The 12th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST (2011)
M. Dellepiane, F. Niccolucci, S. Pena Serna, H. Rushmeier, and L. Van Gool (Editors)

Dense 3D point cloud generation from multiple high-resolution spherical images

Alain Pagani, Christiano Gava, Yan Cui, Bernd Krolla, Jean-Marc Hengen and Didier Stricker

German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany

Abstract

The generation of virtual models of cultural heritage assets is of high interest for documentation, restoration, development and promotion purposes. To this aim, non-invasive, easy and automatic techniques are required. We present a technology that automatically reconstructs large scale scenes from panoramic, high-resolution, spherical images. The advantage of the spherical panoramas is that they can acquire a complete environment in one single image. We show that the spherical geometry is more suited for the computation of the orientation of the panoramas (Structure from Motion) than the standard images, and introduce a generic error function for the epipolar geometry of spherical images. We then show how to produce a dense representation of the scene with up to 100 million points, that can serve as input for meshing and texturing software or for computer aided reconstruction. We demonstrate the applicability of our concept with reconstruction of complex scenes in the scope of cultural heritage documentation at the Chinese National Palace Museum of the Forbidden City in Beijing.

Categories and Subject Descriptors (according to ACM CCS): Vision and Scene Understanding [I.2.10]: 3D/stereo scene analysis—

1. Introduction

The generation of virtual models of cultural heritage objects and scenes is an attractive tool for documentation, preservation and promotion purposes. In particular, techniques for reconstructing small-sized objects in 3D using image information are getting accurate enough to produce useful models for archeology, architecture and educational applications. In such scenarios, the object of interest is often placed in a highly controlled environment where high-resolution images can be acquired. However, in practical, out-of-lab situations, this technology faces challenging issues. For large scale reconstructions, a very large number of images are required to cover the area to reconstruct, because each image sees only a small part of the scene. If the image acquisition takes place outdoor, the lighting conditions can produce strong variations of illumination between images.

The use of panoramic or spherical images for documentation has become a natural extension of the standard perspective images, leading to applications like *Google Street View*, where the user can visualize the area by navigating inside a spherical view. Acquiring high-resolution spherical images is nowadays practical, fast and not necessarily expensive.

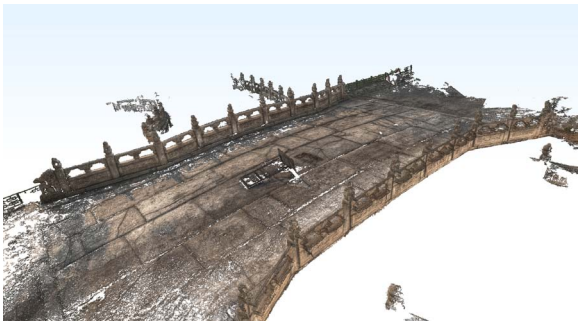

Figure 1: *Dense point cloud of a sculpted bridge. Our method automatically reconstructs this cloud from a set of spherical images.*

This can be done using a standard camera and specialized hardware like motorized spherical panorama heads and dedicated automatic software, or more conveniently by using complete hardware/software packages that automatize all the steps, like Weiss AG's Civetta Camera or SpheronVR's SpheroCam . In their current use however, spherical images serve mainly for direct visualization, *i. e.* after the acquisi-

© The Eurographics Association 2011.

DOI: 10.2312/VAST/VAST11/017-024

delivered by EUROGRAPHICS DIGITAL LIBRARY www.eg.org diglib.eg.org

tion, the scene can be observed only from the point of view the image has been taken.

In this paper, we present a technique that merges the ideas of image-based 3D reconstruction and spherical images. Instead of processing a huge number of standard perspective images, we propose to use as input data a few single but high resolution and high dynamic range images (HDRI). Additionally, we propose to opt for a different image geometry and replace standard perspective images by full spherical images that record the complete scene from one given point in space. A real scene is then captured with help of spatially distributed spherical images. This new kind of image data delivers far more information than a standard digital camera, as it captures the complete scene over an unique full sphere, provides high resolution(up to 14,000×7,000 pixels), and provides consistent and physically meaningful photometric information (HDRI).

1.1. Related work

Image-based 3D reconstruction techniques primarily focused on small-sized objects in perspective images [SCD*06]. The classical problems that need to be solved are (1) find the position and orientation of each camera and (2) compute dense point clouds from image correspondences, using the camera positions as prior information. To solve the first problem, Snavely *et al.* suggest an automated method for computing the cameras poses from perspective images [SSS06]. The second problem is addressed for example by Furukawa and Ponce [FP08], whose algorithm produces a dense point cloud from perspective images, assuming the camera positions are given. In order to achieve complete reconstructions of sites, a lot of images are necessary (1000 and up). For perspective images, this can be done automatically using for example the method of [VVG06]. Recent developments of the 3D reconstruction technology focus in trying to solve the numerous problems that arise when the number of images grows. Some researchers focus on simplifying the topology of unordered sets of images [SSS08], while others develop dedicated implementations in order to decrease the growing computational time. An interesting approach is to use Internet photo collections as input for 3D reconstruction algorithms [GSC*07]. Agarwal *et al.* [ASS*09] have shown that it is possible to use millions of photographs of a city to reconstruct some of its major monuments in less than 24 hours on a Computer Cloud of 500 nodes. Together with the method in [FCS*10], this can produce city-scale reconstruction if enough images are available. The computational demand can also be reduced to 24 hours on a single (high-end) PC, at the cost of quality decrease however [FGG*10].

The usage of wide-angle lenses and panorama for reconstruction has been only partially addressed. Studies on different lens models have shown that a wide angle camera model generally reduces drift accumulation in Structure from Motion techniques [SK05, KBK07], but the usage of multiple perspective cameras as one omnidirectional camera (as in [SY05]) only provides coarse reconstructions. In [MP06], a generic autocalibration method for omnidirectional, central projection cameras is derived, but the provided reconstructions remain quite sparse. Kim and Hilton propose a dense reconstruction technique for spherical camera pairs [KH10]. Different from our work, they use only two spherical views at a time for estimating depth information, considered as a narrow-baseline stereo system. Our method for the computation of the camera's pose and orientation works with wider distances between cameras and therefore relies on point matching between different views of the same objects. Epipolar geometry of spherical images is addressed in [LF05]. We extend this work by introducing a new distance for estimation of the epipolar matrix based on the spherical geodesic distance. In [WH06], the authors suggest that the use of spherical images for 3D reconstruction is possible. We follow this idea and derive the necessary pipeline for producing a dense point cloud out of spherical images. In the context of spherical images, Mauthner *et al.* present a technique for matching regions in omnidirectional views by generating perspective views from the spherical image [MFB06]. We adopt a similar technique, but generate many affine transformations instead of only one for increasing the chances of matching, in the spirit of A-SIFT [MG09]. In addition, our method generates point correspondences rather than region correspondences.

The work of Barazzetti *et al.* [BFRS10] shares different ideas with ours. The authors provide a way to derive 3D metric information from multiple spherical images. However, their approach for the computation of the pose transforms a spherical problem into a perspective one by reprojecting the sphere onto perspective images. In contrast, we use directly spherical coordinates for Structure from Motion. A second difference is that we provide a way to automatically compute dense point clouds, whereas the reconstruction in [BFRS10] mainly needs manual intervention from the user.

This paper contains several contributions. First, we show that accurate and dense reconstruction of large scale scenes is possible with tens of images instead of thousands. Second, we derive an algorithm for computing affine invariant SIFT features on color spherical images. Third, we propose a new metric based on geodesic distance on the sphere and derive the basic geometric tools for Structure from Motion using that metric. Fourth, we show how to optimize the quality of dense reconstruction by adapting the apparent object size in each view before the dense matching step.

The remainder of the paper is organized as follows: Section 2 presents the type of images we use as well as the pipeline of our method. We present our Structure from Motion algorithm for spherical images in section 3 and our Multiple View Stereo method in section 4. In section 5, we present our results in two application scenarios before concluding in section 6.

© The Eurographics Association 2011.

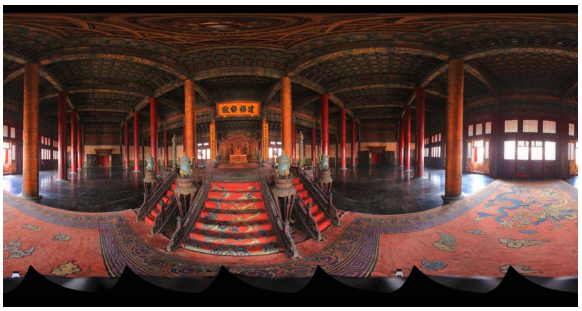

Figure 2: *Example of a spherical image: interior of the* Tai He Dian *temple.*

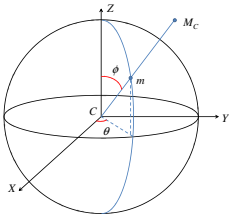

Figure 3: *Local camera coordinate system*

2. 3D reconstruction from spherical cameras

In this section, we present the general idea of our method as well as the different steps in our pipeline. Further details on each step can be found in the subsequent sections.

2.1. Spherical images

As mentioned previously, we use dedicated hardware for acquiring the images. Here we do not focus on the acquisition step, but rather describe the spherical image as input for our algorithms (see Figure 2). The spherical images we use are equivalent to an environment map, that is an image that represents an omnidirectional view of a three-dimensional scene as seen from a particular 3D location. Every pixel in the image corresponds to a 3D direction, and the data stored in the pixel represents the amount of light arriving from this direction. In practice, the environment map is stored as a rectangular pixel array, using the latitude-longitude projection: The environment is projected onto the image using polar coordinates (latitude and longitude). A pixel's x coordinate corresponds to its longitude θ, and the y coordinate corresponds to its latitude ϕ. The upper-right corner corresponds to the spherical coordinates $(\phi,\theta) = (0,0)$ and the lower-left to the coordinates $(\phi,\theta) = (\pi,2\pi)$. The typical resolution of the images is such that a range of 2π in θ is covered by approx. 14000 pixels and a range of π in ϕ is covered by approx. 7000 pixels. The spherical coordinate system of each camera is related to an Euclidean coordinate system as depicted in Figure 3.

2.2. Reconstruction pipeline

In order to reconstruct dense point clouds from spherical images, we first compute the position and orientation of each camera using Structure from Motion (SFM), and then compute dense points clouds using Multiple View Stereo (MVS) techniques. We derive for each of these steps specific approaches to deal with the spherical geometry. Figure 4 shows the different steps of our method.

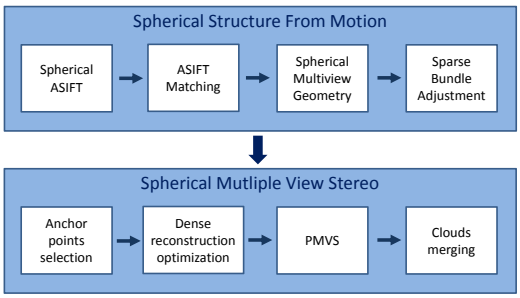

Figure 4: *Outline of our algorithm*

3. Structure from Motion with Spherical Images

The aim of Spherical Structure from Motion is to find the precise position and orientation of every spherical camera. It takes a set of images as input and computes the position and orientation of the cameras as output, as well as a sparse scene geometry. As a first step, we compute salient points on the images as well as correspondences (matches) between these points for pairs of images. These image matches are then used in an incremental scene reconstruction using a bundle adjustment procedure.

3.1. Spherical Affine SIFT with color and HDR images

Matching points between two spherical images is not an easy task. Because of the specific geometry of the cameras, the appearance of a given point on two different images changes drastically. In order to still find matches between spherical images, we opted for a variant of the Affine-SIFT approach (ASIFT) [MG09], a method that simulates all image views obtainable by varying two camera axis orientation parameters left over by the SIFT method [Low04]. The resulting method is mathematically proved to be fully affine invariant. Our method uses the same idea of views synthesis, but using a spherical image as an input.

More precisely, given a spherical image, we compute a regular distribution of n center points M_i over the sphere surface. For each center M_i, we generate a perspective image of fixed field-of-view centered on M_i by projecting the sphere's pixels on a tangential plane touching the sphere at M_i. If we now vary the angle between the normal of that plane and the lined formed by the center of the sphere and M_i, we can generate various synthetic views of the same scene corresponding to different affine transformations in the ASIFT method. For each synthetic view, we apply a salient point detector. Here we use a color variant of SIFT named PC-SIFT [CPS10] for color images and an HDR variant of

© The Eurographics Association 2011.

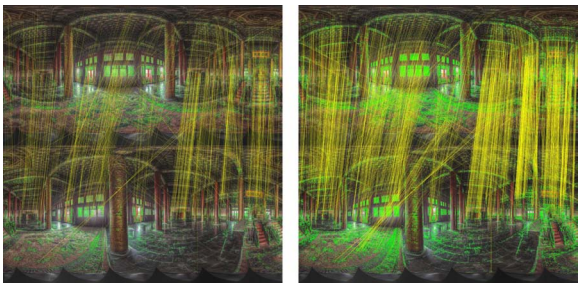

Figure 5: *Spherical ASIFT example: (left) without Affine simulations: 681 matches - (right) with Affine simulations: 2620 matches*

SIFT [CPS11] for HDR images. PC-SIFT converts colors to a perception based color space before computing SIFT features in order to make SIFT descriptors invariant to illumination changes in standard color images. HDR-SIFT bases on PC-SIFT and additionally models the light source distribution using a Gaussian Mixture Model in order to create a material color image before applying PC-SIFT. Details can be found in the cited papers. All the SIFT points found in the virtual views are then reprojected onto the sphere to find their spherical coordinates, but we keep their descriptors from the affine views. In the reprojection process, we merge points that are near to each other, but keep multiple descriptors for each merged point. Figure 5 shows results of the spherical ASIFT method. With affine simulations, we get more than 4 times more matches between two spherical images.

3.2. Structure from Motion on the sphere

Structure from Motion (SFM) is a technique that can recover all the camera poses as well as a sparse scene structure up to a scale [HZ00]. We fix this scale by setting the distance between the two first cameras to the unit distance. The main idea is to compute the epipolar geometry of the two first cameras from the matches, then triangulate all matched points to get 3D coordinates. After this initialization, for each remaining spherical camera, we successively perform following steps: (1) add one new camera in the set, (2) get 2D-3D correspondences from the matches with already triangulated points, (3) compute the pose of the new camera using the DLT method [AAK71] and (4) triangulate all possible points from matches between calibrated cameras. Parts of this procedure need to be treated carefully when dealing with spherical images. We will now explain in details the relevant parts.

3.2.1. Epipolar geometry of two spherical images

Let C_1 and C_2 be the centers of two spherical cameras and $(\mathbf{R},\mathbf{t})$ be the transformation between the cameras. A point P has the 3D coordinates P_i in the Euclidean coordinate frame of the camera C_i, so that following relation holds: $P_2 = \mathbf{R}P_1 + \mathbf{t}$. We name p_1 and p_2 the normalized versions of vectors P_1 and P_2, *i. e.* p_i has unit length and points to the same direction as P_i. The points p_i can be seen as the images of the point P on each spherical image. The remainder of the demonstration follows the same idea as in the perspective case: the vectors p_2, $\mathbf{t}$ and $\mathbf{R}p_1$ are coplanar, the normal of the plane containing the vectors $\mathbf{t}$ and $\mathbf{R}p_1$ is $\mathbf{n} = [\mathbf{t}]_\times \mathbf{R}p_1$, and p_2 lying on this plane can be expressed by $p_2^\mathrm{T}\mathbf{n} = 0$, *i. e.* $p_2^\mathrm{T}[\mathbf{t}]_\times \mathbf{R}p_1 = 0$. We therefore have the epipolar constraint for imaged points

$$p_2^\mathrm{T}\mathbf{E}p_1 = 0,\ \mathbf{E} = [\mathbf{t}]_\times \mathbf{R} \tag{1}$$

$\mathbf{E}$ can be linearly computed using eight pairs of corresponding points [LH81]. In practice, we use much more points and reject outliers using RANSAC [FB81]. Note that equation (1) is the same as in the perspective case. The difference is that in the perspective case, the vectors p_i are normalized image points, *i. e.* usually having their Z coordinate equal to 1, and not having unit length. Here we proved that this equation holds for any central projection camera (including our spherical cameras) as long as p_i are considered as 3D Euclidean vectors. Actually it is well known [HZ00] that solving equation (1) in the perspective case requires to condition the input points to reach sufficient numerical stability. We believe this is actually due to the incorrect normalization on the image plane instead of as a unit vector. Using unit vectors as in the spherical case allows for the computation of the essential matrix without preliminary conditioning.

As in the perspective case, there exist four possible factorizations of $\mathbf{E}$ in $\mathbf{R}$ and $\mathbf{t}$. For perspective images, one usually resolves this ambiguity by stating that the 3D points should lie *in front* of the cameras. Because spherical cameras have no front, we here again have to generalize this concept and state that the image points p_i (unit vectors) and the 3D points P_i must have the same direction. This allows us to find the correct factorization.

We then can parametrize this pose using 5 parameters (3 for the rotation and 2 for the translation, the length of $\mathbf{t}$ should remain 1), and optimize the pose using non linear least squares method to minimize the distance of one point to the epipolar line of the second point. Simply minimizing $p_2^\mathrm{T}\mathbf{E}p_1$ would be erroneous, because this corresponds to the sine of the geodesic distance on the sphere surface. Instead we compute the geodesic distance between the epipolar plane and the image point on the surface of the sphere $dist\_ep\_geodesic = \sin^{-1}(p_2^\mathrm{T}\mathbf{E}p_1)$, and *locally* project this distance to a tangential plane to get a distance of same dimensionality as in the perspective case. If the plane is tangent on one of the points, the local projection gives $dist\_ep\_tangent = \tan(dist\_ep\_geodesic)$, which turns out to have the simple mathematical form:

$$dist\_ep\_tangent = \frac{p_2^\mathrm{T}\mathbf{E}p_1}{\sqrt{1-(p_2^\mathrm{T}\mathbf{E}p_1)^2}} \tag{2}$$

© The Eurographics Association 2011.

Note that equation (2) can also be used in the perspective case. If the intrinsic matrix K of the perspective image is known, then p_i are K-normalized image points. This distance is more accurate for image points that are far to the camera center (especially in wide angle cameras). For spherical images, using the distance of equation (2) instead of the geodesic distance avoids making expensive trigonometric computations.

3.2.2. Pose of a sphere from 2D-3D correspondences

For the same essential reasons as for the epipolar geometry, we can derive the DLT algorithm [AAK71] solving the pose of a camera from 2D-3D correspondences for *spherical* image points seen as 3D unit vectors. If P_W is a 3D point in the world (global) coordinate system and P_C the same point in the Euclidean coordinate system of the spherical camera, our aim is to find $\mathbf{R}$ and $\mathbf{t}$ so that $P_C = \mathbf{R}P_W + \mathbf{t}$. P_C projects as a 3D unit vector p on the sphere, *i. e.* there is a unknown positive λ such that $\lambda p = P_C$, so the cross product of p and $\mathbf{R}P_W + \mathbf{t}$ is zero. This cross product gives us three equations in the unknown elements of $\mathbf{R}$ and $\mathbf{t}$, among which only two are linearly independent. Thus 6 correspondences between a 3D point and its spherical image suffice to retrieve the pose. In practice we use many points together with a robust outlier rejection.

A non linear refinement of the pose can be done using 6 parameters. Here again, the definition of the error should be carefully chosen. In the perspective case, the reprojection distance (image distance between the 2D point and the reprojected 3D point) is often used. In the spherical case, we first compute the geodesic reprojection distance. If p is the unit 3D vector representing the spherical image point and P_r the unit vector representing the reprojection of P on the sphere, then we have $dist\_pt\_geodesic = \cos^{-1}(p.P_r)$. We then *locally* project this distance to a tangential plane. Here the plane is tangent between the two points, so that $dist\_pt\_tangent = 2\tan(dist\_pt\_geodesic/2)$, and the tangential distance is

$$dist\_pt\_tangent = 2\sqrt{\frac{1-p.P_r}{1+p.P_r}} \tag{3}$$

The distance in equation (3) can also be used for perspective images with K-normalized points.

3.2.3. Sparse Bundle Adjustment

Each time a new camera is added, we apply a global bundle adjustment set over all the cameras and all the already computed 3D points. To this aim, we use a modified version of the Sparse Bundle Adjustment (SBA) package from Lourakis and Argyros [LA09]. In our case, we need to compute the reprojection error using spherical image points. We therefore use the Euclidean representation of points on the sphere as unit 3D vectors in the SBA package. Thanks to the spareness of the underlying normal equations, this step can be done very efficiently. We typically compute SBA for tens of thousands of 3D points and hundred thousands of reprojections within a few seconds.

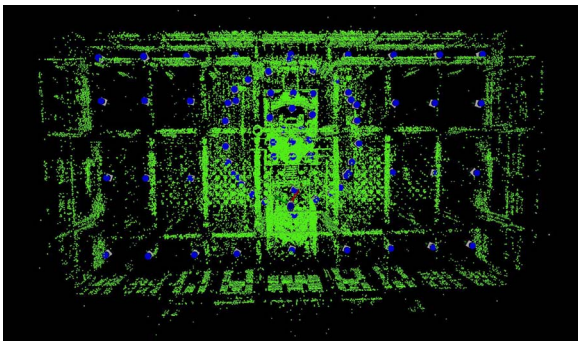

Figure 6: *Sparse point cloud with orientation and position of the 93 spherical cameras (blue dots)*

3.3. Camera poses and sparse point clouds

It is worth noting that using spherical images instead of perspective ones has several advantages in the SFM pipeline. First, due to their omnidirectional field of view, the cameras "capture" 3D points in all the directions. This allows for a natural balance of the errors in the underlying algorithms that reduces the overall error in the camera pose. This contrasts to the perspective case where in both epipolar geometry and DLT, the cameras can be calibrated only with respect to 3D points lying *in front* of the cameras. This inevitably induces a larger error variance in the viewing direction of each camera. An extensive study of this phenomenon is out of the scope of this paper, but it could be verified in all our experiments. Second, the average amount of 3D points (structure) generated for each camera in the SFM is much larger than using perspective images. Figure 6 shows a top view of the result of SFM. We used here 93 spherical cameras covering the surface of a Chinese temple in the Forbidden City. Besides the positions of the camera, the SFM step produced 193,552 3D points (green dots), spanning the complete building including the ceiling and the elevated throne.

4. Dense multi-spheres stereopsis

The Structure from Motion step only delivers a *sparse* reconstruction of the structure. Now our aim is to obtain a dense point cloud from the calibrated images. This step is equivalent to the Multiple View Stereo step of the classical pipeline. Here, we apply a similar method for spherical images.

According to the Middelbury challenge [SCD*06], one of the best methods to recover a dense point cloud from calibrated views is the Patch-based Multiple View Stereo (PMVS) method from Furukawa and Ponce [FP08]. This approach generates and propagates a semi-dense set of patches

© The Eurographics Association 2011.

and gets a very accurate reconstruction. The method is a match, expand, and filter procedure, starting from a sparse set of matched keypoints, that are successively expanded before a filtering step based on visibility constraints (outlier rejection). We therefore chose to adopt a similar approach for our scenario. In the original method however, the images are perspective and the intrinsic parameters of the cameras have to be known. Inspired by PMVS, we propose a spherical variant, that we call S-PMVS.

4.1. Point cloud generation using anchor points

Using the sparse point cloud generated in the SFM step, we can select a number of regularly distributed anchor points A_i over the coarse model. The points A_i can be selected automatically among the reconstructed 3D points on the structure using a constraint to get an uniform distribution, or they can be selected by a user. They must not be one of the 3D points from the SFM step, but should lie approximately on the surface of the expected structure. We typically use between 10 and 30 anchor points. For each anchor point A_i, and each spherical camera C_j, we can generate a perspective image I_{ij} from the spherical image by projecting the pixels from the sphere on a tangent plane perpendicular to the line through A_i and the center of C_j. I_{ij} can be seen as a virtual perspective view, where the anchor point A_i is visible in the center of the image. The field of view of I_{ij} is set using an arbitrary virtual focal length. Thus we can compute the intrinsic parameters of the image I_{ij} as a matrix $\mathbf{K}_{ij}$. In addition, because we know the pose $\mathbf{R}_j, \mathbf{t}_j$ of the spherical camera C_j and the position of the point A_i, we can compute the pose of the virtual camera I_{ij} as the composition of the pose $\mathbf{R}_j, \mathbf{t}_j$ and an internal rotation that makes the Z axis point to A_i. We can therefore deduce the pose $\mathbf{R}_{ij}, \mathbf{t}_{ij}$ of the virtual image I_{ij}, and its projection matrix $\mathbf{P}_{ij} = \mathbf{K}_{ij} \left[\mathbf{R}_{ij} | \mathbf{t}_{ij}\right]$

We now have for each anchor point A_i a set of virtual perspective images I_{ij} - one for each spherical camera center - with their projection matrices $\mathbf{P}_{ij}$. This is used as an input for the standard PMVS algorithm to generate a dense point cloud for the neighborhood of A_i. After generating the points clouds for all the anchor points, we merge all the local clouds in one single cloud.

4.2. Optimization of the dense reconstruction

The generation of point cloud based on anchor points works well if the objects in the vicinity of the anchor points have approximately a constant size across all the images. Because of the different positions of the cameras, and therefore different depths of the point A_i in all the images, this is rarely the case. The PMVS method being based on the normalized cross correlation between patches in different images, this method fails when the object size varies too much. In order to solve that problem, we apply a simple yet powerful idea when generating the virtual images: instead of using a constant focal length f_i for all images, we use a focal length f_{ij} that is proportional to the depth of point A_i in the image I_{ij}. The effect of this technique is to adapt the zoom of each virtual camera to make the apparent size of the object of interest constant. With this optimization, the expansion step of the PMVS method is much more efficient and large parts of the objects can be reconstructed.

5. Results

5.1. Building interior

In the *Tai He Dian* scenario, we acquired 93 spherical images of the interior of the *Hall of Supreme Harmony*, situated in the middle of the Forbidden City in Beijing. The acquisition time for 93 images is approximately 120 minutes. We find approximately 50,000 affine SIFT points on each image. After pairwise matching, we find up to 7,000 matches for each camera pair. Figure 6 shows the distribution of the 193,552 3D points found after the SFM step, as well as the position of the cameras on the floor. We then computed partial point clouds with 38 anchor points. The resulting merged point cloud contains over 100 millions points. In Figure 7, different views of the reconstructed point cloud including an external one are shown. Note that the output of our algorithm is a dense point cloud, not a surface. However, in many regions, the density of the points is such that the point cloud appears as a continuous surface.

5.2. Outdoor reconstruction

In a second scenario (*bridge*), we acquired 98 spherical images of an ancient sculpted bridge from the gardens of the Forbidden City in Beijing, and used 49 anchor points. The resulting point cloud with over 127 millions points is shown in Figure 8. In this particular example, the stones of the bridge and its walls have been completely reconstructed as an apparently continuous surface. Note the level of detail that could be recovered in the reconstruction of fine sculptures on the walls (statues of lions and sculptures of dragons).

6. Conclusion

In this paper, we have presented a method to generate dense 3D point clouds from high-resolution spherical images. The main advantage of our method is that it requires much less images than using a standard perspective camera. In order to cope with the spherical geometry of the cameras, we presented several new algorithms required in the reconstruction pipeline such as: spherical affine SIFT, spherical epipolar geometry and pose computation for spherical cameras using a new distance based on the geodesic distance on the sphere, and more suited to the spherical case. We have shown in two applications that we can generate accurate point clouds with over hundred millions of 3D points from around hundred of camera positions. In our future work, we would like to go

© The Eurographics Association 2011.

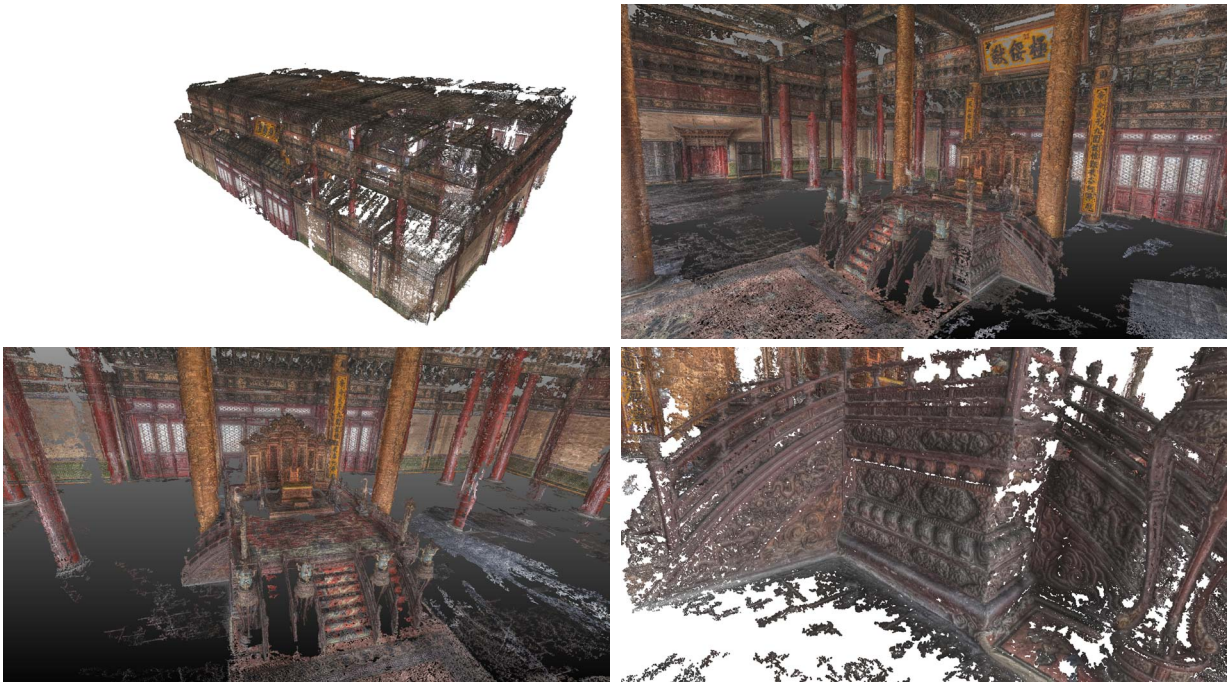

Figure 7: *3D reconstruction of the* Tai He Dian *temple. Top-left: external view. Top-right and second row: examples of inside views*

one step further in the reconstruction pipeline and provide textured meshes based on our current point clouds.

Acknowledgment. This work has been partially funded by the project CAPTURE (01IW09001). The authors would like to thank the **Chinese National Palace Museum** for their collaboration throughout this project.

References

[AAK71] ABDEL-AZIZ Y. I., KARARA H. M.: Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In *Proc. Symposium on Close-Range Photogrammetry* (1971), pp. 1–18. 4, 5

[ASS*09] AGARWAL S., SNAVELY N., SIMON I., SEITZ S. M., SZELISKI R.: Building rome in one day. In *ICCV* (2009). 2

[BFRS10] BARAZZETTI L., FANGI G., REMONDINO F., SCAIONI M.: Automation in multi-image spherical photogrammetry for 3d architectural reconstructions. In *International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST)* (2010). 2

[CPS10] CUI Y., PAGANI A., STRICKER D.: Sift in perception-based color space. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)* (2010). 3

[CPS11] CUI Y., PAGANI A., STRICKER D.: Robust point matching in high dynamic range images through estimation of illumination distribution. In *Annual Symposium of the German Association for Pattern Recognition (DAGM)* (2011). 4

[FB81] FISHLER M. A., BOLLES R. C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. Assoc. Comp. Mach. 24*, 6 (1981), 381–395. 4

[FCS*10] FURUKAWA Y., CURLESS B., SEITZ S. M., , SZELISKI R.: Towards internet-scale multi-view stereo. In *CVPR* (2010). 2

[FGG*10] FRAHM J.-M., GEORGEL P., GALLUP D., JOHNSON T., RAGURAM R., WU C., JEN Y.-H., DUNN E., CLIPP B., LAZEBNIK S., POLLEFEYS M.: Building rome on a cloudless day. In *ECCV* (2010). 2

[FP08] FURUKAWA Y., PONCE J.: Accurate, dense, and robust multi-view stereopsis. *PAMI 1* (2008), 1–14. 2, 5

[GSC*07] GOESELE M., SNAVELY N., CURLESS B., HOPPE H., SEITZ S. M.: Multi-view stereo for community photo collections. In *ICCV* (2007). 2

[HZ00] HARTLEY R., ZISSERMAN A.: *Multiple View Geometry*. Cambridge University Press, 2000. 4

[KBK07] KOESER K., BARTCZAK B., KOCH R.: An analysis-by-synthesis camera tracking approach based on free-form surfaces. In *Annual Symposium of the German Association for Pattern Recognition (DAGM)* (2007). 2

[KH10] KIM H., HILTON A.: 3d modelling of static environments using multiple spherical stereo. In *ECCV Workshop on Reconstruction and Modeling of Large-Scale 3D Virtual Environments (RMLE)* (2010). 2

[LA09] LOURAKIS M. A., ARGYROS A.: SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Trans. Math. Software 36*, 1 (2009), 1–30. 5

[LF05] LI S., FUKUMORI K.: Spherical stereo for the construction of immersive vr environment. In *VR* (2005). 2

© The Eurographics Association 2011.

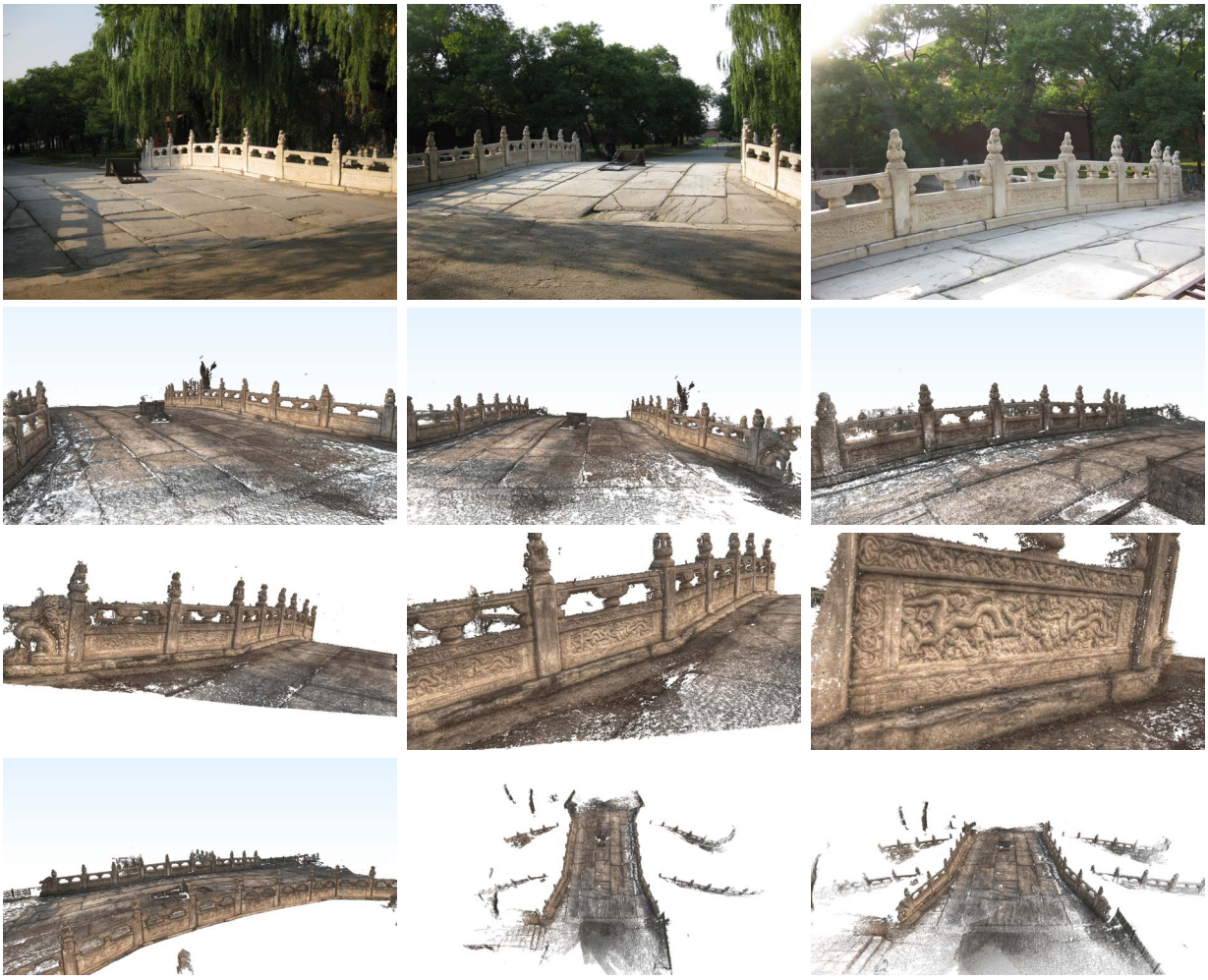

Figure 8: *3D reconstruction of the bridge. First row: real views. Second row: same views from the dense 3D point cloud. Third row: additional close-up views of the sculpted walls. Fourth row: additional external views.*

[LH81] LONGUET-HIGGINS H.: A computer algorithm for reconstructing a scene from two projections. *Nature 293* (1981), 133–135. 4

[Low04] LOWE D. G.: Distinctive image features from scale-invariant keypoints. *Int. J. of Comp. Vision 60* (2004), 91–110. 3

[MFB06] MAUTHNER T., FRAUNDORFER F., BISCHOF H.: Region matching for omnidirectional images using virtual camera planes. In *Computer Vision Winter Workshop* (2006). 2

[MG09] MOREL J., G.YU: Asift: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences 2* (2009), 438–469. 2, 3

[MP06] MICUSIK B., PAJDLA T.: Structure from motion with wide circular field of view cameras. *PAMI 28* (2006), 1135 – 1149. 2

[SCD*06] SEITZ S. M., CURLESS B., DIEBEL J., SCHARSTEIN D., SZELISKI R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR* (2006). 2, 5

[SK05] STRECKEL B., KOCH R.: Lens model selection for visual tracking. In *Annual Symposium of the German Association for Pattern Recognition (DAGM)* (2005). 2

[SSS06] SNAVELY N., SEITZ S. M., SZELISKI R.: Photo tourism: Exploring photo collections in 3d. In *SIGGRAPH Conference Proceedings* (New York, NY, USA, 2006), ACM Press, pp. 835–846. 2

[SSS08] SNAVELY N., SEITZ S. M., SZELISKI R.: Skeletal sets for efficient structure from motion. In *Proc. Computer Vision and Pattern Recognition* (2008). 2

[SY05] SATO T., YOKOYA N.: Omni-directional multi-baseline stereo without similarity measures. In *OMNIVIS* (2005). 2

[VVG06] VERGAUWEN M., VAN GOOL L.: Web-based 3d reconstruction service. *Mach. Vision Appl. 17* (October 2006), 411–426. 2

[WH06] WILLIAMS P., HILTON A.: 3d reconstruction using spherical images. In *Proceedings of the 3rd European Conference on Visual Media Production* (November 2006), p. 179. 2

© The Eurographics Association 2011.