Generation of Orthoimages and Perspective Views with Automatic Visibility Checking and Texture Blending

George E. Karras, Lazaros Grammatikopoulos, Ilias Kalisperakis, and Elli Petsa

Abstract

Conventional orthorectification software cannot handle surface occlusions and image visibility. The approach presented here synthesizes related work in photogrammetry and computer graphics/vision to automatically produce orthographic and perspective views based on fully 3D surface data (supplied by laser scanning). Surface occlusions in the direction of projection are detected to create the depth map of the new image. This information allows identifying, by visibility checking through back-projection of surface triangles, all source images which are entitled to contribute color to each pixel of the novel image. Weighted texture blending allows regulating the local radiometric contribution of each source image involved, while outlying color values are automatically discarded with a basic statistical test. Experimental results from a close-range project indicate that this fusion of laser scanning with multi-view photogrammetry could indeed combine geometric accuracy with high visual quality and speed. A discussion of intended improvements of the algorithm is also included.

Introduction

Problems and error sources faced in orthoimage generation have been extensively reviewed in literature, chiefly with respect to aerial images (Amhar *et al.*, 1998; Schickler and Thorpe, 1998; Palà and Arbiol, 2002; Rau *et al.*, 2002; Sheng *et al.*, 2003), but also high-resolution satellite imagery (Chen *et al.*, 2001). The importance of radiometry is self-evident: orthoimage quality depends on the source images and may be further degraded, e.g., due to radiometric differences among overlapping images, inadequate hole-filling, or remaining shadows. Yet, it is the geometric aspect which forms the core issue of what has been termed "true orthophoto generation," namely the resampling of images in which all visible surface points occupy their correct planimetric position. The primary source of geometric distortion of a conventional orthoimage is two-fold: surface modeling and orthorectification algorithm. Due to surface complexity, the geometric aspect is probably even more important in close-range photogrammetry. Here, an approach is presented for the automatic generation of geometrically correct orthoimages based on multiple images and an existing 3D surface model. Cultural heritage is the specific field of experimental application in this paper.

The Aerial Case

Digital Surface Models

Orthoimages depend not only on the actual accuracy of the surface model, but primarily on its content. When only terrain morphology is modeled in a digital terrain model (DTM), objects on the ground surface, man-made structures, trees, etc., are disregarded. This, of course, results in perspective distortions (displacement, incorrect scale), which produce phenomena like "building leaning" and pertinent occluded areas.

Buildings can be modeled with a variety of techniques, based on automatic and interactive photogrammetric approaches but also on range sensors (e.g., Hoffman, 2004; Rau and Chen, 2004; Zhou *et al.*, 2004.). They can be introduced in the form of digital building models (DBM) and employed autonomously for generating orthophotos supplementary to those from DTMs, with which they will be subsequently merged (Amhar *et al.*, 1998). Alternatively, a DBM may be integrated in advance with a DTM to constitute a digital surface model (DSM), which fully delineates the entire surface (Rau *et al.*, 2002; Oda *et al.*, 2004). Direct collection of DSMs, usually through aerial laser scanning, is also possible (Katzenbeisser, 2004).

Detection of Image Occlusions

It is known that a typical orthorectification algorithm will still not provide the desired outcome even at the presence of a detailed DSM. The man-made structures might be projected correctly, but hidden areas (needing texture-filling from adjacent images) will also be present, occupied by double projections of building roofs (Rau *et al.*, 2002). It has been noted that traditional DEMs or DSMs represent the surface in 2.5D (i.e., a single elevation value is assigned to each planimetric location), whereas only a proper 3D model could meet the requirements of modern large-scale urban orthoimaging tasks (Zhou *et al.*, 2004). But, with few exceptions (notably bridges), the requirements of orthoimaging

George E. Karras, Lazaros Grammatikopoulos, and Ilias Kalisperakis are with the Department of Surveying, National Technical University of Athens (NTUA), GR-15780, Athens, Greece (gkarras@central.ntua.gr; lazaros@central.ntua.gr; ilias_k@central.ntua.gr).

Elli Petsa is with the Department of Surveying, Technological Educational Institute of Athens (TEI-A), GR-12210, Athens, Greece (petsa@teiath.gr).

Photogrammetric Engineering & Remote Sensing
Vol. 73, No. 4, April 2007, pp. 403–411.

0099-1112/07/7304–0403/$3.00/0

in urban areas could essentially be satisfied with 2.5D models. Thus, Rognant *et al.* (2000) employ TIN models derived with 2D Delaunay triangulation for the creation of true orthoimages (see Oda *et al.*, 2004; Palà and Arbiol, 2002). In fact, the key problem here is that conventional orthorectification algorithms are not capable of properly handling even a 2.5D model, since they function on the assumption that all involved model points are indeed visible on the source image, i.e., they cannot tackle the image occlusion problem.

In fact, all publications cited above present alternative approaches for producing correct orthoimages by identifying image occlusions. Several among these (Amhar *et al.*, 1998; Rau *et al.*, 2002; Sheng *et al.*, 2003) adopt variations of the *Z*-buffer approach, a basic method in computer graphics for removing hidden surfaces. All surface polygons are back-projected on the source image to allow creating a depth-buffer, which for every pixel records the shortest distance from the perspective center to the surface. A corresponding index map assigns to each pixel the identification code of the visible surface polygon. Thus, in the orthorectification process, only pixels of a source image which are actually entitled to participate are kept. Sheng *et al.* (2003) have referred to certain limitations of such algorithms, which may result in the appearance of artifacts on the orthoimage. On the other hand, Kuzmin *et al.* (2004) suggest a faster, resolution independent algorithm for polygon-based hidden area detection. The intersections of surface polygons projected onto the source image plane result in the image plane subdivision into faces, for which a list is made with the corresponding overlapping polygons, unambiguously ordered by distance to projection center. Thus, the visible polygon for each face is established. Back-projection of faces defines the areas of the surface model visible on the original image.

Such approaches produce individual orthoimages with blank areas (blind spots), which must be filled with texture from overlapping imagery. Rau *et al.* (2002) select a master image, while the adjacent ones serve as slave images intended to supply the missing information; a seamless mosaicking tool is applied to avoid radiometric discontinuities caused by direct patching from the slave images. Texture may also be selected by considering the best intersection angle between surface and imaging ray (Amhar et *al.*, 1998; Sheng *et al.*, 2003). In the automatic mosaicking and seam line optimization technique of Schickler and Thorpe (1998), a "best" image is selected employing a quality measure for each ortho-pixel, from which a "weight image" for every individual orthophoto is formed. Besides surface slope relative to viewing ray, the distances from the nadir point and from the blind spot are also considered for weighting (the latter criterion actually introduces a significant topological aspect). The approach of Oda *et al.* (2004) produces composite orthoimages via a weighted contribution of individual orthoimages.

The Wider Aspect of Texture Mapping

Despite the undoubted practical importance of orthoimages, it has been pointed out that with the increasing importance of real 3D surface modeling they will, sooner or later, become just one special case of a general image transformation (Amhar *et al.*, 1998). In this wider context of texture-mapping, for example, orthoprojection would actually represent a special case of the approach of Früh *et al.* (2004) for texturing 3D city models (obtained from aerial and ground-based laser scans) with multiple oblique aerial images, which depict both façades and rooftops. For each model triangle an optimal image is selected according to certain criteria; these texture patches from all images are then fused into one "texture atlas" for compact representation and efficient rendering. However, it is probably in close-range applications (and documentation of cultural heritage which is mainly addressed here, in particular) that orthoimaging is best grasped as an exemplifying version of the generic case of photogrammetric texture-mapping.

The Terrestrial Case

Notwithstanding certain additional problems typically pertaining to orthoimaging of cultural items, the pivotal role of surface modeling for a geometrically reliable and visually flawless outcome remains undisputable (Mavromati *et al.*, 2002). An overview of approaches, both automatic and interactive, for 3D model capturing and visualization used in cultural heritage, along with a discussion on their merits and limitations, is found in El-Hakim *et al.* (2003a).

Surface Modeling in Close Range

Object shape may be quite complex in close-range projects. With commercial image matching software being generally unsuitable for archaeological objects, surface points are conventionally collected manually in a digital photogrammetric environment. Even under demanding circumstances, proper point and breakline collection strategies may produce excellent results (Mavromati *et al.*, 2002) but, obviously, they are very cumbersome and time-consuming. Standing at the other extreme of image-based modeling, powerful algorithms in the field of computer vision allow automatic production of textured 3D surface models from image sequences with no prior knowledge about camera or object. Models of high visual quality are thus produced, but it is unclear whether obtained accuracies match the requirements for most mapping applications (Pollefeys *et al.*, 2000). Advanced tools for automatic dense reconstruction from a limited number of multiple wide-baseline images, rather than sequences, are also being reported (Strecha *et al.*, 2003), but their actual metric performance remains to be further evaluated.

On the other hand, the enormous number of surface points sampled today at very fast rates using the powerful technology of laser scanning represent an alternative fully 3D support for orthorectification, but also for generating high quality photo-textured virtual models of real-world scenes in computer graphics (Bernardini *et al.*, 2001). In fact, the promotion of cultural heritage appears as one of the main driving forces of research as regards fusion of laser scanning and color imagery (Beraldin *et al.*, 2002). The poor image quality of model-registered color textures provided by certain commercial 3D systems dictates the acquisition of separate high-resolution images, particularly for the purposes of orthoimaging at very large scales.

The General Question of Visibility

Even after post-processing of raw 3D data and mesh triangulation, the two-fold limitation of common orthoprojection software remains. It does not check what the image actually "sees," i.e., image visibility; it only handles 2.5D surface meshes. But unlike most aerial cases in which, as mentioned, a 2.5D surface description might suffice, in close-range it is very often impossible to dispense with fully 3D models due to object complexity. Hence, a rigorous orthoimaging algorithm is primarily obliged to handle a two-fold question of visibility:

- All surface units actually visible in the direction of orthoprojection must be established, with the outcome that to each orthoimage pixel a unique elevation is assigned.
- Next, it is checked which surface points, among those established above, are also visible on the input image; otherwise, color values must be drawn from adjacent images.

Thus, Wanshou and Yixuan (1999) generated depth maps of both the orthoimage and the source imagery; adhering to the

standard photogrammetric practice, they extracted texture for visible patches from one source image, chosen according to proximity to its projective center.

It is needless to underline that this general way of putting the question of visibility has nothing peculiar to orthoimaging, and hence is also suited, e.g., for generating perspective views.

Texture from Multiple Images

Due to object complexity and irregular image configurations, it is doubtful whether in close range applications a "master-slave image" scheme could be adopted. Besides, in practical situations color and intensity for a surface element will not be identical on different images, due to lighting conditions and effects, but also camera calibration, bundle adjustment, registration, or modeling errors (Baumberg, 2002). Thus, when neighboring orthoimage pixels receive color from different images with varying radiometry, discontinuity artifacts or radiometric distortion may occur. A response to this is color blending, whereby for every point a suitably weighted combination of corresponding textures from all images, or image subsets, is used (Pulli *et al.*, 1998; Neugebauer and Klein, 1999; Buehler *et al.*, 2001; Wang *et al.*, 2001; Rocchini *et al.*, 2002). Such methods for effacing radiometric differences originate from the field of computer graphics, in which realism and illumination are obviously important; understandably, therefore, weighting strategies are mostly formulated in the sense of view-dependent texture mapping, i.e., the interpolation scheme favors images which "see" the object closest in angle to the current viewing direction (Debevec *et al.*, 1996). For instance, Pulli *et al.* (1998) use compatible rays from only three input images taken from directions closest to the current viewing direction.

It has been pointed out, however, that a unique texture map for a 3D model is usually sufficient (Wang *et al.*, 2001). Indeed, for most instances of photogrammetric mapping it seems reasonable to favor a static, rather than a dynamic, texturing process. Thus, a view-independent algorithm would weigh the contribution of input images based on their spatial relation to the object surface itself (distance, angle of view) and on their characteristics (camera constant, resolution) in order to assign a unique color value to each surface unit (Poulin *et al.*, 1998).

Although color blending is, to a certain extent, also an error-absorbing process, geometric and radiometric distortion will still emanate from existing error sources. For instance, uncorrected lens distortion (a grave issue when using off-the-shelf cameras) might drastically degrade the outcome of texture blending. In this sense, the importance of full camera calibration and accurate image registration using photogrammetric bundle adjustment is clear. Nevertheless, additional means for the detection of outliers are still indispensable. The emergence of blurring effects in color blending due to significant differences in resolution of the source images is also an issue to be taken into account (Neugebauer and Klein, 1999; Buehler *et al.*, 2001).

Outline of the Presented Approach

Relying on the related literature, the objective here was to view the basic photogrammetric problem of orthorectification in the broader context of texture-mapping, in order to develop an algorithm by adapting and combining tools from the fields of both photogrammetry and computer graphics. Thus, the presented approach generates automatically correct orthoimages (or perspective views) from a 3D mesh, obtained using terrestrial laser scanning, and multiple image coverage. This procedure requires the following input data:

- a triangulated 3D mesh in the form of *X, Y, Z* triplets describing the object surface;
- grayscale or color images with their interior and exterior orientation parameters; and
- the equation in space of the projection plane (besides orthogonal, oblique parallel projections may also be accommodated) or the interior and exterior orientation elements of the new perspective image.

Briefly put, in its successive steps the implemented algorithm:

- identifies all surface triangles seen in the desired direction of projection and generates the depth map of the new image;
- establishes which of the surface points, identified above as visible on the new image, also appear on each available source image;
- colors the pixels of the new image by weighted blending of texture from all viewing images; and
- provides a tool for automatically discarding outlying color data.

Visibility Checking

The primary task is to handle the two-fold occlusion problem. Following Figure 1, one needs to establish object regions which are not visible in the direction of projection, i.e., self-occlusions (A), and object parts occluded on the image (B). This is synonymous with having established: all object parts which, although recorded on a particular source image, should not appear on the new projection (C); and object regions which are visible in the new projection, but to whose texturing the particular source image is not permitted to contribute.

Surface Occlusion

The question of surface occlusion in the desired direction of projection comes first. Without loss of generality, the focus is here on orthoprojection. The triangulated 3D mesh is orthogonally projected onto the specified projection plane (assumed parallel to the *X-Y* plane of the space coordinate system). A tessellation of the blank orthoimage array into a rectangular grid with cells larger (for example, by a factor of 5) than the orthoimage pixels (this factor depends on available computer memory, model, and orthoimage size) facilitates the speeding

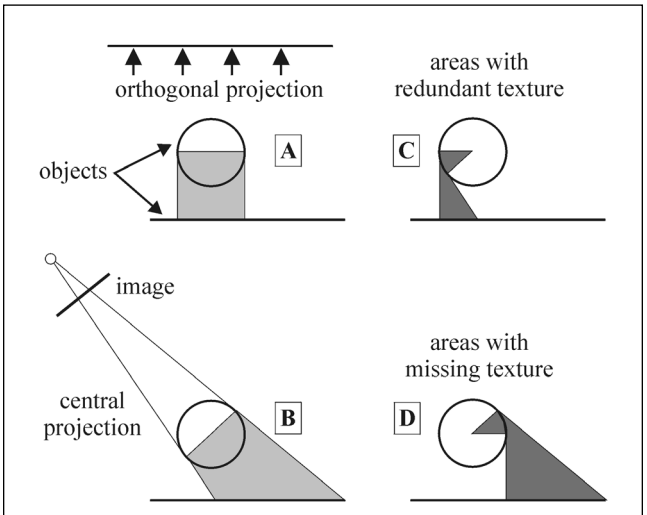


Figure 1. The question of occlusion. A: surface occlusion in the direction of orthoprojection; B: object area occluded in the image; C: object area with redundant image texture (which must not participate in the orthoprojection); D: object area which lacks the image texture required for orthoprojection.

up of the search process. For each projected triangle, its circumscribing orthogonal parallelogram is formed, occupying a number of adjacent grid cells; to these the identity number (ID) of the particular triangle is assigned.

This step finally produces a table containing all triangle ID numbers which are ascribed to each individual grid cell. Thus, all projected triangles containing a certain pixel of the orthoimage are identified by checking only the number of triangles ascribed to the corresponding grid cell. Among model triangles (those of points A and B in Figure 2) intersected in space by the projection ray (m in Figure 2) of a particular orthoimage pixel, the one whose intersection produces the largest *Z*-value (or, more generally, is closest to the projection plane along the direction of projection) is kept (A in Figure 2). This value provides the particular orthoimage pixel with its *Z*-value, which is stored along with the triangle ID number. Thus, the generation of the orthoimage depth map handles the question of surface visibility/occlusion.

Image Occlusion

Coming now to the second visibility issue, first all surface triangles are centrally projected onto all source images involved. For the 3D object coordinates now uniquely associated with each orthoimage pixel, the corresponding image coordinates on every image are calculated. A search scheme similar to that described previously is adopted. Among all model triangles (those of points C, A, and D in Figure 2) intersected by a particular image ray (n in Figure 2, which corresponds to surface point A), the one closer to the projection center (C in Figure 2) is the triangle actually depicted on the image. If its ID number does not agree with that already ascribed at the previous stage to the orthoimage pixel (as it is the case in Figure 2), it is established that the model point associated with this particular orthoimage pixel is occluded on the image. Identical triangle ID numbers, on the contrary, mean that the model point is indeed visible on the image, too; the RGB values can be interpolated at the particular image location and stored.

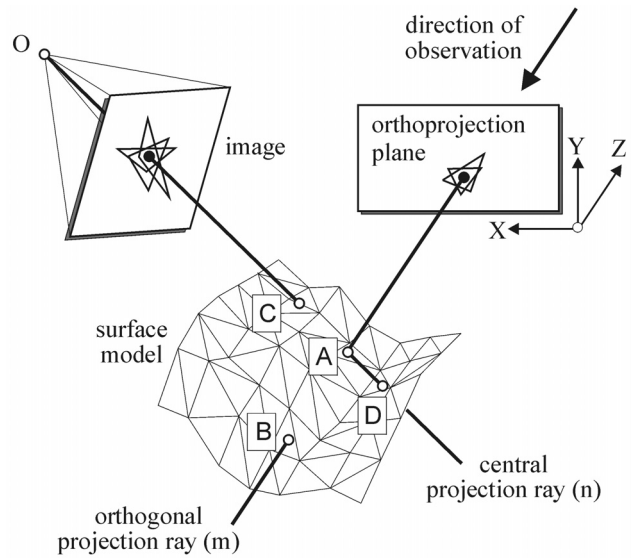


Figure 2. Identification of visibility/occlusion by projection of surface triangles. An orthoprojection ray (m) intersects the surface in points A and B; point A is closer to the projection plane and, hence, is visible. Yet, the central projection ray (n) through A also intersects the surface in points C and D. The image has recorded point C, which is closest to the projection center O. Hence, this image is not entitled to contribute texture for point A.

Texture Blending

The preceding steps establish specific locations on source images which may contribute texture to each individual orthoimage pixel. Two further questions need now to be considered for each orthoimage pixel: how to interpolate the individual color value of each source image, and how to extract from these values the final texture for the orthoimage pixel.

Individual Color Interpolation

To achieve a smoothing effect, the authors currently use bicubic convolution to interpolate a color value at the locations indicated on each image by back-projection (Grammatikopoulos *et al.*, 2004). But if there exist significant differences in scale among images, the standard 4×4 image window represents different dimensions on both the orthoimage and the model. In this case, corresponding interpolations on different images will involve textures from varying surface patches. It is considered to take this into account by selecting a standard window size on the orthoimage, to represent the acceptable area for interpolation, and back-project it each time (through the 3D model) to define for every point of the source images its specific limits for interpolation.

Furthermore, it is equally true that adjacent image pixels do not necessarily pertain to adjacent surface points, with the most obvious example being the brinks of image occlusions. If depth values for all pixels of the original images were available, setting a depth threshold would allow rejecting such pixels from interpolation. In fact, though the visibility-checking algorithm itself does not necessitate the creation of explicit depth maps for the source images, generation of depth maps for the purpose of outlier detection (see next section) is an option which has been implemented here. Their use in the present context, too, is also being currently materialized. It is noted that Früh *et al.* (2004) also employ depth maps to identify pixels as edges if their depth difference from their neighbors exceeds a threshold value.

After interpolation, color values from several images are at hand for every orthoimage pixel, unless of course, the particular surface point is invisible in all images (these orthoimage pixels remain blank). For such regions, hole-filling tools could extract color values from surrounding model regions (Poulin *et al.*, 1998). This issue is not examined here.

Discarding Outlying Color Values

In order to assign final textures to the orthoimage pixels form all viewing images, the outlying values must be first excluded. These may originate from orientation or model faults, as well as from view-dependent features, namely obstacles, specular highlights, transparencies, mirrors, or refractions (Poulin *et al.*, 1998; Rocchini *et al.*, 2001). Yet, more significant in photogrammetry is probably proximity to a surface region invisible to a camera, i.e., an occlusion border (Neugebauer and Klein, 1999; Buehler *et al.*, 2001). Artifacts can emerge, since even small registration or modeling errors could be responsible for erroneous color values derived from an occluded, or an occluding, surface point (Figure 3). Baumberg (2002) has addressed this issue by using an image weight map, feathered at the occlusion borders to ensure a continuous weight function. Bornik *et al.* (2001), on the other hand, use a median filter to single out undesired values falling outside of a user-specified range.

Rather than a fixed threshold value, the authors handle this "occlusion risk" employing an adaptive value

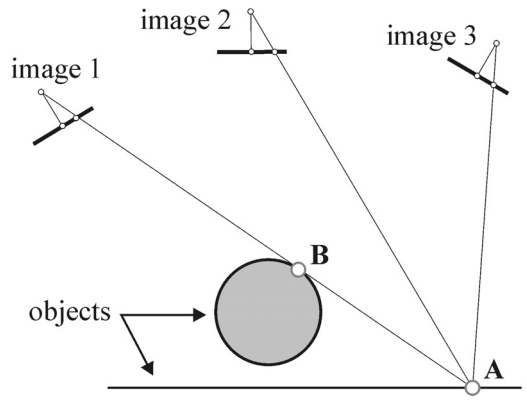


Figure 3. Possible texturing errors at occlusion borders. Images 2 and 3 have unobstructed view of object point A. On the contrary, the projection ray of A in image 1 runs very close to an occlusion border at point B. Hence, due to modeling or orientation inaccuracies, image 1 might contribute to the orthoimage of point A irrelevant texture from point B.

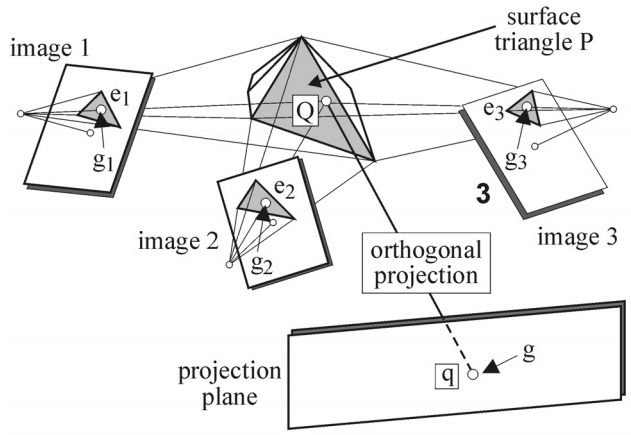


Figure 4. In order to synthesize a texture value g for a particular orthoimage pixel q, which corresponds to an object point Q, all viewing images are used. The final value g is the weighted average of all corresponding individual values g_i, referring to object point Q which belongs to surface triangle P. The weights are derived from the surface areas e_i of the corresponding image triangles (i.e., the central projections of P).

(Grammatikopoulos *et al.*, 2004). From all color values which are available for a particular orthoimage pixel, the mean μ and standard deviation σ are computed, to allow excluding individual color values falling outside the range $\mu \pm \sigma$. As with the median filter, at least three values are needed. The algorithm creates a map with the distribution of orthoimage pixels depicted on 0, 1, 2, and > 2 original images. Hence, additional images may eventually be introduced, if available. Nonetheless, in order to handle the orthoimage pixels for which color values from only two source images exist, image depth maps may be useful. If these two color values differ by more that a limit (set according to the σ-value for preceding pixels), the pixel whose depth value differs more from that of its neighbors may be discarded.

Final Weighting of Color Values

Eventually, the valid contributing color values from all participating images are exploited to generate the final texture of every orthoimage pixel, calculated as the weighted mean of all contributing images. In view-independent texturing, the main factors affecting color quality are:

- scale (imaging distance and camera constant) of the source image;
- viewing angle (the angle formed by the intersection of image ray and the surface triangle); and
- image resolution.

The above factors are all combined to yield the size (in pixel dimensions) of the 2D triangle on each image, which is regarded as a realistic indication for the "strength" of the corresponding color (Poulin *et al.*, 1998). Thus, as illustrated in Figure 4, the contribution of each participating color value is weighted with the surface area of its corresponding 2D image triangle (a weighting scheme also adopted by Visnovcova *et al.*, 2001).

Experimental Evaluation

The algorithm was first developed in Matlab® and finally implemented in C. For its actual experimental evaluation, the entrance of a prominent eleventh century Byzantine church situated in the center of Athens was chosen, an object characterised by explicit self-occlusions (Grammatikopoulos *et al.*, 2004).

Surface Modeling and Photogrammetric Adjustment

The Mensi GS200 laser scanner sampled the surface in three separate scans from a distance of about 5 m (for which a typical σ-value of ± 1.4 mm is given). Six target spheres, also measured geodetically, served for the purposes of registration. This target-based georeferencing of scans resulted in a precision of about ± 2.5 mm. The merged surface point cloud, which consisted of a total of 7 million points, was edited with the Geomagic Studio® software, to be finally reduced to 1 million points. The final TIN model comprised 3 million triangles (a view is seen in Figure 5).

The monument was recorded with a digital 5 Megapixel camera and fixed focusing to ensure invariant camera geometry. Figure 6 shows all seven images used. A total of 18 signalized control points, most of them appearing on all images, were measured with high precision in order to minimize errors in image registration. Thus, self-calibrating bundle adjustment could provide highly accurate results. Accommodation of the first two coefficients of radial symmetric lens distortion gave on the image plane a standard error of $\sigma_o = \pm 0.28$ pixels (it is noted that heavy distortion was present which, if ignored in self-calibration, produced $\sigma_o = \pm 2.40$ pixels). Camera calibration and orientation data were subsequently used for generating color projections of the model.

Multi-view Orthoimage Generation

Thanks to the fact that all images processed here had been acquired under the same lighting conditions, and hence no significant radiometric variations existed among images, the texture averaging procedure was considered as adequate (regarding pre-processing see Visnovcova *et al.*, 2001). For orthorectification, the plane of the facade was selected as the *X-Y* projection plane. The orthoimage pixel size was set to 2 mm (corresponding to the requirements of the scale

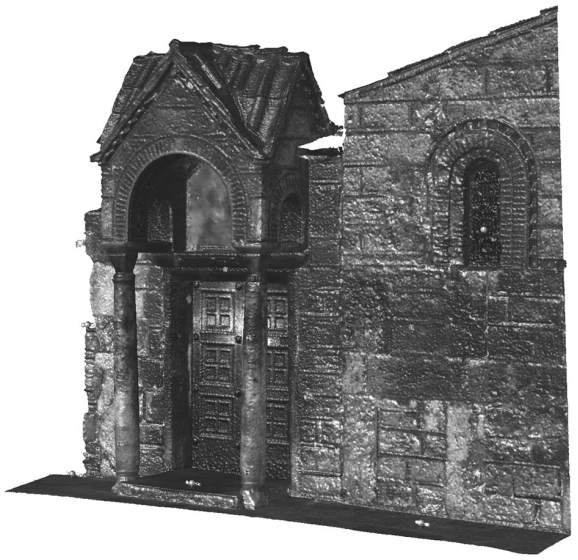


Figure 5. Intensity map of the surface model.

1:20). As mentioned, the algorithm establishes image visibility for all parts of the model which have been previously identified as visible in the direction of orthoprojection. Results for two individual images are seen in Figure 7 showing the orthoimage areas to which these particular source images have valid color to contribute.

The mentioned issue of outlying texture values in the vicinity of an occlusion border is manifestly illustrated in Figure 8. In the orthoimage detail on the left (Figure 8a), which was generated from all seven source images without blunder-filtering, the bright artifact originates from the occlusion border of the particular source image shown in the center (Figure 8b). This fault vanishes if local color values deviating by more than $\pm\sigma$ from their mean are automatically discarded (Figure 8c). Filtering out of such color value outliers not only helps to suppress the emergence of certain artifacts, it also produces sharper images. However, this was possible here thanks to the fact that practically the whole model was visible on three and more images.

The main part of the final orthoprojection, automatically generated with the contribution of all seven images, is seen in Figure 9a. The result is quite satisfactory. The main imperfections are small holes due to lack of texture (no means for elevating the camera had been used) or to surface spots not seen by the scanner. Other flaws, such as some

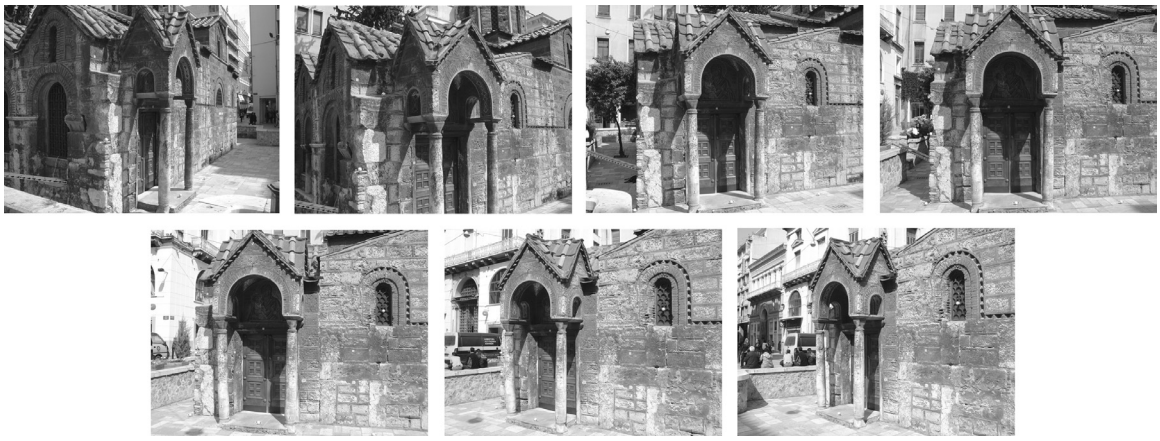


Figure 6. The seven images used for texturing.

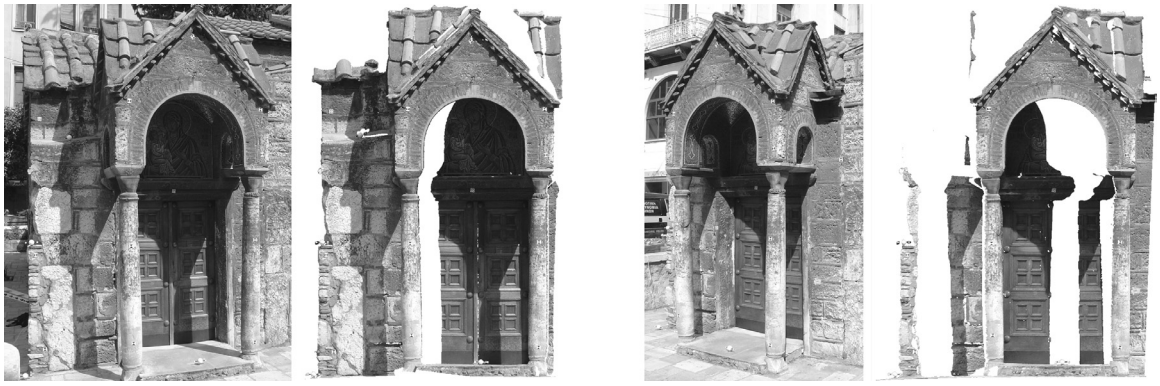


Figure 7. Orthorectification of two individual source images with visibility checking (orthoimage areas not visible on the source images are left blank).

(a) (b) (c)

Figure 8. The artifact in the orthoimage detail (a) clearly originates from the occlusion border of the source image seen in (b). It has been filtered out in (c), where the corresponding detail of the final orthoimage (which is fully presented in Figure 9) is shown.

(a) (b)

Figure 9. Orthoimage (a) and vertical cross-section (b).

aliasing effects which appear at certain edges (yet, these may not necessarily correspond to model regions perfectly perpendicular to the projection plane) are observed only with considerable zooming (i.e., beyond the tolerances of the particular final orthoimage scale).

As regards speed, the current implementation of the algorithm is relatively fast, but can be further improved. For the basic module, a standard PC required five minutes to project the 3 million triangles onto the seven images, in a suitable order to facilitate the last step; one minute to establish the Z-values for each orthoimage pixel and 13 minutes for the production of a final $3{,}720 \times 2{,}775$ texture array from seven source images of $2{,}592 \times 1{,}944$ pixels (Grammatikopoulos *et al.*, 2004).

Generation of Cross-sections

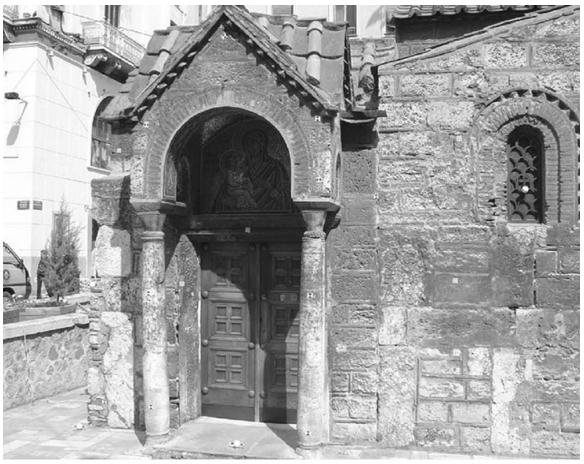
An interesting option of the algorithm is that, in a sense, it may also bring occluded areas to the forefront. For instance, given that the surface model is actually a fully 3D representation, a cross-section may be generated simply by depth thresholding and, thus, reveal hidden modeled regions (provided, of course, that for these source texture is available). Figure 9b presents a vertical cross-section based on all

available model and color data, which includes surface parts not visible on the orthoimage proper.

Generation of Perspective Views

An additional feature of the presented algorithm is that it can also synthesize perspective views. Such novel images are central projections of the surface model with freely chosen values for the geometry of the fictitious camera and its exterior orientation. In Figure 10 an original image (Figure 10a) is seen along with its synthetic counterpart (Figure 10b), merged from the remaining six images using the exact calibration and orientation data of the real image. It has been plausibly remarked that texture blending might introduce a detectable blurring effect (El-Hakim *et a*l., 2003b). It is noted, however, that here the synthetic image appeared as sharp as the source image.

Clearly, the process for creating novel perspective views is similar to that of generating the depth map of a source image. But it should be pointed out, without going into detail (for details see Bräuer-Burchardt and Voss, 2000), that if the lens of the fictitious camera is distortion-free the process is more straightforward, whereas for the projection of a "real" image the re-calculation of the radial distortion



(a)

(b)

Figure 10. Image (a) and synthetic view (b) using identical camera and orientation.

Figure 11. Depth map of a source image.

distribution is indispensable. In Figure 11 the depth map of the image of Figure 10 is shown (light areas are closer to the image projection center).

As mentioned, the depth maps of all source images are means for establishing vicinity to occlusion borders. This is useful not only for facilitating detection of color outliers, but also for possibly excluding "suspect" pixels from local color interpolation.

Conclusion

Combining tools from related fields, an algorithm has been presented for the automatic multi-image synthesis of high quality orthographic views (but also perspective views and cross-sections), given a triangulated 3D surface mesh and image orientation data. Such approaches intend to represent a more general case of photogrammetric texture mapping, of which orthorectification is merely a prominent expression. It is obvious that, although not coincident in scope with photogrammetry, the literature in the computer graphics and computer vision fields amasses a rich theoretical and practical experience concerning high quality model texturing, which is also very beneficial for the purposes of photogrammetric mapping.

The results for the particular object treated here are indeed quite satisfactory, not least thanks to the quality of the input data. Despite certain available means for checking the effects of error sources, the precision of the input data clearly plays a primary role. Evidently, the presented algorithm must be tried under more unfavorable circumstances. Furthermore, its adaptation to the aerial case also needs to be considered.

Of course, further elaboration of the approach is necessary, not least as regards speed. Certain aspects pointed out in Grammatikopoulos *et al.* (2004), e.g., regarding closeness of source pixels to occlusion borders, have already been, or are being, treated. Variations in image scale are also being considered by introducing a surface-fixed rather than an image-fixed color interpolation window. More important, means for both shadow removal and hole-filling still remain to be integrated into the process. Finally, multi-image coverage and precise starting values supplied by the 3D model offer themselves for introducing image matching techniques to allow model and registration refinements (Debevec *et al.*, 1996; Bernardini *et al.*, 2001).

Acknowledgments

This work is co-funded by the E.U. (75 percent) and the Greek Government (25 percent) under the framework of the Education and Initial Vocational Training Program, *Archimedes*. The authors are grateful to Geotech, Ltd. for kindly providing the scans, and to Infocad, Ltd. for kindly providing the license to use Geomagic Studio®.

References

Amhar, F., J. Jansa, and C. Ries, 1998. The generation of true orthophotos using a 3D building model in conjunction with a conventional DTM, *International Archives of Photogrammetry and Remote Sensing*, 32(Part 4):16–22.

Baumberg, A., 2002. Blending images for texturing 3D models, *Proceedings of the British Machine Vision Conference (BMVC 2002)*, pp. 404–413.

Beraldin, J.-A., M. Picard, S.F. El-Hakim, G. Godin, C. Latouche, V. Valzano, and A. Bandiera, 2002. Exploring a Byzantine crypt through a high-resolution texture mapped 3D model: Combining range data and photogrammetry, *Proceedings of the CIPA International Workshop on Scanning for Cultural Heritage Recoding*, Corfu, pp. 65–70.

Bernardini, F., I.M. Martin, and H. Rushmeier, 2001. High-quality texture reconstruction from multiple scans, *IEEE Transactions on Visualization and Computer Graphics*, 7(4):318–332.

Bornik, A., K. Karner, J. Bauer, F. Leberl, and H. Mayer, 2001. High quality texture reconstruction from multiple views, *Journal of Visualisation and Computer Animation*, 12(5):263–276.

Bräuer-Burchardt, C., and K. Voss, 2000. Automatic lens distortion calibration using single views, *Proceedings of the 22th DAGM Symposium (Mustererkennung 2000)*, Springer, Berlin, pp. 187–194.

Buehler, C., M. Bosse, L. McMillan, S. Gortler, and M. Cohen, 2001. Unstructured lumigraph rendering, *Proceedings of the ACM SIGGRAPH Annual Conference Series*, pp. 425–432.

Chen, L.C., C.Y. Lo, and J.Y. Rau, 2001. Generation of true orthophotos from IKONOS GEO images, *Proceedings of the 22nd* Asian Conference on Remote Sensing, Singapore, pp. 1159–1164.

Debevec, P., C.J. Taylor, and G. Malik, 1996. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach, *Proceedings of ACM SIGGRAPH*, pp. 11–20.

El-Hakim, S.F., J.-A. Beraldin, M. Picard, and A.Vettore, 2003a. Effective 3D modeling of heritage sites, *Proceedings of the 4th International Conference of 3D Imaging and Modeling (3DIM '03)*, Banff, Alberta, Canada, pp. 302–309.

El-Hakim, S.F., L. Gonzo, M. Picard, S. Girardi, A. Simoni, E. Paquet, H. Viktor, and C. Brenner, 2003b. Visualization of highly textured surfaces, *Proceedings of the International Symposium on Virtual Reality, Archaeology and Intelligent Cultural Heritage (VAST2003)*, Brighton, U.K., 05–07 November, pp. 231–240.

Früh, C., R. Sammon, and A. Zakhor, 2004. Automated texture mapping of 3D city models with oblique aerial imagery, *Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT)*, Thessaloniki, Greece, pp. 396–403.

Grammatikopoulos, L., I. Kalisperakis, G. Karras, T. Kokkinos, and E. Petsa, 2004. On automatic orthoprojection and texture-mapping of 3D surface models, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 35(Part 5):360–365.

Hoffman, G.R., 2004. Product generation using ADS-40 digital imagery, *Proceedings of the ASPRS Annual Conference*, May 2004, Denver, Colorado, unpaginated CD-ROM.

Katzenbeisser, R., 2004. Calibration and data validation of a LIDAR fiber scanner, *Proceedings of the ASPRS Annual Conference*, May 2004, Denver, Colorado, unpaginated CD-ROM.

Kuzmin, Y.P., S.A. Korytnik, and O. Long, 2004. Polygon-based true orthophoto generation, *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 35(Part 3):529–531.

Mavromati, D., E. Petsa, and G. Karras, 2002. Theoretical and practical aspects of archaeological orthoimaging, *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 34(Part 5):413–418.

Neugebauer, P., and K. Klein, 1999. Texturing 3D models of real world objects from multiple unregistered photographic views, *Proceedings of Eurographics, Computer Graphics Forum*, 18(3): 245–256.

Oda, K., W. Lu, O. Uchida, and T. Doihara, 2004. Triangle-based visibility analysis and true orthoimage generation, *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 35(Part 3):623–628.

Palà, V., and R. Arbiol, 2002. True orthoimage generation in urban areas, *Proceedings of the 3rd International Symposium on Remote Sensing of Urban Areas*, Istanbul, 11–13 June, Vol. 1, pp. 309–314.

Pollefeys, M., R. Koch, M. Vergauwen, and L. van Gool, 2000. Automated reconstruction of 3D scenes from sequences of images, *ISPRS Journal of Photogrammetry and Remote Sensing*, 55:251–267.

Poulin, P., M. Ouimet, and M.-C. Frasson, 1998. Interactively modeling with photogrammetry, *Proceedings of the Eurographics Workshop on Rendering*, pp. 93–104.

Pulli, K., H. Abi-Rached, T. Duchamp, L.G. Shapiro, and W. Stuetzle, 1998. Acquisition and visualization of colored 3-D objects, *Proceedings of the IEEE International Conference on Pattern Recognition*, Brisbane, Australia, 16–20 August, pp. 99–108.

Rau, J., N. Chen, and L. Chen, 2002. True orthophoto generation of built-up areas using multi-view images, *Photogrammetric Engineering & Remote Sensing*, 68(6):581–588.

Rau, J.Y., and L.C. Chen, 2004. Geometrical building modeling and its application to the ortho-rectification for aerial images, *Journal of Photogrammetry and Remote Sensing*, 9(1):53–76.

Rocchini, C., P. Cignoni, C. Montani, and R. Scopigno, 2002. Acquiring, stitching and blending diffuse appearance attributes on 3D models, *The Visual Computer*, 18:186–204.

Rognant, L., J.G Planès, M. Memier, and J.M. Chassery, 2000. Low cost precise urban orthophotos, *Proceedings of the ASPRS Annual Conference*, Washington, D.C., 22–26 May, unpaginated CD-ROM.

Schickler, W., and A. Thorpe, 1998. Operational procedure for automated true orthophoto generation, *International Archives of Photogrammetry and Remote Sensing*, 32(Part 4):527–532.

Sheng, Y., P. Gong, and G.S. Biging, 2003. True orthoimage production for forested areas from large-scale aerial photographs, *Photogrammetric Engineering & Remote Sensing*, 69(3):259–266.

Strecha C., T. Tuytelaars, and L. van Gool, 2003. Dense matching of multiple wide-baseline views, *Proceedings of the 9th IEEE International Conference on Computer Vision,* Cannes, France, Vol. 2, pp. 1994–1201.

Visnovcova J., Z. Li, and A. Grün, 2001. Generating a 3D model of a Bayon tower using non-metric imagery, *Proceedings of the International Workshop on Recreating the Past – Visualization and Animation of Cultural Heritage*, Ayutthaya, Thailand, 26 February – 01 March, unpaginated CD-ROM.

Wang, L., S.B. Kang, R. Szeliski, and H.-Y.Shum, 2001. Optimal texture map reconstruction from multiple views, *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR '01)*, Vol. 1.

Wanshou, J., and Z. Yixuan, 1999. The making of high precision orthoimage of ancient buildings, *Proceedings of the Asian Conference of Remote Sensing (ACRS 1999)*, URL: *http://www.gisdevelopment.net/aars/acrs/1999/ps4/ps4245.asp* (last date accessed: 08 January 2007).

Zhou, G., W. Schickler, A. Thorpe, P. Song, W. Chen, and C. Song, 2004. True orthoimage generation in urban areas with very tall buildings, *International Journal of Remote Sensing*, 25: 5163–5180.

(Received 24 August 2005; accepted 04 October 2005; revised 15 November 2005)