Low Cost Setup for High Resolution Multiview Panorama Recording and Registration

1st Matthias Ueberheide
Computer graphics lab
TU Braunschweig
Braunschweig, Germany
ueberheide@cg.cs.tu-bs.de

2nd Moritz Muehlhausen
Computer graphics lab
TU Braunschweig
Braunschweig, Germany
muehlhausen@cg.cs.tu-bs.de

3rd Marcus Magnor
Computer graphics lab
TU Braunschweig
Braunschweig, Germany
magnor@cg.cs.tu-bs.de

***Abstract*—Affordable consumer panorama solutions are still limited to low resolution and synchronization for multiple cameras has to be done externally. Professional equipment that provides high resolution recording and synchronization is not widely available yet. In this paper we present a low cost setup to record multiple high resolution panorama videos at the same time. We perform an external synchronization. To allow further computer vision algorithms to process these recordings the videos are adjusted and aligned automatically by extracting exact camera positions.**

***Index Terms*—panorama capture, alignment, calibration**

I. INTRODUCTION

For years panorama stills are available through consumer hard- and software, integrated in most smartphones and digital cameras. Over the last years, a multitude of consumer panorama cameras emerged allowing everyone to record omnidirectional video. Those consumer cameras are, while easy to handle, still limited in resolution. Professional solutions are not yet widely available or affordable.

However, the increasing amount of panorama video content shared on social platforms shows the interest in recording and sharing immersive experiences. Not only do panoramas increase the immersion and allow us to revisit places we visited or to relive events we experienced, but also inherently more complete information about the cameras environment can be gathered. The additional information can be used to synthesize stereo effect increasing the immersion even more. Still, the collected information is gathered from a single point in space which results in increasingly large holes in the scene once motion parallax is allowed.

In this paper we present a capture modality that utilizes multiple panorama rigs to collect missing data from the scene. Using the data from multiple rigs requires synchronized high resolution recordings and registered rig positions. We present our own rig-design based on multiple GoPro Cameras that is capable of recording high resolution video. The cameras are synchronized using a wifi signal avoiding the necessity of wires between rigs. A robust solution for camera registration is provided.

The authors gratefully acknowledge funding by the German Science Foundation (DFG MA2555/15-1 “Immersive Digital Reality”).

II. RELATED WORK

A. Camera construction and synchronization

Panorama video recording has come from complex mirror construction with various recent expansion [1] to direct camera array methods which reduce size but have to handle parallax artifacts in overlapping regions resulting from parallax between cameras. Over the last years, multiple commercial panorama cameras became available from systems intended for professional use like Jaunt One, Facebook Surround-360 [2] and Nokia Ozo. These systems are expensive and grant no or only limited access to their internal processing. Consumer grade panorama cameras are available (eg. Samsung Gear 360, Ricoh Theta S and GoPro Fusion 360) and allow cost efficient and easy panorama recording. However, the resolution is limited and synchronization between multiple devices is not supported. Matzen et. al. [3] use two low-cost panorama cameras to synthesize a stereo panorama and perform manual frame synchronization.

In contrast to closed systems, camera array based systems, for example the *Freedom360* and *360Heros*, grant full access to the raw images but also require synchronization within a single rig to avoid artifacts. Anderson et. al. [4] eliminate temporal artifacts by synchronizing the cameras with a wire-based solution. However, the necessity to add wires between the rigs would restrict the usability of the proposed capture modality. Parallax and temporal artifacts aware algorithms (eg. Lee et. al. [5]) can be used to compensate for small temporal offset. Thus frame-accurate synchronization is sufficient to produce high quality panorama video. Dependent on the distance between the panorama rigs manual synchronization might become impossible or at least tedious as small temporal missalignments are less visible between rigs.

Finally, content based synchronization can be performed based on feature trajectories [6], [7]. Similar techniques are available in commercially available panorama video stitchers, for example the AutoPano Video software [8]. However, for large temporal offsets, the results become unreliable.

B. Calibration

The topic of image alignment and full camera registration has been researched extensively. The broad research on this and related topics can be divided into three main categories.

ISBN 978-90-827970-1-5 © EURASIP 2018

Image Statistics. These methods exploit different image statistics for alignment. For parabolic catadioptric images Makadia et. al. [9] propose a method involving spherical Fourier transform. They convert the image into the frequency domain and obtain the rotation from the conservation of harmonic coefficients in the rotational shift theorem. However, the frequency domain is sensitive to translation and non-static objects, making the correlation between moved cameras or frames of dynamic scenes difficult. Gurrieri et. al. [10] align cylindrical panoramas based on dense disparity maps. They first perform a rough alignment of the panoramas by finding the most similar patch to a central region. Afterwards they identify and classify patterns in the disparity maps within this region for further alignment. Since computing disparity maps is error-prone for misaligned images, this method relies strongly on the exhaustive search in the first step.

Vanishing Points. The methods of this category utilize vanishing points for various tasks. Vanishing points are intersections of parallel lines in the world projected in an image. They correlate to points at infinity and therefore their position depends only on the rotation and is independent of translations. This idea has been used for tasks like 3D reconstruction [11] and camera calibration [12]. Moreover, they have been used for camera orientation estimation for conventional images [13] and panorama images [14], [15]. While some approaches work on edge pixels directly [13], [14] to estimate vanishing points, others use extracted lines [15]. All of these methods work best when satisfying the *Manhattan World Assumption* [16], which states that scenes are built in a cartesian grid, so that only a single dominant orientation is assumed. Another approach utilizes vanishing points for *Upright Adjustment* [17] of spherical panoramas to deal with possible camera tilts and resulting wavy horizons and slanted objects. Our featureless registration method builds upon this work to align our panoramas.

Feature Matching. Full camera registration methods usually utilize point correspondences between images for camera registration, which has been shown to be possible up to scale and a four-fold ambiguity given eight point correspondences [18]. This has been used for tasks like 3D reconstruction for conventional cameras [19], [20] and simultaneous localization and mapping (SLAM) algorithms for conventional [21] as well as panorama cameras [22]. Cubic panorama pairs can be aligned to their respective epipole [23], [24] using SIFT feature points on each side of the panorama. However, these approaches align only a pair of images towards their epipole, making them unsuited for more than two images. This limitation was addressed by Salehi et. al. [25] to align complete cubic panorama image data sets to a common direction. Although feature matching methods can produce impressive results, they suffer from various limitations. The matching step can be difficult due to illumination changes, repeating textures, mirrors or other kind of ambiguities within the images. Therefore, special markers have been introduced for robust pose estimation [26], [27]. This is one of the two options we use for camera calibration by adding several markers to our rigs.

III. Panorama Recording and Registration

A. Camera Rig

Recording panorama video requires specialized recording hardware. While some camera solutions are available to consumers, the technical capabilities of those systems are still restricted. Professional systems offer much better quality but multiple rigs are often not available. We construct our own rig based on multiple cameras to be able to create high resolution panorama video with multiple rigs. Six GoPro Hero 4 cameras are placed in a circular pattern in $60°$ steps. Using the widest field of view the cameras can provide, neighbouring camera views overlap almost up to their optical axis. For the overlapping areas depth can be extracted through stereo based depth extraction (e.g. [2]) using only a single rig. At the same time we minimize the parallax between the cameras to reduce stitching artifacts. The circular arrangement has a radius of $5cm$ resulting in $5cm$ parallax between neighbouring cameras. Three additional cameras on the top allow us to capture to full upper hemisphere. The cameras are placed such that their optical axis intersects with the rotational center of the rig. The final stiched panorama has a resolution of 11500×4090.

Next to each camera in the circular arrangement a marker is positioned to allow for automatic calibration of multiple rigs. The markers are placed such that they do not obstruct the view of the cameras and are still as close as possible to the cameras' optical center.

Through extensive use of the wifi features of the cameras battery operation time is reduced significantly. To avoid disassembly after each recording we designed the rig to keep the battery slot accessible in the assembled rig. Without the necessity for an external power supply we keep the rig compact and mobile. Inspection and download of the recorded footage

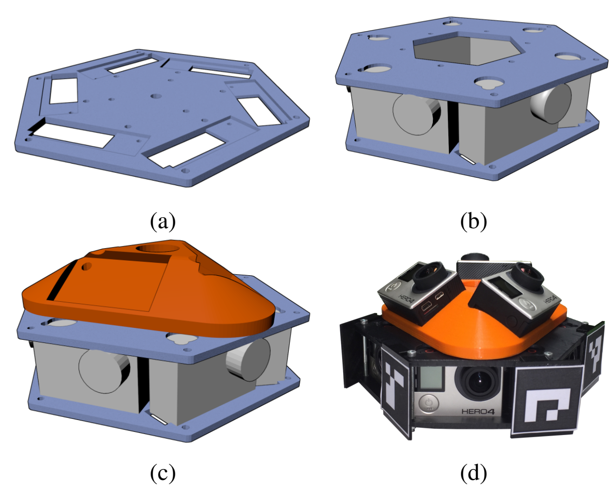

Fig. 1: a) Baseplate of our rig. Battery is accessible in assembled rig. b) Assembled circular rig consisting of 6 cameras c) Additional 3 cameras record the residual part of the top hemisphere d) Complete rig including markers.

ISBN 978-90-827970-1-5 © EURASIP 2018

is also done through the wifi interface so no access to the memory card is necessary. We place multiple of those rigs in the scene which requires additional effort to synchronize recording.

B. Camera Synchronization

Temporal offset between cameras in a multi-camera rig introduces artifacts in stitching areas for moving scene elements. Integrated panorama cameras take care of the synchronization internally, but provide no access to the synchronization to integrate multiple rigs. A single camera rig consists of individual cameras that require synchronization between each other. While large overlap can compensate for small temporal inaccuracies at least a rough synchronization is still required. Additionally, we use multiple rigs that also require synchronization to allow further processing of the recordings. Instead of separating solutions for within-rig and between-rig synchronization we opt for a wireless triggering method.

Our wireless trigger uses the wifi feature of the GoPro Hero 4 in remote control (RC) mode. In this mode the cameras join a network which is hosted by an access point with gopro MAC address. We set up a wireless network which is recognized by the cameras as a remote control.

The cameras expose their functionality via HTTP, which we use to distribute the configuration for all cameras. Additionally, we can make sure that sufficient memory and battery remain. Besides the HTTP-based accessibility the cameras do also listen to a very lightweight UDP protocol. In our experiments we found that the HTTP request for the start of the recording is processed with high delays (sometimes exceeding seconds). Therefore we use a UDP Signal which is sent to all cameras triggering the recording. We verify that all cameras started recording by requesting their status in cases of lost UDP packages. To detect rare cases of severe delay in the start of the recording in single cameras the cameras internal time is set from the controlling computer. These long delays are usually limited to the first recording after camera startup.

C. Calibration

With synchronized recordings of the individual cameras for each rig, the next step is to stitch the images of each individual rig to obtain a spherical panorama image. For this purpose we use the AutoPano Video software [8]. With the known fixed camera positions within our rig the stitching quality can be increased. However, for further processing of the obtained panoramas some problems arise. Without precise leveling of the camera rigs, the obtained spherical panorama images include wavy horizons and slanted objects. A second problem is that the spherical panoramas of our rigs may not face a common direction due to unknown orientations. At last, it is desirable to know the positions of the recording rigs. To deal with these problems we offer two solutions:

- ***Marker-less Alignment*** and *Upright Alignment* [17] of the panoramas
- ***Full registration*** with *Aruco Marker* [26], [27]

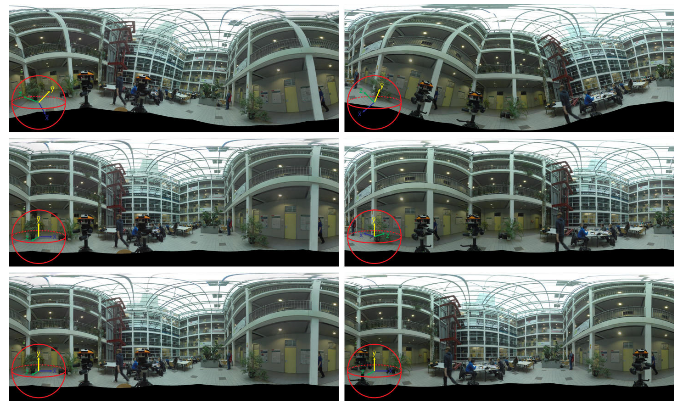

Fig. 2: The effect of the two steps for our *Upright Alignment*. The left column shows the reference panorama. Top: Input panorama pair with arbitrary orientations. Middle: After *Upright Adjustment* $\vec{y}$ of their coordinate systems are aligned. Bottom: *Rotational Alignment* aligns $\vec{x}$ and $\vec{z}$ to the reference panorama.

Marker-less Alignment has the benefit of its flexibility as no further adjustments to either the scene or the rigs have to be made. For our marker-less registration we assume static non-moving rigs, however, we do not expect any explicit distance in between. Therefore, this registration method is only correct up to scale. First we apply *Upright Adjustment* following Jung et. al. [17] to all panoramas to address the first problem of possible tilted camera rigs. A spherical hough transformation is used to identify straight lines within the scene in a panorama as well as detect vanishing points at intersections of several lines. The panorama is rotated to level the horizontal vanishing points in the image and to match the vertical vanishing point with the north pole of the image. The resulting upright adjusted panoramas may still not face a common direction.

For the *Rotational Alignment* we assume to have two upright adjusted images, i.e. a panorama I and a reference panorama $\widehat{I}$ to align to. However, our method can handle any number of panoramas, by subsequently aligning all panoramas to the same reference $\widehat{I}$. We also keep the horizontal vanishing point $h \in H$ of the panorama I and $\widehat{h} \in \widehat{H}$ of the reference panorama $\widehat{I}$ of the *Upright Adjustment*.

We need to find one pair of corresponding vanishing points and assign a new horizontal position to every pixel (n, m) of the image I. However, points at infinity, namely vanishing points, can be visually blocked by closer objects within one image. Hence, the surroundings of corresponding vanishing points can be very different, making matching difficult. Therefore, instead of matching the vanishing points directly, we compare the rotated images given a corresponding vanishing point pair. Assuming a vanishing point $\widehat{h}_i \in \widehat{H}$ of the reference image $\widehat{I}$ to correspond to $h_j \in H$ in the current image I, we rotate I by $\widehat{h}_i - h_j$ around the north pole to obtain $I'_{\widehat{h}_i,h_j}$.

$$I'_{\widehat{h}_i,h_j}(n, m) = I(n, m - (\widehat{h}_i - h_j))$$

ISBN 978-90-827970-1-5 © EURASIP 2018

Then, we compare the obtained image with $\widehat{I}$ to check whether the points actual correspond. Testing with all vanishing point pairs $\widehat{h_i}, h_j$, we obtain the best match. This is similar to the coarse alignment of Gurrieri et. al. [10]. Yet, we only need to check a few rotations, instead of a complete search over the width of the image. Moreover, unless we found only distinct sets of vanishing points, at least one of these rotations is correct. We use the *Structural Similarity (SSIM)* for comparison instead of plain pixel values:

$$\max_{\widehat{h_i}\in\widehat{H},h_j\in H} \left|\left|\mathrm{SSIM}(\widehat{I}, I'_{\widehat{h_i},h_j})\right|\right|_2^2 \tag{1}$$

Here, $\widehat{H}$ is the set of the vanishing points in the reference panorama $\widehat{I}$ and accordingly H is the set of vanishing points in I. $||\cdot||_2^2$ is the squared norm over the channelwise computed *SSIM*. The best matching rotated panorama is assumed to be aligned. Figure 2 highlights the effects of these two steps.

Within these upright adjusted and aligned panoramas we determine the centers of all used rigs once manually to identify their respective direction. This also allows us to triangulate the relative positions of our rigs up to scale.

Full Registration implies full knowledge of the positions and orientations of all rigs. For this we extend our rigs with *Aruco Markers* at each side of every rig without limiting the cameras view. These markers have a definite size to allow precise distance identification. In our case the markers have a side length of 4.1 cm, which allows robust detection with up to 2 m distance. Furthermore, all markers' ids are distinct and fixed in a predetermined order, resulting in a determinable orientation of our rigs.

We detect the markers in all obtained non-stitched images from our cameras instead of in the stitched panoramas. This allows us to regularize the detected markers within one rig as captured marker should be visible in at least two images, increasing the precision of marker detection. The center position as well as the orientation can be computed with two detected markers. With two positions and orientations of markers we know two adjacent sides of our hexagonal rig, which is sufficient for definite identification of the center. The predetermined order of the markers' ids give information about the orientation. At last the computed positions and orientations can be regularized among the used rigs. While the marker size limits the maximum distance between our rigs for full registration, this approach is applicable even for moving rigs.

IV. Results

A. Synchronization

In order to verify our synchronization we project a laser line onto a wall with a marker. All cameras are triggered filming the wall and the framenumber of the line crossing the marker is detected manually.

Using the UDP network protocol to trigger the cameras, we found that the first two attempts to record introduce considerable delays with some cameras requiring two to five seconds to become ready to record. We assume that internal camera resource initialization amounts to prolonged starting times. Performing two short recordings before the actual scene capture we achieve temporal offsets below 7 frames. This synchronization is sufficient for panorama stitching and correction with the large overlap or allows for a fast manual improvement by just a few frames.

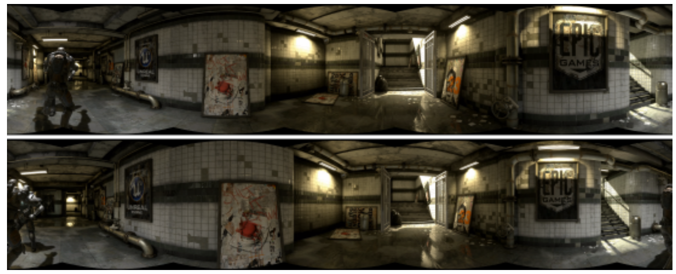

Fig. 3: Perfectly aligned panoramas of the synthetic *Sequencer* scene, used for accuracy evaluation.

Sequential HTTP based triggering requires considerable time and produces offsets well exceeding multiple seconds. Sending parallel HTTP requests to the cameras tends to flood the wifi channel resulting in connection loss to the cameras. Simply using time synchronization we get temporal offsets of roughly two seconds which is mostly due to delay in updating the time and the timer only being precise to one second.

While the UDP trigger performs best, too many packages are dropped in areas with strong wifi interference making UDP to unreliable. In those cases we synchronize the cameras timers beforehand and fall back to an audio cue to provide a marker for later manual synchronization. Still, our system can be used to monitor the recording process and set up the required configuration in this fallback scenario.

B. Calibration

We analyzed our ***Marker-less Alignment*** on synthetic data, to minimize the influence of possible stitching artifacts. For this we used the *Urban* scene of the Unreal Engine 4 [28], placing two groups of virtual cameras into the scene. With perfectly known camera parameters we stitch the images with *Autopano Video* [8], resulting in panoramas with a resolution of 2750×1375, shown in Figure 3.

For accuracy evaluation and stability test we shift one panorama horizontally, which correlates with a *pan* rotation. Then, the *Rotational Alignment* step is performed and the first shift is compared with the suggested rotation of our method. We test our technique with 500 randomly chosen panorama shifts, obtaining no more than 4 pixels of misalignment. This corresponds to a maximum misalignment of $0.52°$. Adding zero-mean gaussian white noise with a variance of 0.01 for further stability tests increased the maximum pixel misalignment to 10 pixels.

For our ***Full Registration*** using *Aruco Marker* we set up three of our rigs in a triangular arrangement with an approximate distance of $1m$ in between. Due to inaccuracies

ISBN 978-90-827970-1-5

in the distortion parameter set determined for all GoPro Hero Black 4, the marker detection becomes unstable especially if the marker are only visible at the edge of an image. Therefore, the accuracy of our registration tends to be more stable if the markers are located in image centers. We can make sure that other rigs are seen close to a cameras center during camera setup circumventing this accuracy degradation. Alternatively, the distortion parameters can be determined per camera to improve the marker detection and localization using readily available camera calibration methods. With markers close to the centers our registration method shows an average error of 2.011%, whereas in the outer regions the average error reaches up to 7.797% offset.

V. Conclusion

We presented a framework for simultaneously capture and registration of multiple high resolution panorama videos using of the shelf hardware. Our synchronization method reduces the temporal offset between the recordings significantly which allows sitching with no or minor manual correction. The wireless approach is not limited to panorama recording but can also be applied to general unstructured arrays without any changes.

With the two presented methods for calibration we can adapt to different applications. While we refer to the *Marker-less Alignment* for the general case if no full registration is needed, this method can only be applied with several parallel lines within the images, restricting it to urban scenes. In the case that the rigs positions are necessary, the *Full Registration* with *Aruco Marker* can be applied, however, distortion parameters should be identified precisely for each camera to allow for a good accuracy.

For future work we plan to combine our synchronization approach with a cable based method within the rigs so only one camera per rig is triggered with a wifi signal. Additionally, the integration of sensors (e.g. accelerometers or compass-sensors) could provide a more general approach for the panorama alignment for natural scenes where no sufficient parallel lines are present.

References

[1] C. Weissig, O. Schreer, P. Eisert, and P. Kauff, “The ultimate immersive experience: panoramic 3d video acquisition,” in *International Conference on Multimedia Modeling*, pp. 671–681, Springer, 2012.

[2] D. Lindell and J. Thatte, “Virtual reality motion parallax with the facebook surround-360,”

[3] K. Matzen, M. F. Cohen, B. Evans, J. Kopf, and R. Szeliski, “Low-cost 360 stereo photography and video capture,” *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 148, 2017.

[4] R. Anderson, D. Gallup, J. T. Barron, J. Kontkanen, N. Snavely, C. Hernández, S. Agarwal, and S. M. Seitz, “Jump: virtual reality video,” *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 198, 2016.

[5] J. Lee, B. Kim, K. Kim, Y. Kim, and J. Noh, “Rich360: optimized spherical representation from structured panoramic camera arrays,” *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 63, 2016.

[6] B. Meyer, T. Stich, M. Magnor, and M. Pollefeys, “Subframe temporal alignment of non-stationary cameras,” in *Proc. British Machine Vision Conference (BMVC)*, Sep 2008.

[7] J. Kopf, “360 video stabilization,” *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 195, 2016.

[8] Kolor, “Autopano Video,” 2004-2017. http://www.kolor.com/autopano-video/.

[9] A. Makadia and K. Daniilidis, “Rotation recovery from spherical images without correspondences,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1170–1175, 2006.

[10] L. E. Gurrieri and E. Dubois, “Optimum alignment of panoramic images for stereoscopic navigation in image-based telepresence systems,” in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pp. 351–358, IEEE, 2011.

[11] R. Cipolla, T. Drummond, and D. P. Robertson, “Camera calibration from vanishing points in image of architectural scenes.,” in *BMVC*, vol. 99, pp. 382–391, 1999.

[12] R. Orghidan, J. Salvi, M. Gordan, and B. Orza, “Camera calibration using two or three vanishing points,” in *Computer Science and Information Systems (FedCSIS), 2012 Federated Conference on*, pp. 123–130, IEEE, 2012.

[13] A. T. Martins, P. M. Aguiar, and M. A. Figueiredo, “Orientation in manhattan: Equiprojective classes and sequential estimation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 5, pp. 822–827, 2005.

[14] M. E. Antone and S. Teller, “Automatic recovery of relative camera rotations for urban scenes,” in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2, pp. 282–289, IEEE, 2000.

[15] J.-C. Bazin, C. Demonceaux, P. Vasseur, and I. Kweon, “Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment,” *The International Journal of Robotics Research*, vol. 31, no. 1, pp. 63–81, 2012.

[16] J. M. Coughlan and A. L. Yuille, “Manhattan world: Compass direction from a single image by bayesian inference,” in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 2, pp. 941–947, IEEE, 1999.

[17] J. Jung, J.-Y. Lee, B. Kim, and S. Lee, “Upright adjustment of 360 spherical panoramas,” in *Virtual Reality (VR), 2017 IEEE*, pp. 251–252, IEEE, 2017.

[18] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[19] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, “Building rome in a day,” *Communications of the ACM*, vol. 54, no. 10, pp. 105–112, 2011.

[20] A. Akbarzadeh, J.-M. Frahm, P. Mordohai, B. Clipp, C. Engels, D. Gallup, P. Merrell, M. Phelps, S. Sinha, B. Talton, *et al.*, “Towards urban 3d reconstruction from video,” in *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pp. 1–8, IEEE, 2006.

[21] N. Karlsson, E. Di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. E. Munich, “The vslam algorithm for robust localization and mapping,” in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 24–29, IEEE, 2005.

[22] T. Lemaire and S. Lacroix, “Slam with panoramic vision,” *Journal of Field Robotics*, vol. 24, no. 1-2, pp. 91–111, 2007.

[23] M. Fiala and G. Roth, “Automatic alignment and graph map building of panoramas,” in *Haptic Audio Visual Environments and their Applications, 2005. IEEE International Workshop on*, pp. 6–pp, IEEE, 2005.

[24] F. Kangni and R. Laganiere, “Epipolar geometry for the rectification of cubic panoramas,” in *Computer and Robot Vision, 2006. The 3rd Canadian Conference on*, pp. 70–70, IEEE, 2006.

[25] S. Salehi and E. Dubois, “Alignment of cubic-panorama image datasets using epipolar geometry,” in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 1545–1548, IEEE, 2011.

[26] S. Garrido-Jurado, R. M. noz Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez, “Automatic generation and detection of highly reliable fiducial markers under occlusion,” *Pattern Recognition*, vol. 47, no. 6, pp. 2280 – 2292, 2014.

[27] S. Garrido-Jurado, R. M. noz Salinas, F. Madrid-Cuevas, and R. Medina-Carnicer, “Generation of fiducial marker dictionaries using mixed integer linear programming,” *Pattern Recognition*, vol. 51, pp. 481 – 491, 2016.

[28] Epic Games Inc., “Unreal Engine 4,” 2004-2017. https://www.unrealengine.com/.

ISBN 978-90-827970-1-5 © EURASIP 2018