Visual modeling: from images to images

Marc Pollefeys and Luc Van Gool

Center for Processing of Speech and Images,

K.U.Leuven, Belgium

Abstract

This paper contains two parts. In the first part an automatic processing pipeline is presented that analyses an image sequence and automatically extracts camera motion, calibration and scene geometry. The system combines state-of-the-art algorithms developed in computer vision, computer graphics and photogrammetry. The approach consists of two stages. Salient features are extracted and tracked throughout the sequence to compute the camera motion and calibration and the 3D structure of the observed features. Then a dense estimate of the surface geometry of the observed scene is computed using stereo matching. The second part of the paper discusses how this information can be used for visualization. Traditionally, a textured 3D model is constructed from the computed information and used to render new images. Alternatively, it is also possible to avoid the need for an explicit 3D model and to obtain new views directly by combining the appropriate pixels from recorded views. It is interesting to note that even when there is an ambiguity on the reconstructed geometry, correct new images can often still be generated.

Keywords: 3D reconstruction, structure from motion, 3D modeling, image-based rendering.

Introduction

Nowadays computer graphics allow to render complex 3D scenes in real-time. Therefore, more and more demand exists for detailed representations of the 3D world. Producing this content using interactive 3D modeling packages has become very expensive and time consuming. In addition, in many cases real world objects or scenes are considered. This has motivated researchers to develop techniques to capture the 3D visual world directly. One of the most promising approaches consists of using images for this purpose.

In the field of computer vision, researchers have been working for many years to obtain 3D representations of scenes from images. Initially this work was targeted towards robotics and automation, e.g. allowing a robot to navigate through an unknown environment. In recent years the focus has shifted to visualization and communication, resulting in much more interaction with the computer graphics community. One of the main focuses has been to provide algorithms that can automatically extract the necessary information from multiple images. In addition, over the last ten years important new insights have been gained in the geometry of multiple images, allowing more flexible approaches to be developed (a good reference for this is the recent book by Hartley and Zisserman [15]).

The first part of this paper presents an automatic processing pipeline that we have been developing over the last few years [25, 27, 31, 33]. Starting from an image sequence the system gradually recovers a detailed 3D reconstruction. Both motion and calibration of the camera are

retrieved automatically during the processing. To achieve this the system combines state-of-the-art algorithms developed in computer vision, computer graphics and photogrammetry.

In the second part of this paper we discuss how the computed information can be used for visualization. A first approach consists of constructing a textured 3D model so that new images can be generated using the standard computer graphics 3D rendering pipeline. However, since a few years alternative approaches have been proposed that generate new images by recombining pixels of recorded images [22, 11]. Therefore, a second approach is presented that allows to generate new images without the need for an explicit 3D model [19, 20]. This approach renders new views directly from the recorded images. Although an explicit 3D model is not required, approximate depth information allows to minimize rendering artefacts. Another interesting aspect of image-based visualization -especially compared to obtaining measurements from images- is that in many cases ambiguities on the reconstruction do not show up during visualization. If the camera motion does not allow self-calibration to yield a unique result (due to the problem of critical motion sequences [37]) correct images can still be rendered under some conditions [34].

Image to 3D processing pipeline

Our processing pipeline starts from a sequence of images and computes all the necessary information to build a 3D model or to perform other types of rendering of the observed scene. The process gradually retrieves more and more information about the scene and about the camera.

First, the relative motion between consecutive images needs to be recovered. This process goes hand in hand with finding corresponding image features between these images (i.e. image

points that originate from the same 3D feature). The next step consists of recovering the motion and calibration of the camera and the 3D structure of the features. This process is done in two phases. At first the reconstruction contains a projective skew (i.e. parallel lines are not parallel, angles are not correct, relative distances are not preserved, etc.). This is due to the absence of an a priori calibration. Using a self-calibration algorithm [30] this distortion can be removed, yielding a reconstruction equivalent to the original scene up to a global scale factor. This *uncalibrated* approach to 3D reconstruction allows much more flexibility in the acquisition process since the focal length and other intrinsic camera parameters do not have to be measured –calibrated– beforehand and are allowed to change during the acquisition.

The reconstruction obtained as described in the previous paragraph only contains a sparse set of 3D points (only a limited number of features are considered at first). Although interpolation might be a solution, this typically yields models with poor visual quality. Therefore, the next step consists in an attempt to match all image pixels of an image with pixels in neighboring images, so that these points too can be reconstructed. This task is greatly facilitated by the knowledge of all the camera parameters which we have obtained in the previous stage. Since a pixel in the image corresponds to a ray in space and the projection of this ray in other images can be predicted from the recovered pose and calibration, the search of a corresponding pixel in other images can be restricted to a single line. Additional constraints such as the assumption of a piecewise continuous 3D surface are also employed to further constrain the search. It is possible to warp the images so that the search range coincides with the horizontal scanlines. An algorithm that can achieve this for arbitrary camera motion is described in [28]. This allows to use an efficient stereo algorithm that computes an optimal match for the whole scanline at once [43]. Thus, we

can obtain a depth estimate (i.e. the distance from the camera to the object surface) for almost every pixel of an image. By fusing the results of all the images together a complete dense 3D surface model is obtained. The images used for the reconstruction can also be used for texture mapping so that a final photo-realistic result is achieved. The different steps of the process are illustrated in Figure 1. In the following paragraphs some of the critical steps are described in some more detail.

Relating images

Starting from a collection of images or a video sequence the first step consists in relating the different images to each other. This is not an easy problem. A restricted number of corresponding points is sufficient to determine the geometric relationship or *multi-view constraints* between the images. Since not all points are equally suited for matching or tracking (e.g. a pixel in a homogeneous region), the first step consist of selecting feature points [12, 35]. These are suited for tracking or matching. Depending on the type of image data (i.e. video or still pictures) the feature points are tracked or matched and a number of potential correspondences are obtained. From these the multi-view constraints can be computed. However, since the correspondence problem is an ill-posed problem, the set of corresponding points can be contaminated with an important number of wrong matches or *outliers*. In this case, a traditional least-squares approach will fail and therefore a robust method is used [40, 10]. Once the multi-view constraints have been obtained they can be used to guide the search for additional correspondences. These can then be employed to refine the results for the multi-view constraints further.

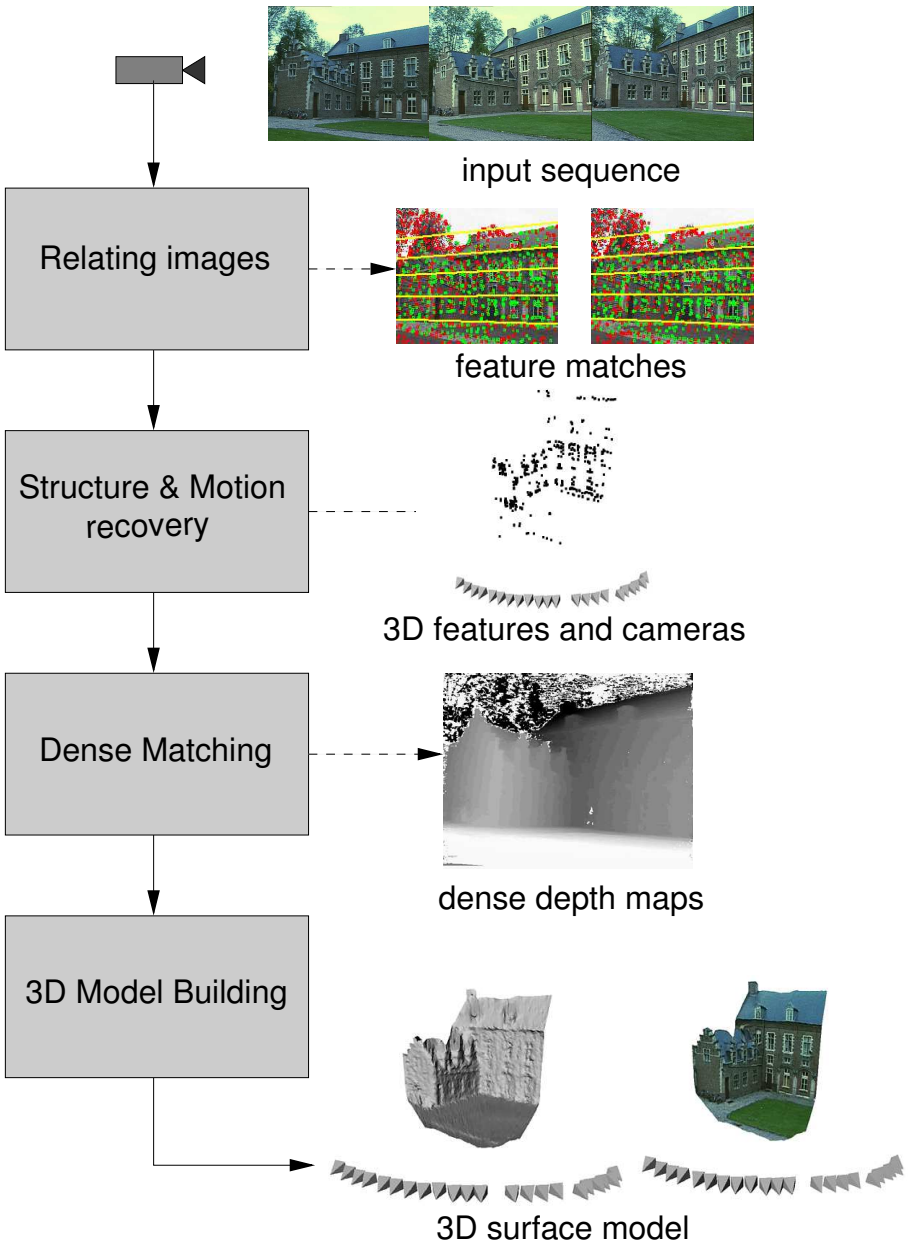

Figure 1: Overview of our 3D recording pipeline.

Structure and motion recovery

The relation between the views and the correspondences between the features, retrieved as explained in the previous section, will be used to retrieve the structure of the scene and the motion of the camera. The approach that is used is related to [1] but is fully projective and therefore not dependent on the quasi-euclidean initialization. This is achieved by strictly carrying out all measurements in the images, i.e. using reprojection errors instead of 3D errors.

At first two images are selected and an initial projective reconstruction frame is set-up [8, 13]. Then the pose of the camera for the other views is determined in this frame and for each additional view the initial reconstruction is refined and extended. In this way the pose estimation of views that have no common features with the reference views also becomes possible. Once the structure and motion has been determined for the whole sequence, the results can be refined through a projective bundle adjustment [42]. Then the ambiguity is restricted to metric (i.e. Euclidean, but with unknown scale) through self-calibration [9]. Our approach is based on the concept of the absolute quadric [41, 30]. Finally, a metric bundle adjustment is carried out to obtain an optimal estimation of the structure and motion.

In some cases it can happen that if the motion is not sufficiently general an ambiguity persists [37]. However, we will see further on that even if this problem occurs it is often still possible to generate correct new views.

Dense surface estimation

To obtain a more detailed model of the observed surface dense matching is used. The structure and motion obtained in the previous steps can be used to constrain the correspondence search. Since the calibration between successive image pairs was computed, the epipolar constraint that restricts the correspondence search to a 1-D search range can be exploited. Image pairs are warped so that epipolar lines coincide with the image scan lines. For this purpose the rectification scheme proposed in [28] is used. This approach can deal with arbitrary relative camera motion while standard homography-based approaches fail when the epipole is contained in the image. The approach also guarantees minimal image sizes. The correspondence search is then reduced to a matching of the image points along each image scan-line. This results in a dramatic increase of the computational efficiency of the algorithms by enabling several optimizations in the computations. A first example comes from the *castle* sequence. In Figure 2 an image pair and the associated rectified image pair are shown.

In addition to the epipolar geometry other constraints like preserving the order of neighboring pixels and bidirectional uniqueness of the match can be exploited. These constraints are used to guide the correspondence towards the most probable scan-line match using a dynamic programming scheme [4]. The matcher searches at each pixel in one image for maximum normalized cross correlation in the other image by shifting a small measurement window along the corresponding scan-line. Matching ambiguities are resolved by exploiting the ordering constraint in the dynamic programming approach. The algorithm was further adapted to employ a pyramidal estimation scheme to reliably deal with very large disparity ranges of over 50% of image size [7].

More details on our stereo algorithm can be found in [43]. The disparity search range is limited based on the disparities that were observed for the features in the structure and motion recovery.

The pairwise disparity estimation allows to compute image to image correspondences between adjacent rectified image pairs and independent depth estimates for each camera viewpoint. An optimal joint estimate is achieved by fusing all independent estimates into a common 3D model using a Kalman filter. The fusion can be performed in an economical way through controlled correspondence linking. This approach was discussed more in detail in [18].

This approach combines the advantages of small baseline and wide baseline stereo. It can provide a very dense depth map by avoiding most occlusions. The depth resolution is increased through the combination of multiple viewpoints and large global baseline while the matching is simplified through the small local baselines.

Constructing visual models

The system described in the previous section computes depth maps for every view as well as the motion and calibration of the camera. This yields all the necessary information to build photo-realistic visual models.

3D surface reconstruction

The traditional approach consists of approximating the 3D surface by a triangular mesh to reduce geometric complexity and to tailor the model to the requirements of computer graphics visualization systems. A simple approach consists of overlaying a regular 2D triangular mesh

on top of one of the images and then build a corresponding 3D mesh by placing the vertices of the triangles in 3D space according to the values found in the corresponding depth map. The image itself is used as a texture map. If no depth value is available or the confidence is too low the corresponding triangles are not reconstructed. The same happens when triangles are placed over discontinuities. This approach works well on dense depth maps obtained from multiple stereo pairs. To reduce the number of polygons without significantly reducing the quality of the model a mesh simplification algorithm can be used [36]. The texture itself can also be enhanced through the multi-view linking scheme [18]. A median or robust mean of the corresponding texture values can be computed to discard imaging artifacts like sensor noise, specular reflections and highlights [24].

To reconstruct more complex shapes it is necessary to combine multiple depth maps. Since all depth-maps can be located in a single metric frame, registration is not an issue. In some cases it can be sufficient to load the separate models together in the graphics system. In general, however, better results are obtained by integrating the different meshes into a single mesh. This can for example be done using the volumetric technique proposed in [5]. Note that in this case also the texture has to be obtained by combining different images. The approach we use selects a view for each vertex (based on average normal and visibility) and then generates the texture by blending between the different views selected for each triangle.

Examples We have recorded a short video sequence from a medusa head decorating an ancient fountain in Sagalassos (an ancient city in Turkey). The 20 second video sequence was recorded with a hand-held consumer video camera (Sony TRV-900). Each twentiest frame was used as

a key-frame by our video to 3D processing pipeline. Three of these frames are seen on the top part of Figure 3. The compute structure and motion is also seen in this figure (middle-left). The camera viewpoints are represented by small pyramids. The depth map used to construct the 3D model is seen on the middle-right of the figure. Finally, the model -with and without texture- is seen at the bottom of the figure. From the shaded model one can see that most of the geometric detail is accurately recovered. By using the image itself as texture map a photorealistic model is obtained. Note from the rightmost view that the 3D model allows to render realistic views that are very different from the original views.

The second example was also recorded on the archaeological site of Sagalassos. In this case the remains of a Roman villa were recorded at different stages during the excavations. Here we consider a specific layer for which 26 pictures were taken with a hand-held photo camera (Nikon F50) and scanned to PhotoCD. The on site acquisition of the images only takes a few minutes so it does not slow down the excavation process. Some of the recorded pictures can be seen on the top part of Figure 4. Note that in this case the geometry of the observed scene is too complex to be reconstructed from a single depth map. Therefore, in this case the 3D model was obtained by combining all the depth maps using a volumetric approach. More details on archaeological applications of our techniques can be found in [26].

Lightfield rendering

For rendering new views two major concepts are known in literature. The first one is the geometry based concept. The scene geometry is reconstructed from a stream of images and a single

texture is synthesized which is mapped onto this geometry. For this approach, a limited set of camera views is sufficient, but view-dependent effects such as specularities can not be handled appropriately. This approach was discussed in the previous section. The second major concept is lightfield rendering. This approach models the scene as a collection of views all around the scene without an exact geometrical representation [22]. New (virtual) views are rendered from the recorded ones by interpolation. Optionally approximate geometrical information can be used to improve the results [11]. It was shown that this can greatly reduce the required amount of images [3]. There are also several intermediate representations that combine view-dependent texture with an explicit 3D surface model, such as view-dependent texture mapping [6] and surface lightfields [44]. The approach presented in this paper allows to render views directly from the calibrated sequence of recorded images with use of local depth maps. The original images are directly mapped onto one or more planes viewed by a virtual camera.

To obtain a high-quality image-based scene representation, we need many views from a scene from many directions. For this, we can record an extended image sequence moving the camera in a zigzag like manner. To obtain a good quality structure-and-motion estimation from this type of sequence and reduce error accumulation it can be important to also match close views that are not predecessors or successors in the image stream [19].

The simplest approach consists of approximating the scene geometry by a single plane. The mapping from a recorded image to a new view or vice-versa then corresponds to a homography. To construct a specific view it is best to interpolate between neighboring views. The color value for a particular pixel can thus best be obtained from those views whose projection center is close to the viewing ray of this pixel or, equivalently, project closest to the specified pixel. For sim-

plicity the support is restricted to the nearest three cameras (see Figure 5). All camera centers are projected into the virtual image and a 2D triangulation is performed. The cameras corresponding to the corners of a triangle then contribute to all pixels inside the triangle. The color values are blended using the baricentric coordinates on the triangle as weights. The total image is built up as a mosaic of these triangles. Although this technique assumes a very sparse approximation of geometry, the rendering results show only small ghosting artifacts (see experiments).

The results can be further improved. It is possible to use a different approximating plane for each triangle. This improves the accuracy further as the approximation is not done for the whole scene but just for that part of the image which is seen through the actual triangle. The 3D position of the triangle vertices can be obtained by looking up the depth value for the projection of the virtual viewpoint in the depth map correspondiong to each vertex. These points can be interpreted as the intersections of the lines connecting the virtual viewpoint and the real viewpoints with the scene geometry. Knowing the 3D coordinates of triangle corners, we can define a plane through them and apply the same rendering technique as described above.

Finally, if the triangles exceed a given size, they can be subdivided into four sub-triangles. For each of these sub-triangles, a separate approximative plane is calculated in the above manner. Of course, further subdivision can be done in the same way to improve accuracy. Especially, if just a few triangles contribute to a single virtual view, this subdivision is generally necessary. It should be done in a resolution according to performance demands and to the complexity of the geometry. Rendering can be performed in real-time using alpha blending and texture mapping facilities of todays graphics hardware. More details on this approach can be found in [21, 19, 16]. A similar approach was presented recently [2].

Example We have tested our approaches with an image sequence of 187 images showing an office scene. Figure 6 (top-left) shows one particular image. A digital consumer video camera (Sony TRV-900) was swept freely over a cluttered scene on a desk, covering a viewing surface of about $1m^2$. Figure 6 (top-right) shows the calibration result. Result of a rendered view are shown in the middle of the figure. The image on the left is rendered with a planar approximation while the image on the right was generated with two levels of subdivision. Note that some ghosting artefacts are visible for the planar approximation, but not for the more detailed approximation. It is also interesting to note that most ghosting occures in the vertical direction because the inter-camera distance is much larger in this direction. In the lower part of Figure 6 a detail of a view is shown for the different methods. In the case of one global plane (left image), the reconstruction is sharp where the approximating plane intersects the actual scene geometry. The reconstruction is blurred where the scene geometry diverges from this plane. In the case of local planes (middle image), at the corners of the triangles the reconstruction is almost sharp, because there the scene geometry is considered directly. Within a triangle, ghosting artifacts occur where the scene geometry diverges from the particular local plane. If these triangles are subdivided (right image) these artifacts are reduced further.

Rendering ambiguous reconstructions

When totally uncalibrated cameras are used, it is only possible to recover the structure of the scene up to an arbitrary projective transformation [8, 13]. However, if some constraints on the intrinsic camera parameters are available it is possible to reduce this ambiguity to metric. This

is in general done through self-calibration. In recent years many different methods have been proposed. Some are based on the assumptions that the intrinsics do not change during acquisition (e.g. [9, 29, 41]). Other methods relax the constraint of constant intrinsics but require the knowledge of one or more intrinsic parameters (e.g. [30]). It was proven that for sufficiently general motion the knowledge that pixels are rectangular is sufficient to allow for successful self-calibration [30].

In practice, however, the motion of the camera is often restricted and there remains an ambiguity on the reconstruction. This is known as the problem of critical motion sequences (CMS). It was first discussed by Sturm [37] and further studied in [17, 23, 38, 32]. Depending on the constraints available for self-calibration different classes of motions can be identified as critical. For each of these classes a specific ambiguity remains on the reconstruction. For the constraint of constant intrinsics camera parameters the most important CMS classes are pure translation, pure rotation, orbital motion and planar motion [37]. If the constraints are that all intrinsics are known except for the focal length that can freely vary, the most important cases are forward motion, pure rotation, translation and rotation about the optical axis and hyperbolic and/or elliptic motion [38].

It depends on the application whether some ambiguity is acceptable or not. There are two main classes of applications for 3D reconstructions from images. The first one consists of metrology applications and in most cases no ambiguity can be tolerated. The second class of applications consists of visualization. In this case the goal is to generate novel views based on original images. Considering this application, the important point is not the correctness of the reconstruction, but the correctness of the novel views that are generated from it.

This problem was addressed in [34] and also partially in [23]. Here we will discuss the results we have obtained in [34]. In that paper we have derived a theorem that allows us to conclude that it is possible to generate correct new views (i.e. with no observable distortion), even starting from an ambiguous reconstruction. In this case, we should, however, restrict the motion of the virtual camera to the type of the CMS recovered in the reconstruction. For example, if a model was acquired by a camera with constant intrinsic parameters performing a planar motion on the ground plane and thus rotating around vertical axes, then we should not move the virtual camera outside this plane nor rotate around non-vertical axes. But, if we restrict our virtual camera to this critical motion in the virtual world, then all these motions will correspond to Euclidean motions of the original camera in the real world and no distortion will be present in the images. Note that the *recovered* camera parameters should be used (i.e. the ones obtained during the self-calibration process). This constraint can be relaxed when varying camera parameters are considered. In fact, this result is related to the more general rule that for the generation of new views interpolation is more desirable than extrapolation.

In fact, it is also possible to derive a practical approach that can characterize the expected ambiguity that could be observed in a particular novel view. For this purpose the self-calibration algorithm has to be run twice, once with the original sequence and once with the original sequence extended with the virtual camera. By comparing the uncertainty ellipsoids around the solution one can obtain an idea of the observable ambiguity. If the fact of adding the virtual camera largely reduces the undertainty ellipsoid, then an important ambiguity will be observable. If the uncertainty ellipsoid is left unchanged, then the potential ambiguity is unobservable from that specific viewpoint.

This approach was used to develop a special viewer that could warn the user if ambiguities might become apparent. In this case the background color would change from gree/light to red/dark. Using the self-calibration algorithm described in [29] on the castle sequence, used in the first part of this paper for illustration, a large uncertainty remained that corresponded to a scaling along the average optical axis. For purpose of illustration we distorted our model according to this uncertainty so that we could visually verify the predictions of the viewer. A few views are shown in Figure 7. It should be clear that even some views very far away from the originally recorded images can be rendered without risk of ambiguity (green/light views), while some others that are less far away are showing a lot of ambiguity (red/dark views).

Conclusion

In this paper we have presented an image processing pipeline that takes a video or image sequence as input and automatically computes camera motion and calibration, scene structure and depth maps from it. These results can then be used to generate different types of visual models. Explicit 3D models as well as lightfield representations can be computed and used for rendering. This approach integrates state-of-the-art algorithms in computer vision, computer graphics and photogrammetry. The approach was illustrated with a number of real examples. Finally, we discussed the possiblity of rendering novel views in the presence of an ambiguity on the 3D structure of the model. Our approach could for example be used to automatically optimize a fly-through in a virtual environment containing 3D models obtained from image sequences.

Acknowledgement

We would like to acknowledge the contributions of Maarten Vergauwen, Kurt Cornelis, Frank Verbiest, Jan Tops, Reinhard Koch and Benno Heigl to the presented work. Marc Pollefeys is a post-doctoral fellow of the Fund for Scientific Research - Flanders (Belgium). The financial support of the FWO project G.0223.01 and the IST projects InView (IST-2000-28459) and ATTEST (IST-2001-34396) are also gratefully acknowlegded.

References

[1] Beardsley P, Zisserman A, Murray D. Sequential Updating of Projective and Affine Structure from Motion, *International Journal of Computer Vision* (23), No. 3, Jun-Jul 1997, pp. 235-259.

[2] Buehler C, Bosse M, McMillan L, Gortler S, Cohen M. Unstructured Lumigraph Rendering, *Proc. SIGGRAPH* 2001, pp. 425-432.

[3] Chai J-X, Tong X, Chan S-C, Shum H-Y. Plenoptic Sampling, *Proc. SIGGRAPH 2000*, pp.307–318.

[4] Cox I, Hingorani S, Rao S. A Maximum Likelihood Stereo Algorithm, *Computer Vision and Image Understanding* 1996; Vol. 63, No. 3, pp. 542–567.

[5] Curless B and Levoy M. A Volumetric Method for Building Complex Models from Range Images, *Proc. SIGGRAPH '96*, pp. 303–312.

[6] Debevec P and Yizhou Y and Borshukov G, Efficient View-Dependent Image-Based Rendering with Projective Texture-Mapping, *Proc. Eurographics Rendering Workshop* 1998, pp. 105–116.

[7] Falkenhagen L. Hierarchical Block-Based Disparity Estimation Considering Neighbourhood Constraints. *Proc. International Workshop on SNHC and 3D Imaging* 1997, pp. 115–122.

[8] Faugeras O. What can be seen in three dimensions with an uncalibrated stereo rig, *Computer Vision - ECCV'92*, Lecture Notes in Computer Science, Vol. 588, Springer-Verlag, 1992, pp. 563–578.

[9] Faugeras O, Luong Q-T, Maybank S. Camera self-calibration: Theory and experiments, *Computer Vision - ECCV'92*, Lecture Notes in Computer Science, Vol. 588, Springer-Verlag, 1992, pp. 321–334.

[10] Fischler M and Bolles R. RANdom SAmpling Consensus: a paradigm for model fitting with application to image analysis and automated cartography, *Communications of the ACM* 1981, 24:381-95.

[11] Gortler S, Grzeszczuk R, Szeliski R and Cohen MF, The Lumigraph, *Proc. SIGGRAPH '96*, pp 43–54.

[12] Harris C, Stephens M, A combined corner and edge detector, *Fourth Alvey Vision Conference*, 1988, pp. 147–151.

[13] Hartley R, Gupta R, Chang T. Stereo from uncalibrated cameras, *Proc. Conference Computer Vision and Pattern Recognition* 1992, IEEE Computer Society Press, pp. 761-764.

[14] Hartley R. Euclidean reconstruction from uncalibrated views, in *Applications of Invariance in Computer Vision*, Mundy J L, Zisserman A, Forsyth D (eds.), Lecture Notes in Computer Science, Vol. 825, Springer-Verlag, 1994, pp. 237–256.

[15] Hartley R, Zisserman A. *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.

[16] Heigl B, Koch R, Pollefeys M, Denzler J, Van Gool L. Plenoptic Modeling and Rendering from Image Sequences taken by Hand-held Camera, *Proc. DAGM'99*, pp. 94–101.

[17] Kahl F, Triggs B, Åström K. Critical Motions for Auto-Calibration When Some Intrinsic Parameters Can Vary, *Journal of Mathematical Imaging and Vision* 13, 2000, pp. 131–146.

[18] Koch R, Pollefeys M, Van Gool L, Multi Viewpoint Stereo from Uncalibrated Video Sequences. *Proc. European Conference on Computer Vision*, Lecture Notes in Computer Science, Vol. 1406, Springer-Verlag, 1998, pp. 55–71.

[19] Koch R, Pollefeys M, Heigl B, Van Gool L, Niemann H. Calibration of Hand-held Camera Sequences for Plenoptic Modeling, *Proc. International Conference on Computer Vision* 1999, IEEE Computer Society Press, pp. 585–591.

[20] Koch R, Heigl B, Pollefeys M. Image-Based Rendering from Uncalibrated Lightfields with Scalable Geometry, In *Multi-Image Analysis* Klette R, Huang T, Gimel'farb G (eds.), Lecture Notes in Computer Science, Vol. 2032, Springer-Verlag, 2001, pp. 51–66.

[21] Koch R, Heigl B, Pollefeys M, Van Gool L, Niemann H. A Geometric Approach to Lightfield Calibration, *Proc. CAIP99*, Lecture Notes in Computer Science, Vol. 1689, Springer-Verlag, 1999, pp. 596–603.

[22] Levoy M, Hanrahan P, Lightfield Rendering, *Proc. SIGGRAPH '96*, pp 31–42.

[23] Ma Y, Soatto S, Košecká J, Sastry S. Euclidean Reconstruction and Reprojection Up to Subgroups, *Proc. International Conference on Computer Vision* 1999, IEEE Computer Society Press, pp. 773–780.

[24] Ofek E, Shilat E, Rappopport A, Werman M. Highlight and Reflection Independent Multiresolution Textures from Image Sequences, *IEEE Computer Graphics and Applications*, vol.17 (2), March-April 1997, pp. 18–29.

[25] Pollefeys M, Koch R, Van Gool L. Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Internal Camera Parameters, *Proc. International Conference on Computer Vision* 1998, Narosa Publishing House, pp. 90–95.

[26] Pollefeys M, Koch R, Vergauwen M, Van Gool L. An Automatic Method for Acquiring 3D Models from Photographs: applications to an Archaeological Site, *ISPRS International Workshop on Photogrammetric Measurement, Object Modeling and Documentation in Architecture and Industry*, Thessaloniki, *International Archive of Photogrammetry and Remote Sensing*, Vol. XXXII, Part 5W11, 1999, pp. 76–80.

[27] Pollefeys M, Koch R, Vergauwen M, Van Gool L. Hand-held acquisition of 3D models with a video camera, *Proc. Second International Conference on 3-D Digital Imaging and Modeling (3DIM)* 1999, IEEE Computer Society Press, pp. 14–23.

[28] Pollefeys M, Koch R, Van Gool L. A simple and efficient rectification method for general motion, *Proc. International Conference on Computer Vision* 1999, IEEE Computer Society Press, pp. 496–501.

[29] Pollefeys M, Van Gool L. Stratified self-calibration with the modulus constraint, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, August 1999, Vol 21, No.8, pp. 707–724.

[30] Pollefeys M, Koch R, Van Gool L. Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Internal Camera Parameters, *International Journal of Computer Vision*, 32(1), 1999, pp. 7–25.

[31] Pollefeys M, Koch R, Vergauwen M, Van Gool L. Automated reconstruction of 3D scenes from sequences of images, *ISPRS Journal Of Photogrammetry And Remote Sensing* (55)4, 2000, pp. 251–267.

[32] Pollefeys M, Van Gool L. Some Geometric Insight in Self-Calibration and Critical Motion Sequences, Technical Report Nr. KUL/ESAT/PSI/0001, PSI-ESAT, K.U.Leuven, 2000.

[33] Pollefeys M, Vergauwen M, Verbiest F, Cornelis K, Tops J, Van Gool L. Virtual Models from Video and Vice-Versa, *Proc. International Symposium on Virtual and Augmented*

Architecture (VAA01), Fisher B, Dawson-Howe K, O'Sullivan C (eds.), 2001, Springer-Verlag, pp. 11–22.

[34] Pollefeys M, Van Gool L. Do ambiguous reconstructions always give ambiguous images?, *Proc. International Conference on Computer Vision* 2001, IEEE Computer Society Press, pp. 187–192.

[35] Shi J, Tomasi C, Good Features to Track, *Proc. Conference on Computer Vision and Pattern Recognition* 1994, IEEE Computer Society Press, pp. 593–600.

[36] Schroeder W, Zarge J, Lorensen W. Decimation of triangle meshes. *Computer Graphics (SIGGRAPH '92 Proceedings)*, 1992, 26(2), pp. 65–70.

[37] Sturm P, Critical Motion Sequences for Monocular Self-Calibration and Uncalibrated Euclidean Reconstruction, *Proc. Conference on Computer Vision and Pattern Recognition* 1997, IEEE Computer Society Press, pp. 1100–1105.

[38] Sturm P, Critical Motion Sequences for the Self-Calibration of Cameras and Stereo Systems with Variable Focal Length, *Proc. British Machine Vision Conference* 1999, pp 63–72.

[39] Tomasi C and Kanade T, Shape and motion from image streams under orthography: A factorization approach, *International Journal of Computer Vision*, 9(2), 1992, pp. 137–154.

[40] Torr P, *Motion Segmentation and Outlier Detection*, PhD Thesis, Dept. of Engineering Science, University of Oxford, 1995.

[41] Triggs B, The Absolute Quadric, *Proc. Conference on Computer Vision and Pattern Recognition* 1997, IEEE Computer Society Press, pp. 609–614.

[42] Triggs B, McLauchlan P, Hartley R, Fiztgibbon A, Bundle Adjustment – A Modern Synthesis, In *Vision Algorithms: Theory and Practice*, Triggs B, Zisserman A, Szeliski R (eds.), Lecture Notes in Computer Science, Vol.1883, Springer-Verlag, 2000, pp. 298–372.

[43] Van Meerbergen G, Vergauwen M, Pollefeys M, Van Gool L. A Hierarchical Symmetric Stereo Algorithm Using Dynamic Programming, *International Journal on Computer Vision* 47(1/2/3), 2002, pp. 275–285.

[44] Wood D, Azuma D, Aldinger K, Curless B, Duchamp T, Salesin D, Stuetzle W. Surface Light Fields for 3D Photography, *Proc. SIGGRAPH* 2000, pp. 287–296.

Figure 2: Original image pair (left) and rectified image pair (right).

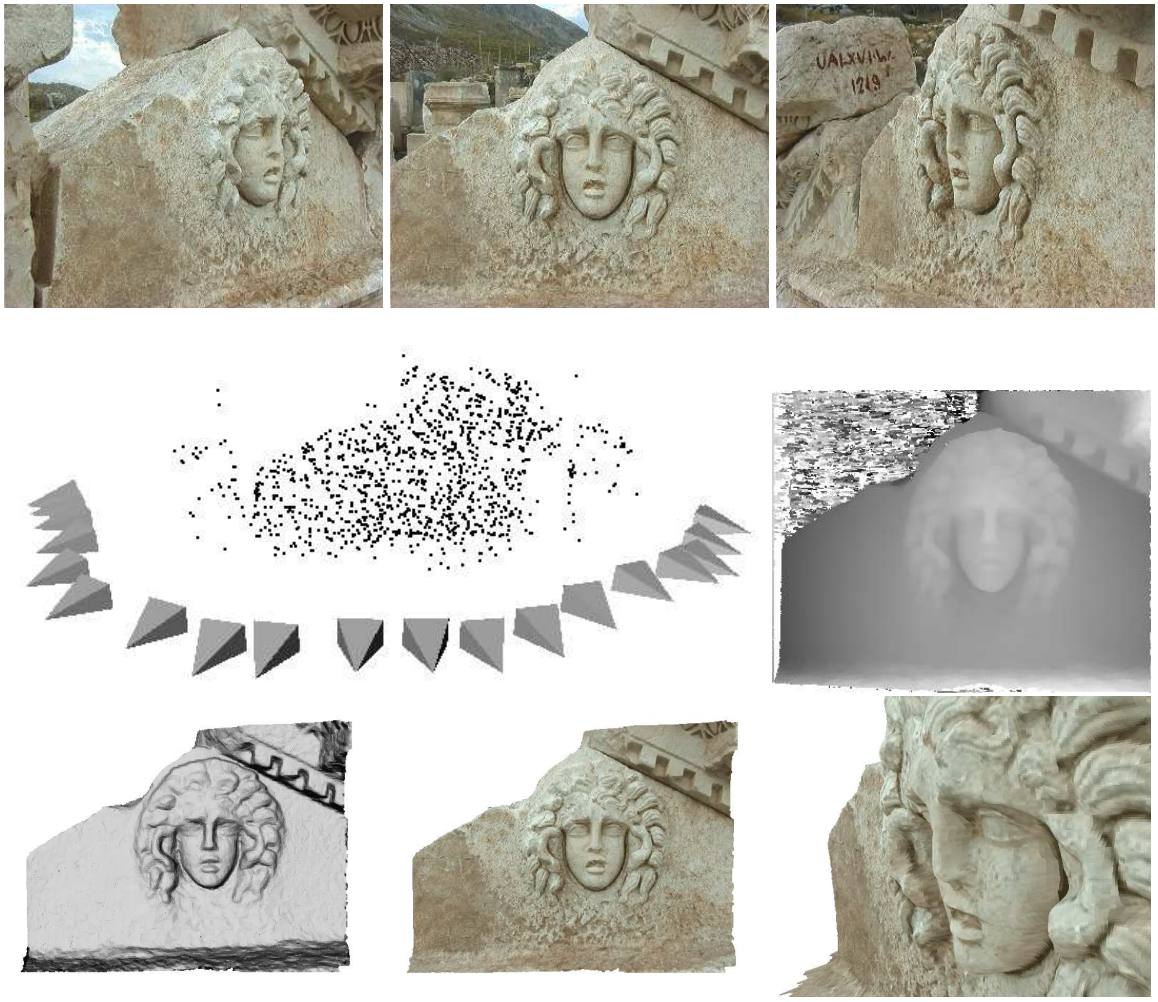

Figure 3: 3D model of a decorative Medusa head recorded at the ancient site of Sagalassos in Turkey. Top: 3 views of the original video, middle: reconstruction of 3D feature points with computed camera motion for the keyframes and one of the computed depth/range images, bottom: shaded and textured views of the recovered 3D model.

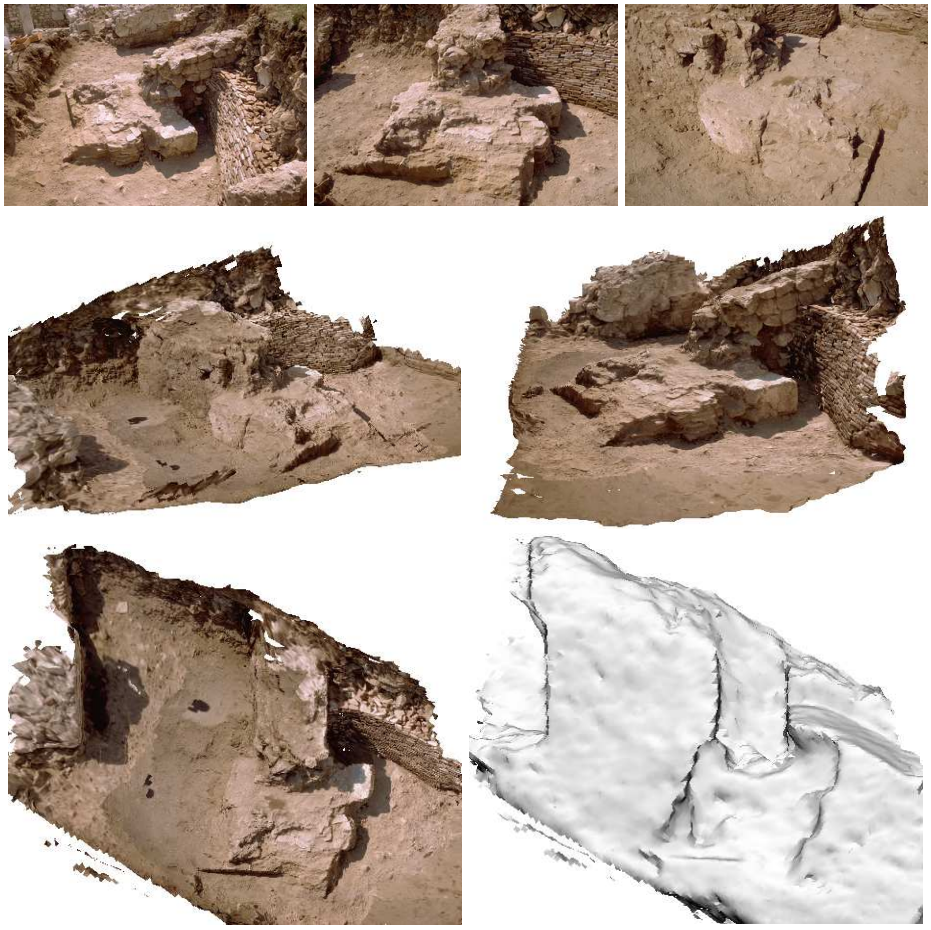

Figure 4: Integrated 3D representation of the excavations of an ancient roman villa in Sagalassos.

Top: two side-views of the 3D model, bottom: texture and shaded top-view of the 3D model.

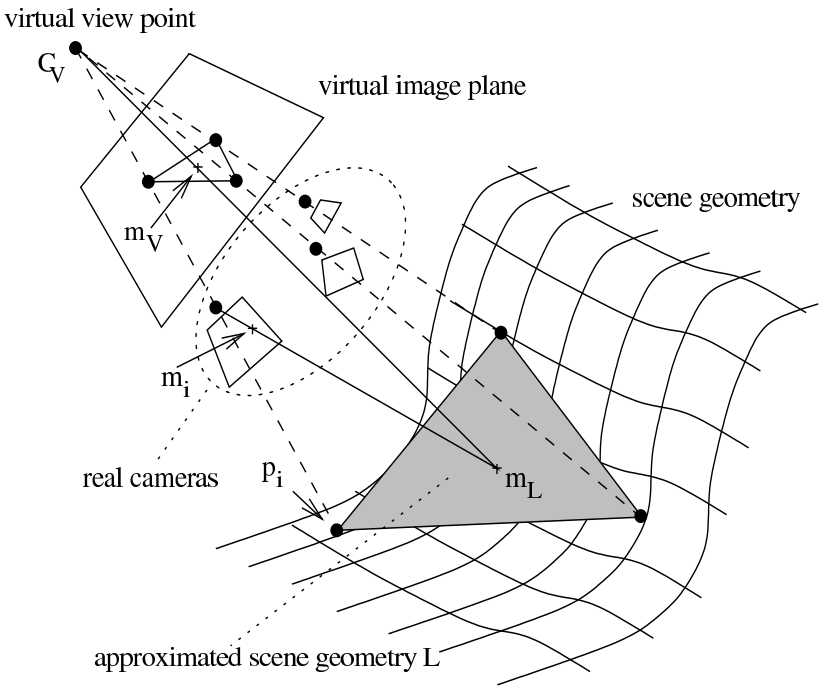

Figure 5: Drawing triangles of neighboring projected camera centers and approximating geometry by one plane for the whole scene, for one camera triple or by several planes for one camera triple.

Figure 6: Unstructured lightfield rendering: image from the original sequence (top-left), recovered structure and motion (top-right), novel views generated for planar (middle-left) and view-dependent (middle-right) geometric approximation, details for different levels of geometric approximation (bottom).

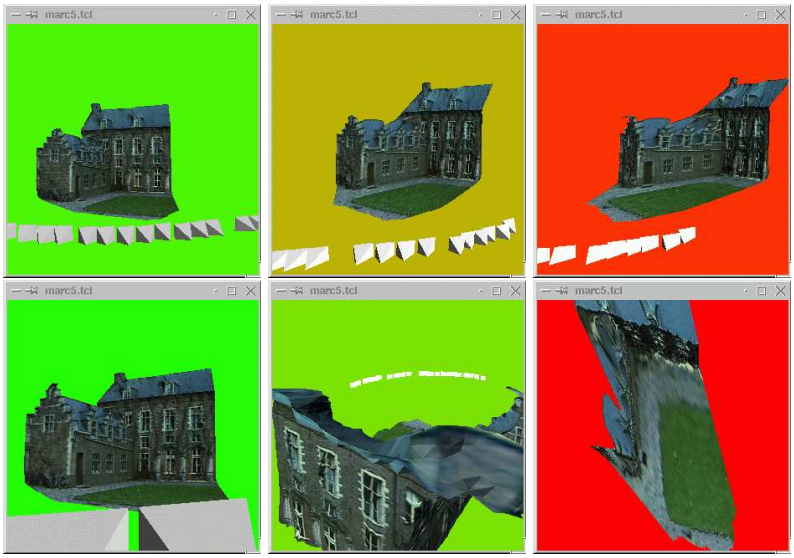

Figure 7: Different views of the castle with estimated relative ambiguities of 0.5, 1.5, 3 (top) and 0.1, 1, 8 (bottom).