Polygon-based Texture Mapping for Cyber City 3D Building Models

Fuan Tsai and Hou-Chin Lin

Center for Space and Remote Sensing Research
National Central University
300 Zhong-Da Rd., Zhong-Li, Taoyuan 320, Taiwan
Tel: +886-3-4227151 ext. 57619
Fax: + 886-3-4254908
Email: ftsai@csrsr.ncu.edu

To appear in: International Journal of Geographical Information Science (accepted in Sep., 2006)

Polygon-based texture mapping for cyber city 3D building models

Fuan Tsai* and Hou-Chin Lin

Centre for Space and Remote Sensing Research, National Central University, Taiwan

Abstract

Three-dimensional building model is one of the most important components in a cyber city implementation. Currently, however, most building models do not have sufficient and accurate texture information. The lack of texture not only makes 3D building models less realistic in visualization, it may also fail to provide needed information in intricate applications. This study developed a polygon-based texture mapping system to produce near photo-realistic texture mappings for 3D building models. Textures of building exteriors were generated from mosaics of close-range photographs acquired with commodity digital cameras. The developed system integrated multiple digital photographs to create texture mosaics that were continuous in geometric outlines and smooth in colour shadings and correctly mapped them onto corresponding building model façades A test example demonstrated that the resultant building model had more complete and accurate texture features as well as near photo-realistic appearance.

Keywords: texture mapping; visualization; building modelling; cyber city; image mosaicking

1. Introduction

As sciences and technologies advance in remote sensing and related fields, their applications have become more sophisticated. Among them, three-dimensional (3D) geoinformatics is one of the emerging and fast growing topics that have attracted attentions and development efforts from researchers and scientists in these fields. The revolutionary power of computer graphics, visualization, and other information technologies further extends 3D geoinformatics into a more diversified academic and industrial sector. Cyber city is one of the most sophisticated examples of 3D geoinformatic systems and applications. A cyber city resembles the layouts, activities, and functionalities of a real-world community. Implementation of cyber city requires a comprehensive integration of remote sensing, geographic information systems (GIS), and information technologies. It has been identified as one of the most appealing challenges in the research and development of geoinformatics (McEachren & Kraak, 2001; Kraak, 2002).

The fundamentals of cyber city undoubtedly rely on the accurate reconstruction of 3D terrain and building models as well as realistic texture mapping of model surfaces. As remotely sensed data acquired from new generations of high resolution sensors and new types of data (such as LIDAR and three-line scanners) become ubiquitous, 3D terrain and building models can be constructed effectively. Algorithms developed for this purpose have been proposed and achieved various degrees of success.

* Corresponding author: Email: ftsai@csrsr.ncu.edu.tw; Tel.: +886-3-4227151-57619; Fax: + 886-3-4254908

For example, Lu et al. (2006) and Rau and Chen (2003) reconstructed building models by extracting 3D boundary lines from aerial photographs, whereas Gunadai et al. (2002) and Zhang et al. (2005) integrated LIDAR with other data sets for building model generation. However, commonly used airborne and satellite remote sensing data can only provide limited texture information of buildings primarily because of the restriction in sensor looking angles. Therefore, current 3D building models often do not have sufficient texture information about the exteriors or façades of buildings. Some city model visualization systems attached pseudo texture images onto building models for a better presentation (Beck, 2003), but they did not represent true building textures.

The lack of accurate texture attributes not only made 3D building models less realistic, but sometimes could also seriously hinder their usability. To address this issue, this study developed a polygon-based system to produce near photo-realistic texture mapping of 3D building models using close-range photographs collected with commodity digital cameras. The objective was to create an effective and efficient texture mapping system that can generate texture images of 3D building models as complete and seamless as possible and to map them onto corresponding model objects correctly.

2. Texture mapping of 3D building models

To generate complete texture information of a building façade from digital photographs, it commonly requires the integration of multiple images. Image mosaicking has been used in a variety of visual-related applications. A novel technique for generating image mosaics of real-world environment was to merge sequences of video frames (Chon et al., 2004; Gibson et al., 2003; Nicolas, 2001). However, general image mosaicking algorithms were not designed for photo-realistic texture mapping and might not fulfil the requirements of creating complete and seamless building texture information, which is critical in cyber city visualization and applications. In addition, directly using (airborne) video sequences for building texture often requires intensive manual treatments to eliminate occlusions, blurs and other artefacts (Zhang et al., 2005).

Another popular approach, especially for displaying open and wide spaces, was using panoramic views created from stitched photographs (Coorg and Teller, 2000; Peleg et al., 2000; Zhu et al., 2001). Zheng and Shi (2003) further developed a "route panorama" to present photo-realistic street views. Nevertheless, although panoramic visualization can generate nice scenes, it displays only facial appearances instead of modelling accurate building textures (Pontinen, 2004).

For 3D building texture mapping, because individual digital texture photographs are usually taken at different viewing conditions (view points, looking angles, zoom factors etc.), they are of assorted perspectives, scales, brightness, contrasts, colour shadings, and other properties. These variations need to be adjusted in order to integrate into a seamless mosaic. Therefore, the first challenge of 3D building texture mapping is to merge images pertaining to the same building façade into a complete texture mosaic that is continuous in geometric outlines and in colour shadings.

In this regard, twin snakes algorithm (Kerschner, 2001) was a popular approach to detect seams in ortho-images for mosaicking of remote sensing imagery, but it is apparently not appropriate for building texture mosaicking because the original texture images are usually not orthorectified. If the camera parameters are known, geometric distortions of images can be corrected using photogrammetric methods of perspective photo mapping (Huang, 2001; Spann & Kaufman, 2000). Unfortunately, most commodity digital cameras do not provide complete viewing parameters unless augmented with additional (and often expensive) equipments such as GPS, INS, and digital compass (Szeliski and Shum, 1997; Tian et al., 2003).

For images with unknown or incomplete viewing parameters, the geometries of buildings in images might be approximated if geometric constrains are well known (van den Heuvel, 1998), but high-precision photogrammetric corrections (orthorectification) will be difficult, if not impossible, to accomplish. A few algorithms were developed to work out original or related camera parameters, projection geometry, and pose. For example, Coorg and Teller (2000) estimated camera positions and rotations from correlations of overlapped images. Vision-based modelling methods were also suggested for obtaining relative pose between the cameras and 3D scene geometry from motion imagery (Kumar et al., 2000). It was also demonstrated (Lee et al., 2002) that sensor models can be reconstructed by computing vanishing points (Caprile & Torre, 1990; Guillou et al., 2000) of building line segments in images. Zhang (2000) presented a technique of using a planar pattern to calibrate cameras in computer vision systems. The same technique was also applied in an augmented virtual environment for dynamically visualizing moving objects and 3D scenes (Neumann et al., 2004).

Debevec (1996) and Fu (2002) both identified building boundaries from images to determine facets of the building and to map corresponding texture blocks to model surfaces from cropped areas selected from an image spool. Another approach was using highly textured points as seed points to obtain relationships between two overlapped images (Kim et al., 2003). A similar technique was also applied in the frequency domain to integrate multiple images (Xie et al., 2003). Although these approaches might be solutions to concerns in geometric space, they do not address issues in colour shadings, nor could they effectively deal with problems of shadows, blurs and blocked-areas (occlusions) of building appearance.

For realistic texture mapping using mosaics of close-range images, besides the geometric correction or registration, variations in colour space of individual images also need to be minimized. A commonly adopted technique was using histogram matching or equalization to force colour and shading distributions of candidate images to be within the same range (Du et al., 2001). However, directly apply this method to close-range images for texture generation may cause serious misrendering of colour shadings and sometimes produce results of poor quality such as hazy or low-contrast images.

The principle of seamless image mosaicking is smoothing the transition between overlapped areas of images. For example, feathering (alpha blending) weights the inputs of images as a function of distances (from the seamline). Burt and Adelson (1983) demonstrated a multi-resolution spline algorithm for smooth image mosaic. Adelson et al. (1984) further presented a pyramid blending algorithm that used different alpha masks in different bands. Levin et al. (2004) developed two complex image stitching algorithms in the gradient domain to optimize the mosaicking quality. These are all effective methods, but they all require intensive computation and most of them can only deal with a pair of input images at a time.

Some researchers also suggested using weighted image blending techniques to create panoramic and spherical mosaic images (Coorg & Teller, 2000; Efros & Freeman, 2001; Nicolas, 2001; Shum & Szeliski, 2000; Uyttendaele et al., 2001). Although effective in generating general panorama, these methods may not be adequate for building texture mapping. Other than that panoramic views usually do not have actual model characteristics, another reason is that original texture images for building façades may have shadows or portions blocked by foreign objects (occlusions). These non-interested regions should not be included in the mosaic in order to produce the best and most comprehensive texture information.

Regardless the method used for correction, adjustment, and mosaicking, textures need to be mapped onto their corresponding 3D building façades The mapping process involves transformations from an image or texture space to the object (model) space. Some accomplished this by interactively or semi-automatically selecting appropriate texture blocks from an image spool and mapping them to the correct positions, orientations, and attitudes in the object space (Fu, 2002; Huang, 2001). Others opted for an reverse course, i.e. transforming object models to fit them to a texture image according to viewing parameters of the image (Haala & Bohm, 2003). The main disadvantage of the former method is that it does not deal with the incompatibility of colour shadings in different texture blocks. On the other hand, the later approach works only for a fixed view and requires a full set of known camera parameters for each texture image, which, as mentioned previously, are difficult to obtain if using commodity digital cameras for image acquisition. Debevec et al. (1998) presented an image-based rendering algorithm for generating view-dependent texture mappings of architectural scenes from multiple images based on the projective texture mapping algorithm (Segal et al., 1992). Their method is capable of filling "holes" of the texture, but only works when the locations and geometries of original photographs are known and requires ray-tracing for the visibility test.

The issues described above must be addressed in order to make efficient and realistic visualization of 3D cyber city buildings. Therefore, this study developed a polygon-based approach designed specifically for cyber city 3D building model texture mapping. The remaining of this paper will describe and discuss the developed system in detail.

3. Polygon-based texture mapping system

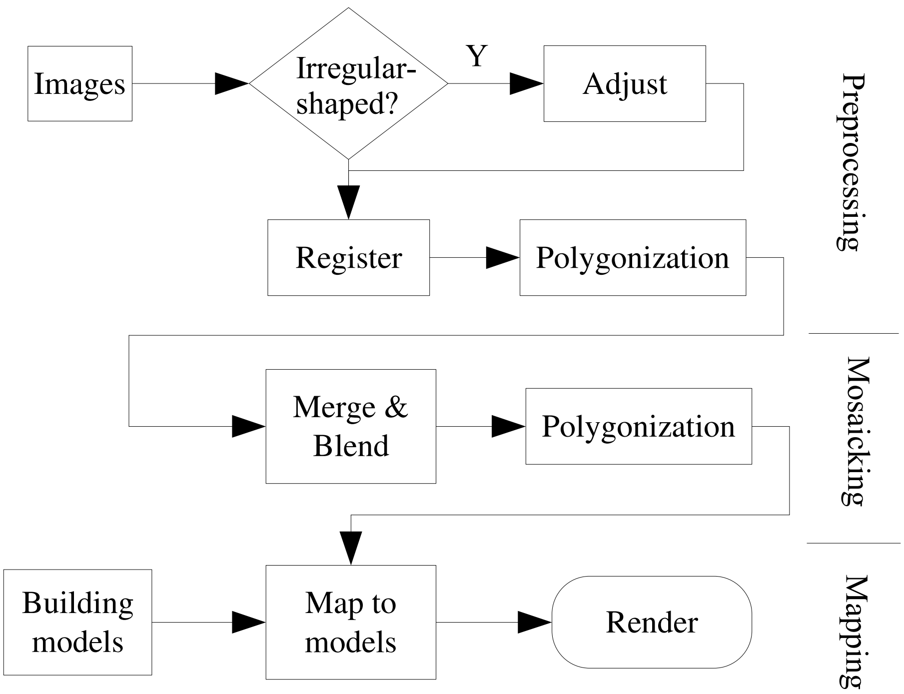


Figure 1: Polygon-based texture mapping processing pipeline.

The polygon-based texture mapping approach employed in this study is a hybrid system comprised of image adjustment, merging, blending and mapping transformation.

The general procedure is illustrated in Figure 1. The processing pipeline can be roughly categorized into three phases. As shown in the diagram, digital photographs were first registered and polygonized in the preprocessing phase. Then they were merged into mosaics and polygonized again. Finally, complete textures were mapped to the model (object) space polygon by polygon.

3.1. Preprocessing

Besides image acquisitions and other data preparations, two operations should be applied to original digital photographs before passing down to the texture mapping system's processing pipeline. They were registration and polygonization. The main purpose of registration was not merely to geometrically correct the images. Nor did it have to do with transforming texture features to the model coordinate system. Instead, it was to transform a group of photographs pertaining to the same object façade to a common texture space, so they could be merged together correctly. On the other hand, the function of polygonization was to identify and separate areas of interest (AOI) from individual texture images as polygons. These AOI polygons would then be used in the second processing phase as the basis for integration.

3.1.1. Image registration

Two types of registration were used in the system. First, for a set of overlapped photographs with close view-points and similar looking-angles, they could be registered to a "reference image" or "base image" if the distortions in geometry were not prominent. The registration was done by identifying tie points on overlaps of individual photographs and transforming them from individual image spaces to a common texture space using an eight-parameters transformation as described in Equation 1 and 2.

$$x = \frac{a_0 X + a_1 Y + a_2}{c_0 X + c_1 Y + 1} \qquad (1)$$

$$y = \frac{b_0 X + b_1 Y + b_2}{c_0 X + c_1 Y + 1} \qquad (2)$$

where X and Y are coordinates of a pixel in the image space; x and y are the transformed coordinates of the pixel in the texture space. The eight parameters, $<a_0\ a_1\ a_2\ b_0\ b_1\ b_2\ c_0\ c_1>$, of the non-linear system can be solved using least-squares-fitting from equations constructed with tie points. Details of solving Equation 1 and 2 with least-squares-fitting can be found in most Geomatics texts and are not repeated here. The advantage of this registration mechanism is that it requires only adequate tie points to construct the transform system; no camera parameters and other data are needed.

It is possible to automatically identify tie points with interest point detection algorithms, such as Harris Corner Detector (Harris & Stephens, 1988), and match them with Normalized Cross Correlation or similar techniques. However, when using digital cameras for image acquisition, the number of images (and tie points) required to generate complete textures for each building is limited. Therefore, it may not make much difference whether to specify the tie points interactively or by computer programs, in terms of time and effort used. For the examples presented in this paper, the tie points were selected interactively unless otherwise noted.

For images with large tilting angles or images with little or no overlaps, the register-to-image approach could not be applied because the geometric distortion was indisputable or no relationships could be established among tie points for the eight-parameters transformation to work. Consequently, they were registered to building models directly.

As illustrated in Figure-2, the control points used for this type of registration included roof and ground points of building corners and a few selected points in between. Coordinates of roof control points were looked up from building computer-aided design (CAD) layouts or blueprints, whereas the elevations of ground control points were supplied by digital terrain model (DTM) data. Coordinates of other points were approximated using interpolation. After coordinates of all control points were assigned, the image was transformed (warped) to the same texture space which in this case was identical to the model space. The disadvantage of applying this registration approach was that because some of the control points were generated by interpolation, minor displacements of the control points might appear after transformation and caused "ghost artefacts". In addition, the bottom portion of a texture image is more likely to be blocked by trees or other objects; therefore, it may be difficult to identify ground control points on the image.

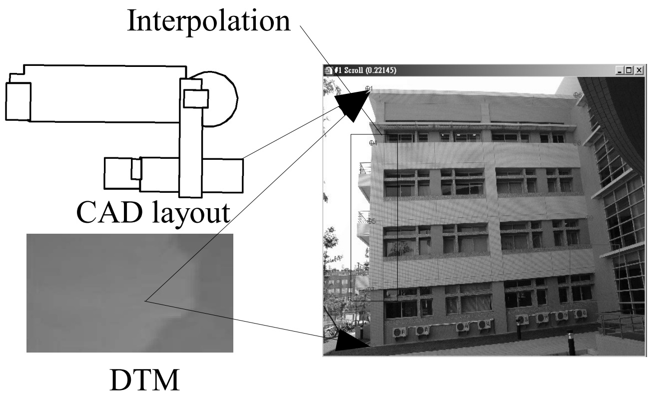


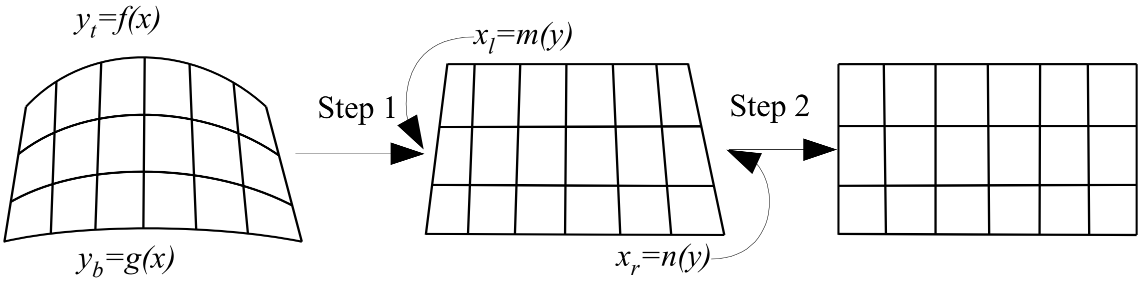
Figure 2: Register to model.

3.1.2 Adjustment of irregular-shaped façade texture images

If the texture images consist of irregular-shaped (non-planar) façades, they need to be adjusted before registration. Figure-3 shows an example with a semi-cylindrical structure and its adjusted result. Because geometric distortions of this kind of texture images depend highly on the looking angles, it is difficult to directly register them either to a base-image or to the model. In this study, texture images of irregular-shaped building façades were processed with a two-steps adjustment (Figure 4) before registration. The first step was to approximate the top and bottom lines of the façade with polynomial functions and then stretched the image along its vertical axis so that the top and bottom curves of the façade were pulled into horizontal lines as illustrated in Figure-4b and Equation 3. Secondly, the image was stretched horizontally until the vertical lines were aligned correctly (Figure-4c and Equation 4). As displayed in the adjustment example (Figure-3b), most of the semi-cylindrical façade was transformed to an orthogonal space and ready to register. However, textures near both the right and left boundaries were still distorted and should be detached from the image.

(a) (b)

Figure 3: Original texture image for an irregular-shaped facade (a) and its adjusted result (b).



(a) (b) (c)

Figure 4: Two-step adjustment of irregular-shaped facade images.

$$y_j = y_t + \frac{y_t - y_b}{MAX(y_t - y_b)} j \qquad (3)$$

$$x_i = x_l + \frac{x_r - x_l}{MAX(x_r - x_l)} i \qquad (4)$$

where y_b, y_t, x_l, and x_r are polynomial functions to describe the bottom, top, left and right boundaries of the target texture block; i and j represent the pixel indices in horizontal and vertical (parametric) directions. $MAX(y_t\text{-}y_b)$ is the maximum distance between the top and bottom polynomials, while $MAX(x_r\text{-}x_l)$ is the maximum distance between the left and right polynomials. Equation-3 deforms the image so that horizontal line features of the building level out. Vertical line features further straighten up in Equation-4.

3.1.2. Polygonization

After registration, the next step was to identify AOI's on the transformed images and create their enclosing polygons. The main purpose of this process was to exclude textures that did not belong to the same façade but are shown on the same image. The AOI's would then be used as texture blocks for merging and blending. One thing to note in the polygonization process is that, shadows and occlusions can be specified as negative polygons (or "holes" inside a polygon). An advantage of treating shadows and occlusions as negative polygons is that it can avoid creating too many fragmented AOI's on a single image.

For better performance, edge detection was applied to the transformed images and blocks of interested façades were cropped out as the preliminary AOI's. Then, non-interested regions were identified and removed from the AOI's. For example, trees could be classified using Greenness Index (GI); other foreign objects and shadows could be identified using region growing or morphological operations. However, a disadvantage of using automated algorithms for identifying and removing non-interested regions is that these algorithms tend to generate scattered small polygons. On the other hand, interactively specifying unwanted features as negative polygons will produce more pristine façade texture images.

3.2. Mosaicking

Texture blocks belonging to the same façade were integrated in this phase to generate an image mosaic. The objective was to create complete texture images continuous in both geometric outlines and in the colour domain for all building façades The registration should address most of the geometric continuity problems, so the emphasis was placed on minimizing seams in colours and shadings. This study employed a polygon-based fast algorithm to integrate building texture blocks from multiple images and to blend colour and shading properties of overlapped areas simultaneously. The algorithm is basically an extension of alpha blending. In alpha blending, the transition of pixel grey values depends solely on the distance of a pixel to the seam. However, it tends to leave artefacts in the mosaic, if the original images are not aligned perfectly (Levin et al., 2004). In addition, if one or more of the overlapped AOI's are extraordinarily larger or smaller, their impacts are not necessarily carried over into the weightings. To reduce these drawbacks (especially the later), the blending algorithm developed in this study also takes the distance to the centroid of an AOI into account. The algorithm can be described as a simple equation:

$$P_m = \sum \left(w_i P_i \right); \quad w_i \propto \left(\frac{d_{ei}^a}{d_{ci}^b} \right); \quad \sum w_i = 1 \qquad (5)$$

For a pixel of the image mosaic, P_m, its new grey value in each colour band was determined by a weighted sum of all corresponding pixels in the texture image spool. The idea is that if a pixel is closer to the centroid of a polygon (AOI), that polygon should be given more weights. On the other hand, if a pixel is located near an edge of a polygon, that polygon should have less contribution to the overall weighting. Accordingly, the weighting is proportional to the minimum distance to edge, d_e, and to the inverse of the distance to centroid, d_c. The power factors, a and b, are constants and can be specified dynamically according to the degree of smoothness desired. In our tests, linear weighting for both distances (i.e. a=b=1) was good enough to generate visually smooth mosaics in most cases. On special occasions, higher orders (such as quadratic or cubic) of a and/or b might be required to make the transitions more plausible.

3.3. Map to model

The mosaicking process created texture images for building façades, but they still needed to be adhered to the model surfaces. Because the original images might consist of side-looking textures of other façades, the final mapping operation was also performed in a polygon-basis. In other words, only portions of the texture needed to be attached to the building model facets block by block. Therefore, a second polygonization should be performed before the mapping transformation.

Theoretically, if original photographs were registered to models, the texture mosaic will inherit correct coordinates in the model (object) space. In this case, polygons of AOI's are already mapped to their correct positions in the model space. However, if the texture mosaic was generated from photographs registered to a reference image or a coordinate system other than the object coordinate system of the model, the AOI's on the mosaic of texture images need to be transformed from their texture space to the object space. For regular-shaped (rectangular or other planar) façades, the mapping process is usually a simple linear transformation. Nevertheless, for irregular-shaped or non-planar façades, the transformation was carried out in parametric coordinate systems instead of the world coordinate system, so the transformation required only rotations, translations, and scales to avoid complex non-linear operations.

4. Results and discussions

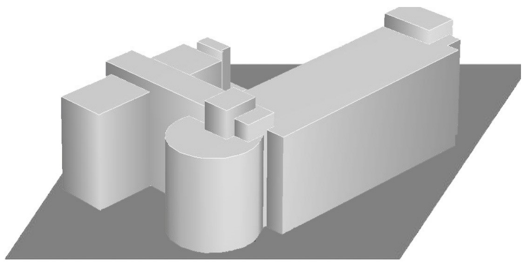


Figure 5: Test building model.

A fairly complicated building (Figure-5) was used to test the performance of the developed polygon-based texture mapping system. The building model was created using LIDAR data and aerial photographs in conjunction with DTM and building CAD layout files. Texture images for all building façades were acquired using a Sony digital camera.

Figure-6 shows three digital photographs pertaining to the same building face and a texture mosaic of them is displayed in Figure-7. The middle image of Figure-6 was used as the base image to which the other two images were registered. The result indicated that the developed polygon-based mosaic processing was very effective.

Figure 6: Three texture images to mosaic.

As can be seen in Figure-7, the mosaic had continuous outlines in geometry and the colours and shadings were balanced and integrated seamlessly. The discontinuity in colours and shadings were minimized with the polygon-based algorithm according to the distance parameters as previously described in Equation-5.

For comparison, another mosaic of Figure-6 is displayed in Figure-8, in which the colour shadings were blended simply using the average of pixel grey values in overlapped areas. Closely examine the areas indicated with the arrow marks (which are boundaries of overlapped areas of original texture polygons). In Figure-7, the colour shadings are smooth and continuous (at least visually) across overlapping edges of polygons, but the discontinuities (seams) are evident in Figure-8.

Figure 7: Mosaicking and blending result using distance to centroid and minimum distance to edge. The mosic is continuous in geometry and colour domains.

Figure 8: Mosaicking and blending result using average. The seams are evident in this mosaic.

One may notice that there are ghost artefacts (for example, the second left tree) appearing in above texture mosaic examples). This was caused by spatial distortions or deviations in geometric registration of original texture photographs. The polygon-based merging algorithm might help reduce some of the artefacts. Nonetheless, a rigorous geometric correction is more likely to be the ultimate solution to this problem.

Another texture mosaic example is shown in Figure-9 and Figure-10. In this case, the left and right images in Figure-9 were registered to the middle image before mosaicking. The three original texture photographs all had considerable occlusions. Like the previous example, the resultant texture mosaic of AOI's (Figure-10) was also geometrically aligned and with seamless colour shadings after the polygon-based merging and blending process. More importantly, the texture mosaic of target AOI's was more complete and with minimum blocked areas or shadows. This example further elaborates the effectiveness of the polygon-based mosaic process.

Figure 9: Original texture images to mosaic.

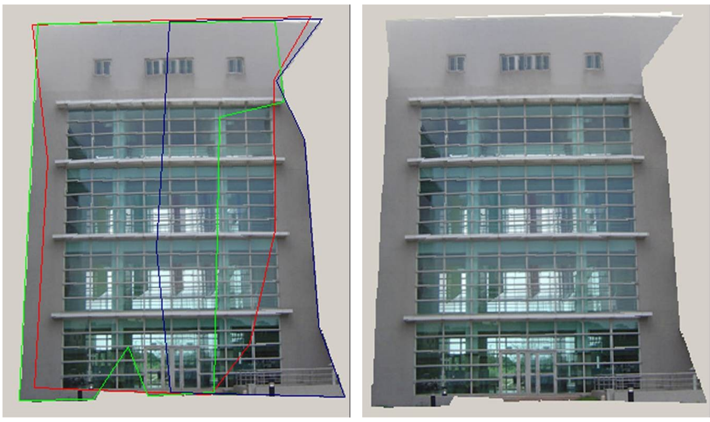


Figure 10: Mosaic image of Figure 9 with (left) and without (right) their original AOI polygons.

Apart form the smooth blending in colour shadings, another advantage of using the developed polygon-based mosaicking algorithm is that it is fast and computationally efficient. In comparison with the spline-based and image pyramid approaches (Adelson et al., 1984; Burt and Adelson, 1983), which require complicated Lagrange operations to repeatedly split and merge images, or the Gradient-domain Image Stitching methods (Levin et al., 2004), the proposed approach is more straightforward and requires little computation.

For instance, whether a pixel is located inside a polygon or not can be quickly determined by counting the number of intersections of a half-infinite vertical or horizontal line (originating from the pixel) and edges of the polygon. If the number of intersections is odd, the pixel is inside; otherwise, it is outside. This mechanism also works on complex polygons and is very useful when dealing with shadows and occlusions in texture mapping. As mentioned previously, occlusions and other non-interested areas can be treated as negative polygon boundaries or holes inside an AOI polygon and the rule of inside-outside test still applies. With the test, a pixel located inside a hole will be identified as outside AOI polygons and will be skipped during the blending process. Also, the distance from a pixel to a boundary line segment (for finding the minimum distance to edge in Equation-5) can be calculated using simple geometry or even approximated with the distance to a nearest vertex. This provides a significant advantage in large-scale cyber city applications.

With the polygon-based algorithm, complete texture mosaics were constructed for all façades of the test building model. They were then mapped onto corresponding model facets using the mapping transformation algorithm as described in 3.3. For the test example, most of the model facets were rectangular, so mapping required only linear transformations. However, the semi-cylindrical structure of the test building model was a special case. In computer graphics, sophisticated methods were developed for texture mapping an image to complicated surfaces (e.g. Guo et al., 2005). However, although irregular-shaped, the building façades can be transformed to a planar parametric space. Therefore, in this study, linear transformations could still be used for the texture mapping of irregular-shaped façades, if they were carried out in planar parametric space.

The complete texture mosaic for this part of the building was generated from six overlapped digital photographs. The original texture photographs were adjusted using the two-step adjustment procedure described previously. They were then merged and blended to form a complete mosaic as displayed in Figure-11 in a parametric coordinate system.

Figure 11: Mosaicking result of the semi-cylindrical structure in a parametric coordinate system.

The horizontal axis in Figure-11 represents the central angle, θ, and the vertical axis is the building height, h. The texture mosaic can be directly mapped onto the semi-cylindrical surface using linear transformations in parametric space since the coordinates of a point on the surface can be simply represented as $[r\cos\theta, r\sin\theta, h]$, where r is the radius of the structure cross-section and can be obtained from CAD layout data or measured from building plans. This approach avoids not only complex non-linear transformations but also the monotonous manual selection of control points for constrained texture mapping (Guo et al., 2005), thus minimizing the decrease in performance.

Figure 12 and 13 demonstrate a few perspective views of the final texture mapping result of the test building. As displayed in the figures, the textures are complete and align correctly with the model. The shadings and colours of the texture are smooth. In general, the example demonstrated here proves that the developed polygon-based system is an economical approach to effectively and efficiently produce near photo-realistic texture mapping for cyber city 3D building models.

The most noticeable artefact in Figure-11 and Figure-13 is probably the rippled pattern of the aliasing effect in the texture. This aliasing side effect was a by-product of resampling during mosaicking or transformation in the texture mapping system. There are algorithms and techniques to alleviate the aliasing effect, but they are beyond the scope of this research and are not discussed in this paper.

Because some minor structures (for example, the verandas and balconies) were not presented in the building model, they were treated as parts of a facet texture instead of independent texture blocks. Since the original texture images were not orthorectified, the tilt-view effects (e.g. seeing ceilings and bottoms of the balconies) are still visible in the texture mosaics and mapping results.

In addition, although most of the non-interested regions were successfully cleaned off, a small number of them (two palm trees, a red signboard, a lamppost, and part of a car) were not removed from the texture. Completely eliminating these objects would generate more fragmented small AOI polygons during mosaicking and required more raw texture photographs in different view angles, but the improvement in the texture mapping quality of the whole building model would be insubstantial.

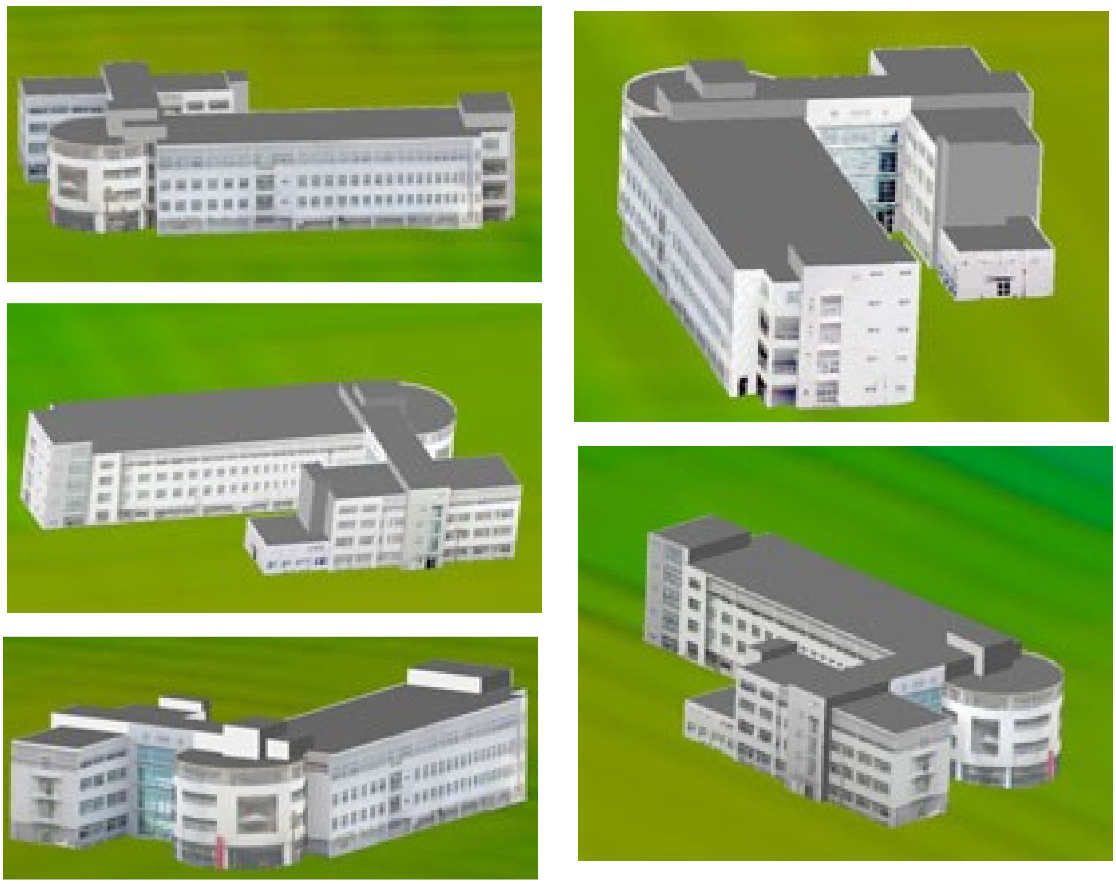


Figure 12: Perspective views of complete texture mapping result of the test example.

Figure 13: Enlarged perspective view of the texture mapping result.

5. Conclusions and future work

This study developed a polygon-based texture mapping system for cyber city 3D building models. The developed polygon-based algorithm and procedure can generate complete and seamless texture mosaics of building façades from multiple close-range digital photographs and correctly map the texture mosaics onto their corresponding model façades The algorithm for image mosaic is simple and straightforward, so it can provide high computational efficiency. The mapping of textures to model objects is also independent to viewing parameters of visualization and requires only linear transformation in most cases, even for irregular-shaped or non-planar building façades The test example demonstrates that the developed polygon-based texture mapping system can produce complete, coherent and near photo-realistic appearance of 3D building models effectively and efficiently. This should have a significant contribution to the improvement in reality and practicality of cyber city implementations and applications.

Although the developed system is an effective texture mapping approach for cyber city building models, there are still rooms for improvement. For example, a more rigorous and automated geometric correction will definitely increase the quality and efficiency of image mosaicking in the system. Currently, it takes about an hour to texture the test building (not including image acquisition time) for an experienced operator. The bottleneck of the system is identifying tie points for image registration. An interest-point detection and matching system for automatic identification of tie points is under development. More works have also been planed to enhance the quality of using GI and image morphology for automatic removal of non-interested regions from texture images. In addition, when the developed system is deployed in a large-scale cyber city implementation, other issues such as level of details (LOD) should also be addressed to further enhance the performance. These will be the priorities for our future researches in texture mapping of cyber city 3D building models.

Acknowledgements

The authors would like to thank Prof. Linag-Chien Chen and the staff of Digital Photogrammetry Laboratory of CSRSR, NCU, Taiwan for their kindly assistances in photogrammetry and image registration. This research was supported in part by the NCU-ITRI Joint Research Centre under project number NCU-ITRI-930304 and the National Science Council of Taiwan ROC (NSC-94-2211-E-008-031).

References

Adelson, E. H., C. H. Adelson, J. R. Bergen, P. J. Burt, and J. M. Ogden, 1984, Pyramid method in image processing. *RCA Engineer*, 29(6), pp. 33-41.

Beck, M., 2003, Real-time visualization of big 3D city models. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XXXIV-5/W10, CD-ROM.

Burt, P. J. and E. H. Adelson, 1983, A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics*, 2(4), pp. 217-236.

Caprile, B. and V. Torre, 1990, Using vanishing points for camera calibration. *International Journal of Computer Vision*, vol. 4, pp. 127-140.

Chon J., T. Fuse and E. Shimizu, 2004, Urban visualization through video mosaics based on 3-D multibaselines. *International Archive of Photogrammetry and Remote Sensing*, vol. XXXV-B3, pp. 727-731.

Coorg, S. and S. Teller, 2000, Spherical mosaics with quaternions and dense correlation. *International Journal of Computer Vision*, 37(3), pp. 259-273.

Debevec, Paul E., 1996, Modeling and rendering architecture from photographs. Ph.D. Dissertation, Department of Computer Science, University of California at Berkeley, CA USA.

Debevec, P.E., Y. Yu, and G. Borshukov, 1998, Efficient view-dependent image-based rendering with projective texture mapping. *Proc. 9th Eurographics Workshop on Rendering*, June 29 – July 1, 1998, Vienna, Austria.

Du, Y., J. Cihlar, J. Beaubien, and R. Latifovic, 2001, Radiometric normalization, composition, and quality control for satellite high resolution image mosaics over large area. *IEEE Transactions on Geoscience & Remote Sensing*, 39(3), pp. 623-634.

Efros, A and W. Freeman, 2001, Image quilting for texture synthesis and transfer. *Proc. ACM SIGGRAPH 2001*, pp. 341-346.

Fu, B.-G., 2002, Automatic perspective texture mapping on 3D building models. MS Thesis, Department of Geomatics, National Cheng-Kong University, Tainan, Taiwan. (in Chinese, with English abstract.)

Gibson, S., B. J. Hubbold, J. Cook, and T. L. J. Howard, 2003, Interactive reconstruction of virtual environments from video sequences. *Computer & Graphics*, 27(2), pp. 293-391.

Guillou, E., D. Meneveaux, E. Maisel and K. Bouatouch, 2000, Using vanishing points for camera calibration and coarse 3D reconstruction from a single image. *The Visual Computer*, vol. 16, pp. 396-410.

Gunadi, C. R., H. Shimizu, K. Kodama and K. Aizawa, 2002, Construction of large-scale virtual environment by fusing range data, texture images, and airborne altimetry data. *Proc. 1st. Int'ernational Symposium 3D Data Processing*

Visualization and Transmission (3DPVT02), pp. 772-775.
Guo, Y., J. Wang, H. Sun, X. Cui and Q. Peng, 2005, A novel constrained texture mapping method based on harmonic map. *Computers & Graphics*, vol. 29, pp. 972-979.
Haala, N. and J. Bohm, 2003, A multi-sensor system for positioning in urban environments. *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 58, pp. 31-42.
Harris, C. and M. Stephens, 1988, A combined edge and corner detector. *Proc. 4th Alvey Vision Conference*, pp. 189-192.
Huang, W.-L., 2001, A study of facade texture mapping on 3D building model by close-range photogrammetry. MS Thesis, Department of Geomatics, National Cheng-Kong University, Tainan, Taiwan. (in Chinese, with English abstract.)
Kerschner, M., 2001, Seamline detection in colour orthoimage mosaicing by use of twin snakes. *ISPRS Journal of Photogrammetry and Remote Sensing*, 56(1), pp. 54-64.
Kim, D. H., Y-I. Yoon and J-S. Choi, 2003, An efficient method to build panoramic image mosaics. *Pattern Recognition Letters*, vol. 24, pp. 2421-2429.
Kraak, M.-J., 2002, Some aspects of geovisualization. *GeoInformatics*, 5(8) , pp. 26-37.
Kumar, R., H.S. Sawhney, Y. Guo, S. Hsu and S. Samarasekera, 2000, 3D manipulation of motion imagery. *Proc. 2000 International Conference on Image Processing (ICIP 2000)*, vol. 1, pp. 17-20.
Lee, S.C., S.K. Jung and R. Nevatia, 2002, Automatic integration of facade textures into 3D building models with a projective geometry based line clustering. *EUROGRAPHICS 2002*, 21(3), pp. 259-273.
Levin, A., A. Zomet, S. Peleg and Y. Weiss, 2004, Seamless image stitching in the gradient domain. *ECCV'04*, pp. 377-389.
Lu, Y. H., J. C. Trinder and K. Kubik, 2006, Automatic building detection using the Dempster-Shafer algorithm. *Photogrammetric Engineering & Remote Sensing*, 72(4), pp. 395-403.
McEachren, A. M. and M.-J. Kraak, 2001, Research challenges in geovisualization, *Cartography and Geographic Information Science*, vol. 28, pp. 3-12.
Neumann, U., S. You, J. Hu, B. Jiang and I.S. Sebe, 2004, Visualizing reality in an augmented virtual environment. *PRESENCE*, 13(2), pp. 222-233.
Nicolas, H., 2001, New methods for dynamic mosaicing. *IEEE Transactions on Image Processing*, 10(8), pp. 1239-1251.
Peleg, S., B. Rousso, A. Rav-Acha and A. Zomet, 2000, Mosaicing on adaptive manifolds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10), pp. 1144-1154.
Pontinen, P., 2004, On the geometrical quality of panoramic images. *International Archive of Photogrammetry and Remote Sensing*, vol. XXXV-B5, pp. 82-86.
Rau, J-Y. and L-C. Chen, 2003, Robust reconstruction of building models from three-dimensional line segments. *Photogrammetric Engineering & Remote Sensing*, 69(2), pp. 181-188.
Segal, M., C. Korobkin, R. van Widenfelt, J. Foran, and P. Haeberli, 1992, Fast shadows and lighting effects using texture mapping. P*roc. ACM SIGGRAPH'92*, pp. 249-252.
Shum, H-Y. and R. Szeliski, 2000, Construction of panoramic image mosaics with global and local alignment. *International Journal of Computer Vision*, 36(2), pp. 101-130.
Spann, J.R. and K.S. Kaufman, 2000, Photogrammetry using 3D graphics and projective textures. *IAPAS*, vol. 33, Amesterdam.

Szeliski, R. and H.Y. Shum, 1997, Creating full view panoramic image mosaics and environment maps. *Proc. ACM SIGGRAPH'97*, pp.251-258.
Tian, G. Y., D. Glehii, D. Taylor, 2003, Comprehensive interest points based imaging mosaic. *Patten Recognition Letters*, vol. 24, pp. 1171-1179.
Uyttendaele, M., A. Eden, R. Szeliski, 2001, Eliminating ghosting and exposure artifacts in image mosaics. *Proc. IEEE Conference on Computer Vision & Pattern Recognition (CVPR) II*, pp. 1141-1154.
van den Heuvel, F. A., 1998, 3D reconstruction from a single image using geometric constrains. *ISPRS Journal of Photogrammetry and Remote Sensing*, 53(6), pp. 354-368.
Xie, H., N. Hicks, G. R. Keeler, H. Huang, V. Kreinovich, 2003, An IDL/ENVI implementation of the FFT-based algorithm for automatic image registration. *Computers & Geosciences*, vol. 29, pp. 1045-1055.
Zhang, Y., Z. Zhang, J. Zhang and J. Wu, 2005, 3D building modelling with digital map, LIDAR data and video image sequences. *Photogrammetric Record*, 20(111), pp. 285-302.
Zhang, Z., 2000, A flexible new technique for camera calibration. *IEEE Trans. Patten Analysis and Machine Intelligence*, 22(11), pp. 1330-1334.
Zheng, J.Y. and M. Shi, 2003, Mapping cityscapes to cyber space. *Proc. 2003 International Conference on Cyberworlds (CW2003)*, pp. 166-173.
Zhu, Z.G., E. Riseman, A. Handson, 2001, Parallel-perspective stereo mosaics. *Proc. 8th International Conference on Computer Vision (ICCV2001),* vol. 2, pp. 723-730.