Mixed-Reality Spatial Configuration with a ZED Mini Stereoscopic Camera

Räumliche Mixed Reality Konfiguration mit einer ZED Mini Stereoskopischen Kamera

Dimo Chotrov, Zlatka Uzunova, Yordan Yordanov, Stoyan Maleshkov

Virtual and Augmented Reality Laboratory, Research and Development and Innovation Consortium
Sofia, Bulgaria, vr-lab@sofiatech.bg

***Abstract* —** This paper starts with a short overview of how a depth map can be calculated from images captured with stereoscopic cameras, and in particular - the work principal of the ZED Mini stereoscopic camera. After that we present three approaches we have identified that can be used to create a spatial configuration based on a real-world environment. The result of this spatial configuration is a mixed environment containing virtual objects aligned with the physical objects from the real environment. At the end we provide the results from experiments we have conducted using developed prototypes to validate the proposed approaches.

***Zusammenfassung* —** Am Anfang wird einen kurzen Überblick gegeben, wie eine Tiefenkarte aus mit stereoskopischen Kameras aufgenommenen Bildern berechnet werden kann, und insbesondere das Arbeitsprinzip der stereoskopischen Kamera ZED Mini. Anschließend stellen wir drei Ansätze vor, mit denen eine räumliche Konfiguration basierend auf einer reellen Umgebung erstellt werden kann. Das Ergebnis dieser räumlichen Konfiguration ist eine gemischte Umgebung mit virtuellen Objekten, die mit den physischen Objekten aus der reellen Umgebung ausgerichtet sind. Am Ende stellen wir die Ergebnisse von Experimenten vor, die wir mit entwickelten Prototypen durchgeführt haben, um die vorgeschlagenen Ansätze zu validieren.

I. Introduction

Creating a virtual configuration of some space to see how it will look like before placing the actual physical objects in it allows trying out different combinations of objects and layouts. Through the application of mixed reality this can be done in the actual space for which the virtual configuration is intended combining the real environment with the virtual models to be placed in it. Using a stereoscopic camera capable of generating a depth map and calculating measurements of the real environment, a mixed reality application can preserve the depth information from the real environment (when presenting it to the user) and seamlessly combine it with the virtual objects.

II. Depth Calculation Using Stereoscopic Camera

A. General Principle for Estimating Depth from Stereoscopic Images

The distance from a selected point in the 3-dimentional scene to the image plane is estimated using two images of the scene, taken from different points of view at the same time. The relations between the 3D points of objects in the scene and their projections onto the 2D image planes are described by the rules of the epipolar geometry which define constraints between corresponding points in the image planes. The general principle is illustrated on Fig. 1 [1], [2].

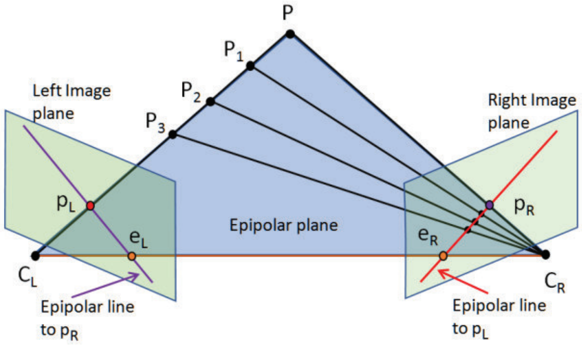

Fig. 1. Epipolar geometry of stereo vision.

The scene is captured simultaneously from two cameras (Left and Right), which are located at different positions and their optical centers are coded C_L and C_R respectively. The line between C_L and C_R forms the baseline while the three points: P, C_L and C_R build the epipolar plane. Each optical center projects into a point in the image plane of the other camera: the optical center C_L projects into e_R, while C_R projects into e_L. These two image points are called epipoles. Let's consider the 3D point P located in the scene at unknown distance from the baseline. This point is captured by both cameras and projected as 2D points: on the Left image plane as p_L and on the Right image plane as p_R. If we move the 3D point P from its initial position towards the optical center of the left camera at indicative positions P1, P2 and P3, the corresponding projections of these points into the image plane of the other (Right) camera build a line passing through e_R called epipolar line which corresponds to the projected point p_L in the other image plane. The same is valid for the projected point p_R in the Right image plane - it is related to the epipolar line created in the other (Left) image plane. The epipolar lines are formed at the intersection of the epipolar plane with both image planes and pass through the corresponding epipoles. If the 3D point P is moving in the scene the epipolar plane rotates around the baseline. An important feature known as epipolar constraint is the condition that if the relative position of the two cameras and the projection p_L of the 3D point P in the Left image plane are known then the projection p_R of the same 3D point P in the Right image plane must lie on the epipolar line (e_R, p_R). This constraint is valid for all points (e.g. P_1, P_2, P_3) which lie on the 3D line P, C_L. By analogy we can observe the same constraint for points which lie on the line P, C_R, their projection in the Left image plane must lie on the epipolar line (e_L, p_L). Another interpretation of the epipolar constraint is that it defines a mapping between points in the left image plane and lines in the right image plane and vice versa. This projective mapping from points to lines can be represented by the fundamental matrix F, which gives algebraic description of epipolar geometry.

The epipolar constraint allows us to calculate the position of the 3D point P from known positions of the projections p_L and p_R using the condition that if these projections correspond to the same 3D point, then the projection lines must intersect at P. The described principle called triangulation forms the basis of 3D reconstruction e.g. estimating the position of the 3D point from the known positions of the projections of this point in the image planes [3].

Several numerical methods have been developed for solving the problem for estimating the position of 3D points in the scene based on measurements of their projections on the two image planes. One common approach for estimating the position of the 3D points is to minimize the sum of the squared errors between the measured image positions and the re-projected positions applying pseudo-inverses or singular values decomposition numerical methods [1], [3], [4]. A major problem of this approach is that it fails to deliver consistent and stable results when the measurements are noisy. In such situation the RANSAC (RANdom SAmple Consensus) method is more appropriate [1], [3], [4]. It delivers a suitable result with a certain probability, which increases with the number of iterations. The big number of calculations which have to be performed in this case can be executed in parallel on GPUs [5].

B. The ZED Mini Stereoscopic Camera

To date, the technological achievement of 3D sensors has not allowed them to offer depth perception at longer ranges and outdoors. The ZED Stereo camera manufactured by Stereolabs is the first sensor to introduce longer range depth perception (up to 20m) indoors and outdoors [7]. This enables a variety of new applications in areas like robotics, augmented reality, security, etc. The ZED Mini camera allows for a smaller maximum supported range of 12-15m and due to the smaller distance between its two lenses which approximates the average human's interpupillary distance, it is better suited for seeing nearby objects.

The ZED camera reproduces the way humans' natural vision works. Human eyes are separated horizontally by about 65mm which means that each eye has a slightly different view of the scene. These different perspectives are merged by the brain to create the feeling of depth and 3D motion in space. The device uses its two lenses and triangulation to "understand" its surroundings and produce a 3D model of the observed scene. The output data consists of a high-resolution side-by-side colour video through a USB 3.0 interface. This video contains two synchronised left and right video streams which are used by the ZED application on the host computer to calculate a depth map of the scene. In a depth map the camera stores a distance value (Z, in metric units) for each pixel (X, Y) in the image which is calculated from the back of the left lens of the camera to the scene object [7](Fig. 2).

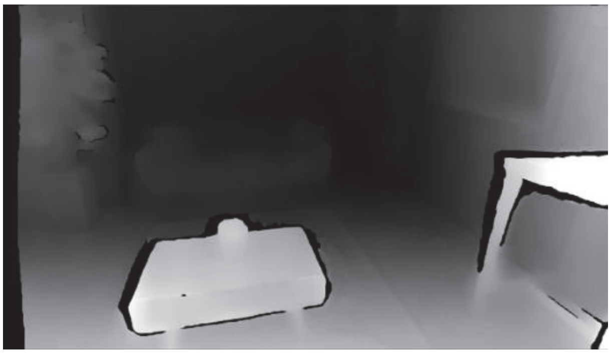

Fig. 2. A depth map created by ZED Mini [7]

The depth map is a 32-bit encoded image which cannot be displayed directly without converting it first to a monochrome image. This grayscale representation (8-bit) has values between 0 and 255 with 0 meaning the most distant possible depth value and 255 – the closest possible. A significant advantage of the device is its capability to create the depth map in real-time thus avoiding the need of a pre-scan of the environment [7].

III. Approaches for Spatial Configuration

In this paper we present three possible approaches for creating a virtual spatial configuration based on a real environment using the SDK of the ZED Mini stereoscopic camera. The purpose is to be able to create a virtual scene with objects that are aligned to the real environment and the physical objects in it. The three approaches we have identified are:

- Using hit tests against the calculated depth map;
- Using hit tests against reference planes generated according to surfaces identified in the depth map;
- Using hit test against a generated spatial map of the environment.

In all approaches a specialized tracked controller is used as a pointing device to generate a ray with which to specify the desired position where a virtual object should be placed.

A. Hit Testing Against the Depth Map

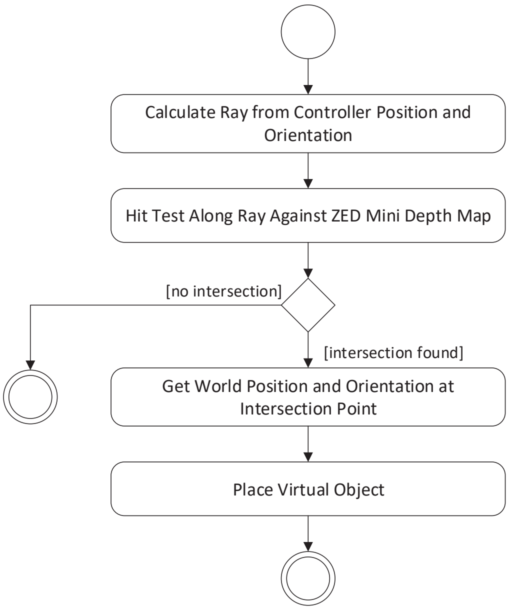
Calculate Ray from Controller Position and Orientation
Hit Test Along Ray Against ZED Mini Depth Map
[no intersection]
[intersection found]
Get World Position and Orientation at Intersection Point
Place Virtual Object

Fig. 3. Algorithm for placing a virtual object by hit testing against the ZED Mini generated depth map.

This approach follows the algorithm shown on Fig. 3. First a ray is generated starting from the tracked controller's position and in the direction of its orientation. This ray is passed to a method in the ZED Mini SDK which generates points with increasing distance (with a specified step) along the ray and for each point checks whether its distance to the camera is greater than the distance calculated in the depth map. If so a hit is detected, and the position of the point is returned. Then the normal at that position in the real environment is queried and an object is placed with the calculated (virtual) position and orientation.

B. Hit Testing Against Reference Planes

This approach is composed of two stages (see Fig. 4):

- First one or more "reference planes" are generated. The algorithm is similar to the hit test algorithm from the first approach but this time if a hit is found its position is passed to a method in the ZED Mini SDK which generates a plane containing all points from the depth map identified as belonging to the same surface.
- In the second step a hit test is performed against the generated reference planes and the virtual object is placed at the intersection point on the reference plane.

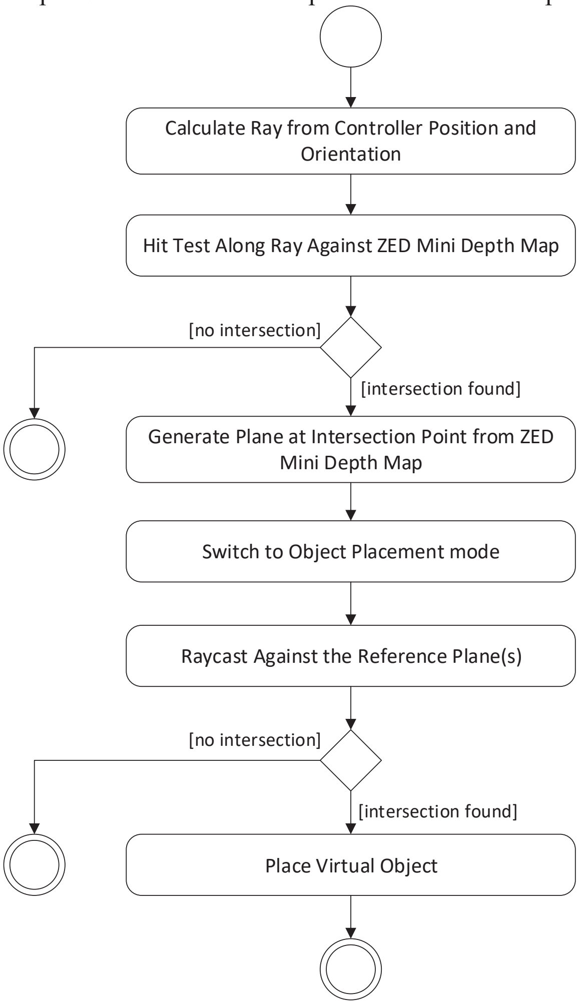
Calculate Ray from Controller Position and Orientation
Hit Test Along Ray Against ZED Mini Depth Map
[no intersection]
[intersection found]
Generate Plane at Intersection Point from ZED Mini Depth Map
Switch to Object Placement mode
Raycast Against the Reference Plane(s)
[no intersection]
[intersection found]
Place Virtual Object

Fig. 4. Algorithm for placing a virtual object using reference planes.

C. Hit Testing Against a Generated Spatial Map

The third approach is similar to the second one – it also consists of two stages, but instead of reference planes during the first stage the ZED Mini stereoscopic camera is used to generate a spatial map of the environment. When the spatial map is complete the virtual objects are placed by hit testing against the spatial map.

IV. Experiments

Series of Experiments were performed to validate the three implemented approaches. The experiments were conducted using a ZED Mini stereoscopic camera attached to the front of an HTC Vive Headset where HTC Vive's controllers were used for interaction.

The developed prototypes use 3D models of furniture for the creation of a virtual spatial configuration. Fig. 5 shows a sample virtual spatial configuration in a real environment.

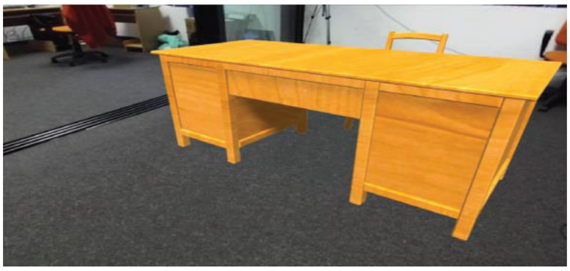

Fig. 5. Example of creating a virtual spatial configuration.

A. Using hit tests against the calculated depth map

Pressing the trigger on the Vive Controller generates a ray that detects where the point of the environment is and places the object oriented by the normal vector of the intersection point (Fig. 6).

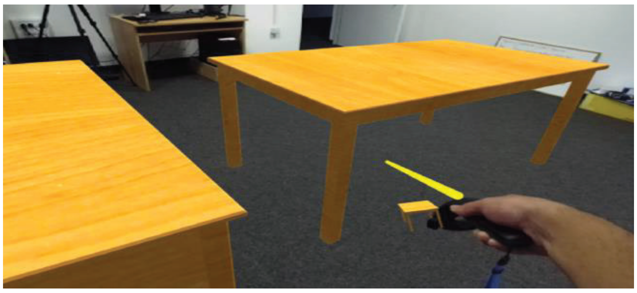

Fig. 6. Example of using hit tests against the calculated depth map.

B. Using hit tests against reference planes generated according to surfaces identified in the depth map;

In this approach when the trigger on the controller is pressed a reference plane is generated (Fig. 7). After that, virtual objects are placed on the desired position using the reference plane. Sometimes the hit point is created behind the real object and the plane is not positioned correctly to the surrounding environment. On the other hand, when the planes are generated properly several objects can be placed on one level (Fig. 8) which is more difficult to achieve with the previous method.

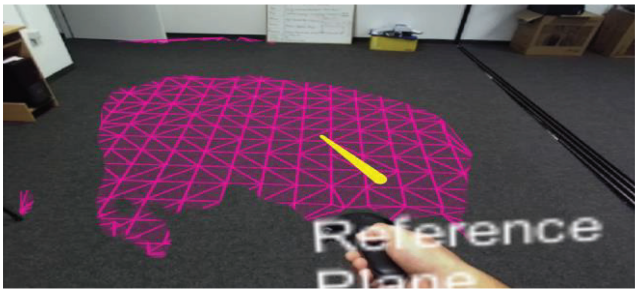

Fig. 7. Using real world hit test to generate a reference plane from surfaces identified in the depth map.

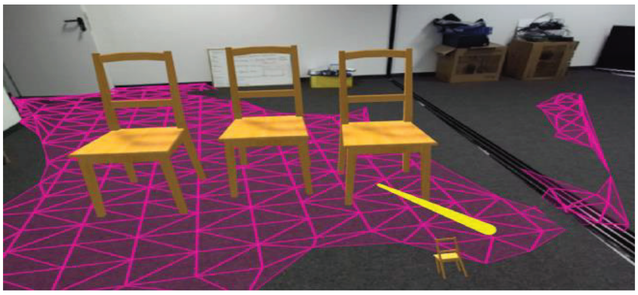

Fig. 8. Example of placing several objects on one plane.

C. *Using hit test against a generated spatial map of the environment.*

During the initial extraction the spatial map is created accurately, and the virtual spatial configuration can be created correctly. Once the user moves horizontally into space, the spatial map no longer coincides with the real environment, and the virtual scene with placed objects “swims” in the air at a slight distance from the ground. If the user moves vertically, the displacement from the ground in height can become significant.

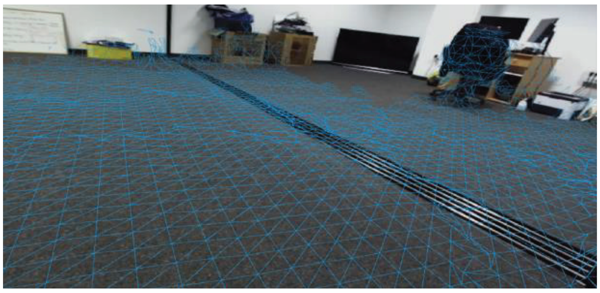

Fig. 9. Example of generating spatial map of the environment.

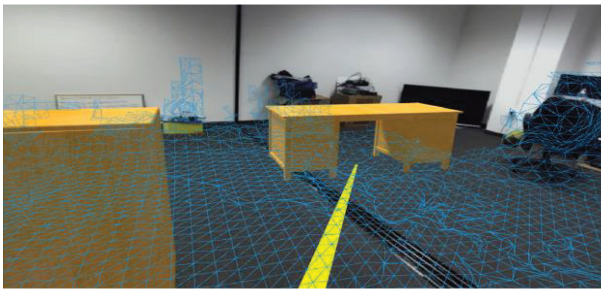

Fig. 10. Example of using hit test against a generated spatial map of the environment.

The conducted experiments showed that it is possible to use the ZED Mini stereoscopic camera in combination with an HTC Vive Headset to create a spatial configuration, but the results are not always satisfactory.

Acknowledgment

The results presented in this paper are obtained in the framework of research activities at the Virtual and Augmented Reality Laboratory, funded by the Research and Development and Innovation Consortium, Sofia Tech Park, project "Research and development of methods and algorithms for interaction in mixed reality environments performing tasks for spatial configuration and collocation of objects".

References

[1] R. Hartley, A Zisserman, *A Multiple View Geometry in Computer Vision*, 2-nd ed., Cambridge University Press, 2003.

[2] M. Aladem, *Robust Real-Time Visual Odometry for Autonomous Ground Vehicles*, Master Thesis, University of Michigan-Dearborn, 2016.

[3] R. Hartley, P. Sturm, "Triangulation", *Computer vision and image understanding*, Vol. 68, Nr. 2, pp. 146-157, 1997.

[4] Y. Gao, *A user-oriented markerless augmenter reality framework based on 3D reconstruction and loop closure detection*, Ph.D. thesis, University of Birmingham, 2016.

[5] S. Ouerghi, R, Boutteau, X. Savatier, F. Tlili. "CUDA Accelerated Visual Egomotion Estimation for Robotic Navigation". *12th International Conference on Computer Vision Theory and Applications*, Feb 2017, Porto, Portugal. SCITEPRESS - Science and Technology Publications, Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, pp.107-114, <10.5220/0006171501070114>.

[6] M. Senthilvel, R. Soman, K. Varghese, "Comparison of Handheld devices for 3D Reconstruction in Construction", *34-th International Symposium on Automation and Robotics in Construction (ISARC)*, 2017.

[7] Depth Sensing, https://www.stereolabs.com/docs/depth-sensing/, as seen on 18.11.2018