positions. This allows to enhance the original texture in many ways by accessing the color statistics. Some features that are derived naturally from the linking algorithm are:

1. **Highlight and reflection removal**: A median or robust mean of the corresponding texture values is computed to discard imaging artifacts like sensor noise, specular reflections and highlights [20].
2. **Super-resolution texture**: The correspondence linking is not restricted to pixel-resolution, since each sub-pixel-position in the reference image can be used to start a correspondence chain. The correspondence values are obtained from the disparity map through interpolation. The object is viewed with a camera of limited pixel resolution but from many slightly displaced viewpoints. This can be exploited to create super-resolution texture by fusing all images on a finer resampling grid [13].
3. **Best view selection for highest texture resolution**: For each surface region around a pixel the image which has the highest possible texture resolution is selected, based on the object distance and viewing angle. The composite image takes the highest possible resolution from all images into account.

A detailed discussion of texture fusion is out of the scope of this contribution but we will give some examples of it in the experiment section.

5 3D Surface Modeling

The dense depth maps as computed by the correspondence linking need to be approximated by a 3D surface representation suitable for visualization. So far each object point was treated independently. To achieve spatial coherence and a connected surface, the depth map is spatially interpolated using a parametric surface model. We employ a bounded thin plate model with a second order spline

to smooth the surface and to interpolate small surface gaps in regions that could not be measured. This spatially smoothed surface is then approximated by a triangular wire frame to reduce geometric complexity and tailor the model for visualization [15]. Texture mapping of the texture onto the wire frame model greatly enhances the realism of the models which are stored in VRML file format for easy exchange with visualization systems.

The mesh triangulation currently utilizes the reference view only to build the model. It can deal with shadow occlusions but not with true 3D object occlusions. In a next step a viewpoint independent mesh generation will be implemented to allow closure of the object surface. We will incorporate a surface registration scheme based on the Iterated Closest Point algorithm ICP [2] to fit surfaces from different view points. The surface parts that are overlapping are then retriangulated to allow for one consistent surface mesh [14].

Another problem to be tackled is the fusion of different projective frames into a global coordinate system. Often it is not possible to record a contiguous video stream of an extended object where all views are calibrated in the same projective frame. When moving around or inside of a building, for example, scene cuts are inevitable and the system will generate an independent calibration for each scene cut. Registration of the independent frames consists of 3D position and scale adaptation. We are developing an interactive 3D interface where the user can easily register those frames.

6 Experiments

In this section the performance of the algorithm is tested on the two outdoor sequences *Castle* and *Fountain*.

6.1 Castle Sequence

The *Castle* sequence consists of 22 images of 720x576 pixel resolution taken with a standard semi-professional camcorder that was moved freely in front of a building. Figure 4 shows results from camera tracking. Four of the images (left) and the estimated 3D-structure of the building with calibrated camera positions are displayed from a front view (right). The rectangular appearance of the building, the regular spacing of camera positions, and measurements on the reconstructed surface [23] confirm the metric qualities of the calibration.

To judge the geometric and visual quality of the reconstruction, different perspective views of the model were computed and displayed in Fig. 5. In the shaded view (left), the geometric details like the window and door niches are seen. A close-up look from a position that a human observer would take reveals the high visual quality of the model (center). To demonstrate the texture fusion capabilities of the algorithm, the specular reflection in the upper right window was removed by a texture median filtering and a super-resolution texture with zoom factor of 4 was generated from the image sequence (right). The region shows the reference image (above) and the generated median super-resolution texture without reflection (below).

Fig. 4. Left: 4 of 22 images of the castle which were used for the reconstruction. Right: a view of the 3D model with feature points superimposed in black and the calibrated camera positions visualized as pyramids.

Performance Evaluation for the Castle Sequence. The above reconstructions showed some qualitative results. The quantitative performance of correspondence linking can be tested in different ways. One measure already mentioned is the visibility of an object point. In connection with correspondence linking, we have defined visibility V as the number of views linked to the reference view. Another important feature of the algorithm is the density and accuracy of the depth maps. To describe its improvement over the 2-view estimator, we define the fill rate F and the average relative depth error E as additional measures.

Fig. 5. Left: shaded view. Center: close-up view. Right: 4x zoomed original region (above), generation of median-filtered super-resolution texture (below).

Visibility $V[views]$: average number of views linked to the reference image.
Fill Rate $F[\%]$: $\frac{\text{Number of valid pixels}}{\text{Total number of pixels}}$
Depth error $E[\%]$: relative depth error e_d for all valid pixels.

The 2-view disparity estimator is a special case of the proposed linking algorithm, hence both can be compared on an equal basis. The 2-view estimator operates on the image pair $(i, i+1)$ only, while the multi view estimator operates on a sequence $1 < i < N$ with $N >= 3$. The above defined statistical measures were computed for different sequence lengths N. Figure 6 displays visibility and relative depth error for sequences from 2 to 15 images, chosen symmetrically around the reference image. The average visibility V shows that for up to 5 images nearly all views are utilized. For 15 images, at average 9 images are linked. The amount of linking is reflected in the relative depth error that drops from 5% in the 2 view estimator to about 1.2% for 15 images.

Linking two views is the minimum case that allows triangulation. To increase the reliability of the estimates, a surface point should occur in more than two images. We can therefore impose a minimum visibility V_{min} on a depth estimate. This will reject unreliable depth estimates effectively, but will also reduce the fill-rate of the depth map.

The graphs in Fig. 6(center) show the dependency of the fill rate and depth error on minimum visibility for N=11. The fill rate drops from 92% to about 70%, but at the same time the depth error is reduced to 0.5% due to outlier rejection. The depth map and the relative error distribution over the depth map

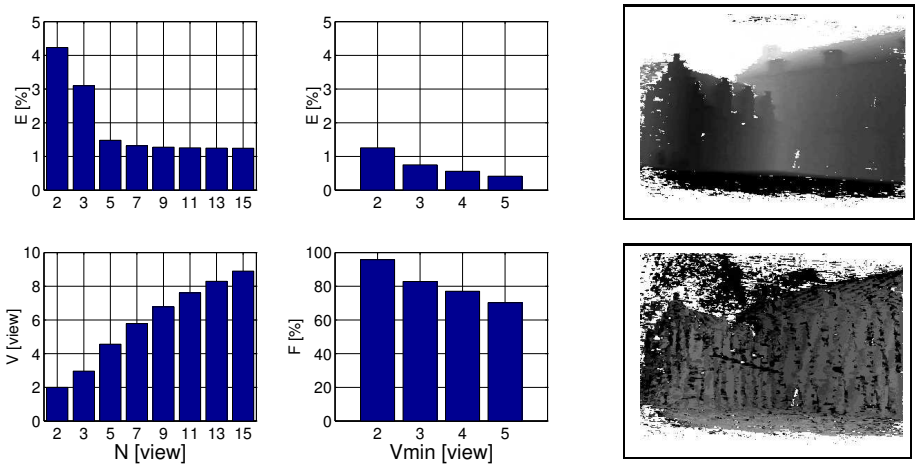

Fig. 6. Statistics of the castle sequence. Left: Influence of sequence length N on visibility V and relative depth error E. Center: Influence of minimum visibility V_{min} on fill rate F and depth error E for $N = 11$. Right: Depth map (above: dark=near, light=far) and error map (below: dark=large error, light=small error) for $N = 11$ and $V_{min} = 3$.

is displayed in Fig. 6(right). The error distribution shows a periodic structure that in fact reflects the quantization uncertainty of the disparity resolution when it switches from one disparity value to the next.

6.2 Fountain Sequence

The *Fountain* sequence consists of 5 images of the back wall of a fountain at the archaeological site of Sagalassos in Turkey, taken with a digital camera with 573x764 pixel resolution. It shows a concavity in which once a statue was situated. Figure 7 shows from left to right images 1 and 3 of the sequence, the depth map as computed with the 2-view estimator, and the depth map when using all 5 images. The white (undefined) regions in the 2-view depth map are due to shadow occlusions which are almost completely removed in the 5-view depth map. This is reflected in the fill rate that increases from 89 to 96%. It should be noted that for this sequence a very large search range of 400 pixels was used, which is over 70% of the image width. Despite this large search range only few matching errors occurred.

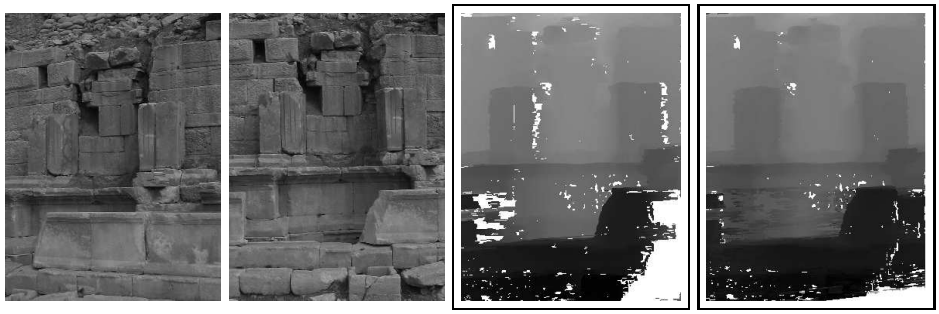

Fig. 7. Left: First and last image of sequence. Right: Depth maps from the 2-view and the 5-view estimator (from left to right) showing the very dense depth maps.

The performance characteristics are displayed in the Table 1. The fill rate is high and the relative error is rather low because of a fairly wide baseline between views. This is reflected in the high geometric quality of depth the map and the reconstruction.

Table 1. Statistics of fountain sequence for visibility V, fill rate F and depth error E.

N[view]	V[*views*]	F[%]	E[%]
2	2	89.8728	0.294403
3	2.85478	96.7405	0.208367
5	4.23782	96.4774	0.121955

The visual reconstruction quality for the fountain is displayed in Fig. 8. Even fine details like the relief carved into the stones are preserved. The side and top views of the overall model show the accurate and detailed structure due to the wide triangulation angle over the sequence, and the textured close-up view reveals a highly realistic sensation.

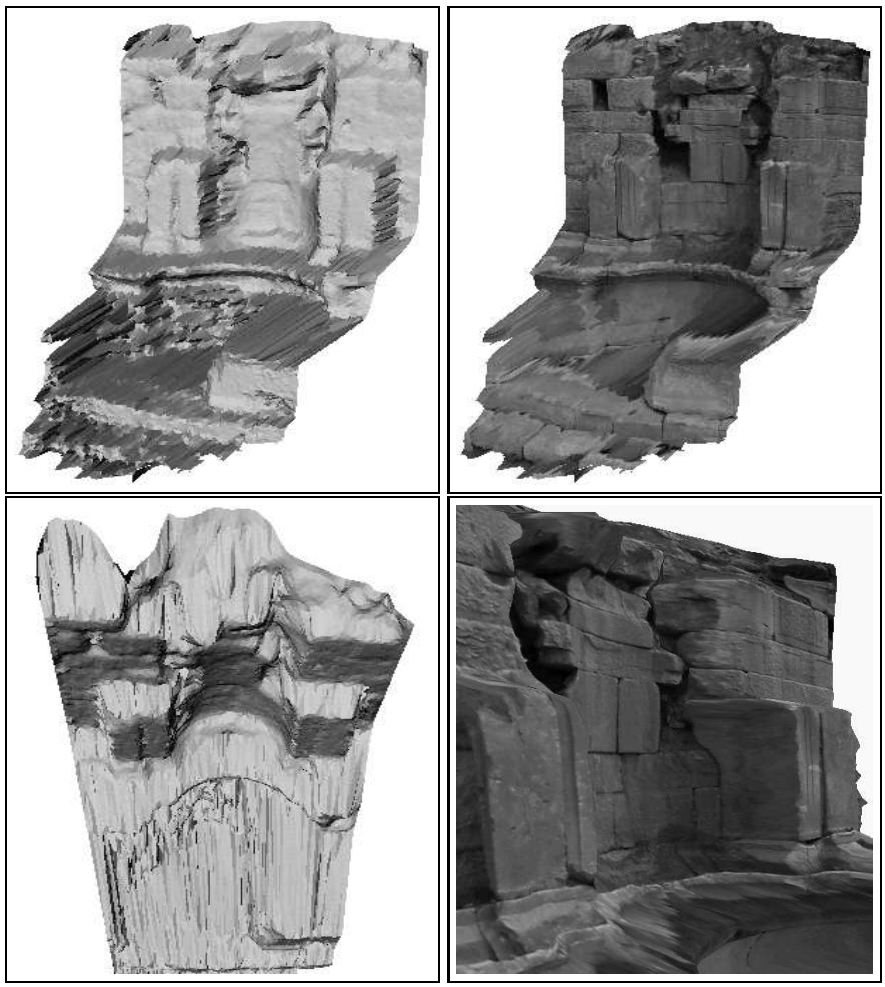

Fig. 8. Above: shaded and textured front view of model, Below: shaded top view and close-up of the textured model.

7 Conclusion

In this contribution we developed a correspondence linking scheme that computes dense and accurate depth maps based on the sequence linking of pairwise estimated disparity maps. The correspondence linking is the basic tool which allows to perform a variety of different operations on the image data. Depth and texture fusion, outlier detection and texture enhancement are some of the proposed applications.

The algorithm is embedded in a surface reconstruction system that allows for fully automatic generation of textured 3D surface models from image sequences, acquired with uncalibrated hand-held cameras. The performance analysis showed that very dense depth maps with fill rates of over 90% and a relative depth error of 0.1% can be measured with off-the-shelf cameras even in unrestricted outdoor environments such as an archaeological site.

Acknowledgments

We would like to thank Andrew Zisserman and his group from Oxford for supplying us with robust projective reconstruction software. I. Cox and L. Falkenhagen supplied versions of their disparity matching code. A specialization grant from the Flemish Institute for Scientific Research in Industry (IWT) and the financial support from the EU ACTS project AC074 VANGUARD are also gratefully acknowledged.

References

1. P. Beardsley, P. Torr and A. Zisserman: 3D Model Acquisition from Extended Image Sequences. In: B. Buxton, R. Cipolla(Eds.) Computer Vision - ECCV 96, Cambridge, UK., vol.2, pp.683-695. Lecture Notes in Computer Science, Vol. 1064. Springer Verlag, 1996.
2. Y. Chen and G. Medioni: Object Modeling by Registration of Multiple Range Images. Proceedings 1991 IEEE International Conference on Robotics and Automation, pp. 2724-2729, Sacramento (CA), 1991.
3. I. J. Cox, S. L. Hingorani, and S. B. Rao: A Maximum Likelihood Stereo Algorithm. Computer Vision and Image Understanding, Vol. 63, No. 3, May 1996.
4. L.Falkenhagen: Hierarchical Block-Based Disparity Estimation Considering Neighborhood Constraints. International Workshop on SNHC and 3D Imaging, September 5-9, 1997, Rhodes, Greece.
5. O.Faugeras: Three-Dimensional Computer Vision - a geometric viewpoint. MIT-Press, 1993.
6. O. Faugeras: What can be seen in three dimensions with an uncalibrated stereo rig. Proc. ECCV'92, pp.563-578.
7. O. Faugeras, Q.-T. Luong and S. Maybank: Camera self-calibration - Theory and experiments. Proc. ECCV'92, pp.321-334.
8. G.L.Gimel'farb: Symmetrical approach to the problem of automating stereoscopic measurements in photogrammetry. Cybernetics, 1979, 15(2), 235-247; Consultants Bureau, N.Y.

9. G.L.Gimel'farb and R.M.Haralick: Terrain reconstruction from multiple views. Proc. 7th Int. Conf. on Computer Analysis of Images and Patterns (CAIP'97), Kiel, Germany, Sept. 1997. (G.Sommer, K.Daniilidis, and J.Pauli, Eds.). Lecture Notes in Computer Science 1296, Springer: Berlin e.a., 1997, pp.694-701.
10. C.G. Harris and J.M. Pike: 3D Positional Integration from Image Sequences. 3rd Alvey Vision Conf, pp. 233-236, 1987.
11. R. Hartley: Estimation of relative camera positions for uncalibrated cameras. Proc. ECCV'92, pp.579-587.
12. A. Heyden and K. Åström: Euclidean Reconstruction from Image Sequences with Varying and Unknown Focal Length and Principal Point. Proc. CVPR'97.
13. M. Irani and S. Peleg: Super resolution from image sequences. Tenth International Conference on Pattern Recognition (Atlantic City, NJ, June 16-21, 1990), IEEE Catalog No. 90CH2898-5, 1990, subconference C, 115-120.
14. R. Koch: 3-D Modeling of Human Heads from Stereoscopic Image Sequences. Proc. DAGM'96, Informatik Aktuell, Springer Heidelberg, Germany, Sept. 1996.
15. R. Koch: Surface Segmentation and Modeling of 3-D Polygonal Objects from Stereoscopic Image Pairs. Proc. ICPR'96, Vienna 1996.
16. R. Koch: Automatische Oberflächenmodellierung starrer dreidimensionaler Objekte aus stereoskopischen Bildfolgen. Ph.D. Thesis, Fortschr.-Ber. 10/499. Düsseldorf: VDI Verlag 1997.
17. A. Kochan: Eine Methodenbank zur Evaluierung von Stereo-Vision-Verfahren. Ph.D. Thesis, TU Berlin, June 1991.
18. L. Matthies, T. Kanade, and R. Szeliski: Kalman filter-based algorithms for estimating depth from image sequences. International Journal of Computer Vision, vol. 3, 1989, 209-236.
19. M. Okutomi and T. Kanade: A multiple-baseline stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15, no. 4, 1993, 353-363.
20. E. Ofek, E. Shilat, A. Rappoport, M. Werman: Highlight and Reflection-Independent Multiresolution Textures from Image Sequences. IEEE Computer Graphics and Applications vol. 17 (2), March-April 1997.
21. M. Pollefeys, L. Van Gool and M. Proesmans: Euclidean 3D Reconstruction from Image Sequences with Variable Focal Lengths. Proc. ECCV'96, vol.1, pp. 31-42.
22. M. Pollefeys and L. Van Gool: Self-calibration from the absolute conic on the plane at infinity. Proc. CAIP'97.
23. M. Pollefeys, R. Koch and L. Van Gool: Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Internal Camera Parameters. Proc. ICCV'98, Bombay, India, 1998.
24. P.H.S. Torr: Motion Segmentation and Outlier Detection. PhD thesis, Dept. Eng. Science, University of Oxford, UK, 1995.
25. B. Triggs: The Absolute Quadric. Proc. CVPR'97.
26. Z. Zhang and O. Faugeras: 3D Dynamic Scene Analysis. Springer Verlag, 1992.