Object-Based Mobile Augmented Reality for a 3D Model

ROSANNY SIHOMBING[1] & VOLKER COORS[1]

Abstract: Augmented Reality (AR) is a convincing technology for manipulating user's visual perception of the real world. A single image-based tracking approach is not sufficient to recognise a 3D object from different viewpoints. The basic issue of this approach is it can no longer recognise a 3D model when the viewing perspective is changed since the image captured by the camera is no longer match the reference image. This paper presents an object recognition approach to track markerless 3D building blocks as the AR target. This was achieved by extracting the feature points of the building blocks through a preliminary scanning process using Vuforia Object Scanner. Then a dataset of these feature points is used to recognize the building blocks so that the tracking process can be performed and virtual models can be imposed. To make the AR scene more realistic, the virtual model shadow is added using shadow mapping approach. The experiment shows the presented method is able to recognize the building blocks from side to side, which leads to a robust tracking performance, while the shadow mapping successfully adds realistic shadows to the scene. This robust performance also resulting in a stable appearance of the augmented virtual model.

1 Introduction

A complex and large physical 3D city model is often used as a tool for urban planning, since it maps important characteristics of the city, such as green areas, main roads, buildings and towers, and another significant landmarks (Figure 1). However, due to its complexity, any changes made to the city model would be a costly process in terms of money, time, and resources. Therefore, an inexpensive and simple way to display possible addition or modification of the physical city model would give the city planners efficiency and effectiveness they need when doing urban planning.

The basic concept of AR is to manipulate human visual perception of the real world by mixing real object and computer-generated virtual object in the real world, in such a way that they are accurately aligned in real-time (AZUMA et al. 2001). Over the last decade, both operating system and hardware quality of mobile devices are increasing rapidly, making them capable of doing all heavy computations needed for AR (BILLINGHURST et al. 2015).

1.1 Previous Work

On our previous work (SIHOMBING 2016), we have implemented an AR mobile application using a markerless image-based approach for indoor urban planning. It enables multiple city planners to view and assess city plans directly on a 2D city map or an indoor 3D physical city model in real-time using an android device (Figure 2). It was done by obtaining the feature points of the

---

[1] Hochschule für Technik Stuttgart, Schellingstr. 24,
D-70174 Stuttgart, E-Mail: [rosanny.sihombing, volker.coors]@hft-stuttgart.de

reference image and utilise them to recognize the object target before performing the tracking and adding the virtual model to the AR scene.

While this method gives a robust tracking performance and a convincing result with the 2D city map, it does not perform well with the 3D city model (Figure 3). The application ability is limited to the scene recognition from a particular viewpoint when using the 3D city model as a target. This limitation happened due to a perspective issue when using a single image as a reference for a 3D object. When the viewing perspective changes, so does the 3D object image captured by the camera. As the result, the 3D object cannot be recognised anymore through its reference image. Therefore, on this case a single reference image is not sufficient to have a robust tracking performance. To solve this problem and improve what has been achieved, a more suitable recognition approach needs to be applied.

The main goal of this paper is improving the result we have achieved (SIHOMBING 2016) by (1) using a 3D city model as the object target for urban planning and (2) rendering the shadow of the virtual 3D model in order to add realism to the whole scene. The remaining sections of this paper is structured as follows. Section 2 explains the methods to achieve the objective. The experiments and corresponding evaluations are presented on section 3, while the conclusion and future works can be found on section 4.

Fig. 1: Physical city model of Hamburg [source: www.hamburg.de/stadtmodell]

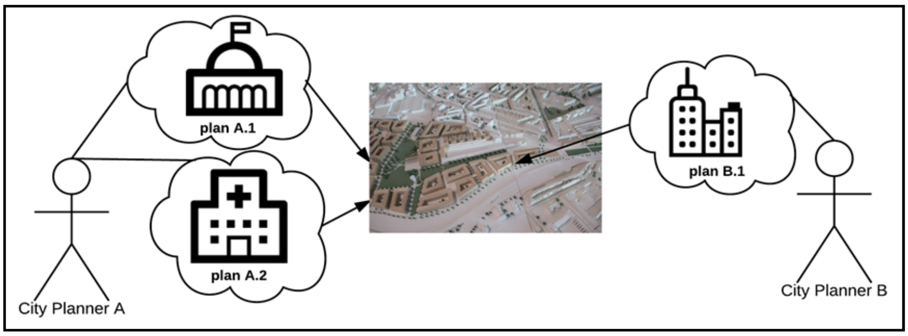

Fig. 2: City planners with their own ideas on plans on the physical city model

Fig. 3: 2D image-based approach for both city map and physical building blocks. (above) An A4 size 2D map is used as the target, which can be recognised quickly and the virtual building blocks can be imposed without being disappear. (below) A 2D reference image of untextured building blocks is used as the target. The physical building blocks can be recognised only from a very limited perspective and the virtual model was clearly misplaced.

2 Methods

The recognition method presented on this paper aim to recognise and track untextured building blocks, which act as the object target, so that the augmented virtual model can be seen from a broader perspective. On the next step, the shadow of the virtual model will be casted to the ground to level up the realism.

2.1 Recognising 3D Object

The more complex geometric shape a 3D object has, the more feature points it has. These feature points determine how quick a 3D object can be recognised. We utilize Vuforia Object Scanner (PTC INC. 2018) in order to obtain 3D object feature points by scanning the object. After gathering the feature points, a quick test can be performed to check whether or not the 3D object can be recognised through those feature points. If the desired performance has not achieved yet, then more feature points must be gathered by continuing the previous scanning process (Figure 4). The detail of this process can be found on (PTC INC. 2018).

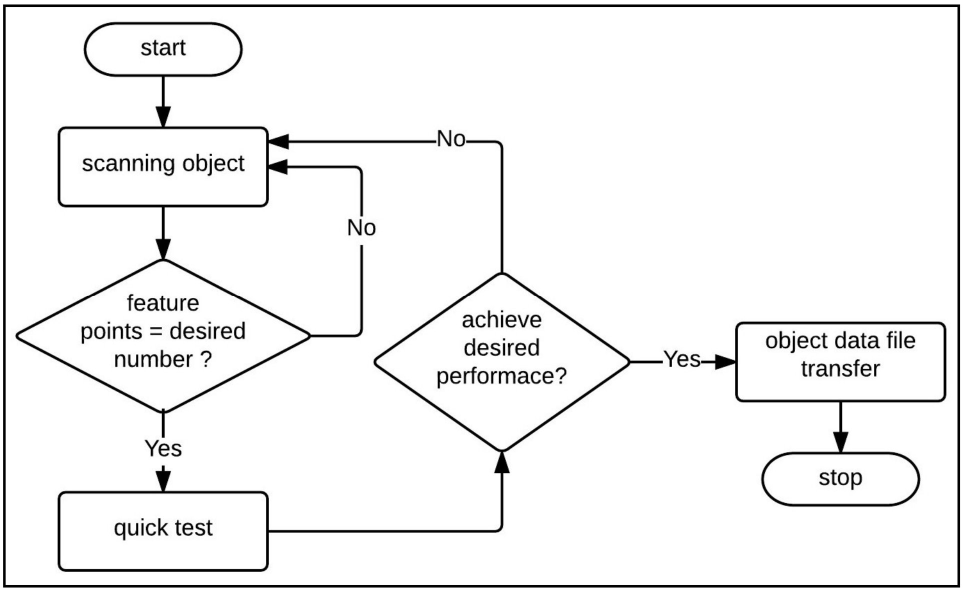

Fig. 4: Scanning process to obtain feature points

2.2 Tracking The Target and Placing The Virtual Model

Tracking is a looping process which starts as soon as the object target is recognised. When the tracking is running, the pose matrix is always updated depending on the camera pose (Figure 5). Pose matrix represents the pose of an object target as seen from the camera. In eq. 1 this matrix is denoted by $\boldsymbol{P}$ and calculated based on the camera pose towards the object target. This 3x4 matrix consists of a 3x3 rotation matrix, denoted by $\boldsymbol{r}$, and a 3x1 translation matrix, denoted by $\boldsymbol{t}$. The rotation matrix shows how the object target is rotated, while the translation matrix indicates the object target position relative to the camera.

$$P = \begin{bmatrix} r11 & r12 & r13 & tx \\ r21 & r22 & r23 & ty \\ r31 & r32 & r33 & tz \end{bmatrix} \tag{1}$$

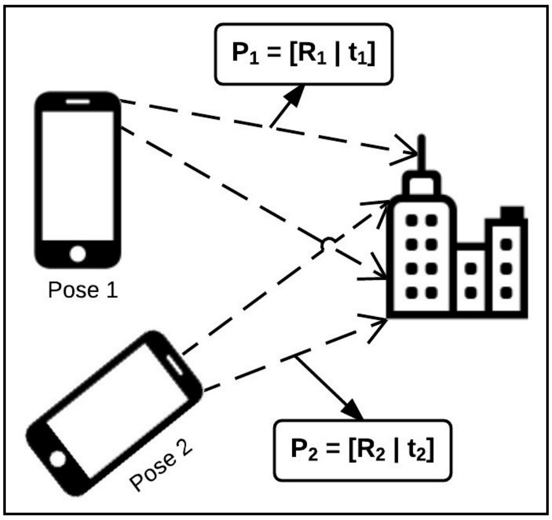

Fig. 5: Camera pose and pose matrix

To place the virtual model in the desired location, it must be transformed so that its centre point match the centre point of the 3D object target. The same requirement goes to the scale and orientation of the virtual model, they need to be transformed so the virtual model fits the 3D object target. These transformations process can be done manually in Unity3D (UNITY 2018) using its Graphical User Interface. The virtual model used on this paper is provided by the Hamburg City and reconstructing one is outside the scope of this paper.

2.3 Shadow Mapping

As shadow is important for human spatial perception, casting the shadow of the virtual is expected to level up the realism of the AR scene (SUGANO et al. 2003). The technique used to cast the shadow is the common shadow mapping technique (WILLIAMS 1978). To implement this technique, it needs to be adjusted to AR where a shadow receiver that is invisible to the user must be added to the scene. Therefore, an invisible planar object which acts as the ground must be created and placed in world space centre (0, 0, 0) under the virtual model, while the existing virtual model is assigned as the shadow caster and also receiver. The light source's position is configured manually. This invisible shadow receiver approach on AR is similar to (CASTRO et al. 2012) and also (HALLER et al. 2003) even though shadow volume technique was used to cast the shadow. The following steps describe the approach to map virtual shadow on this paper:

1) Create an invisible planar plane as the ground. Put it on coordinate (0, 0, 0) and assign it as shadow receiver.
2) Define light source position.
3) Assign the virtual model as both shadow caster and receiver.

3 Experiments and Evaluations

The mobile application has been implemented using Vuforia SDK (PTC INC. 2018) as the AR framework and Unity3D (UNITY 2018) as the 3D engine on a Nexus 9 tablet running Android 6.0.1. After the implementation, we had 2 major experiments which aim to (1) demonstrate the tracking performance of the proposed approach and (2) evaluate the shadows in regard to making the whole AR scene more realistic. Both experiments used untextured 3D model provided by The City of Hamburg as the object target (Figure 6). This 3D model is a copy of some part of Hamburg physical city model as pictured in fig. 1.

Fig. 6: 3D object target

3.1 Tracking

3.1.1 Experiment

The virtual model for this experiment consists of 8 textured virtual building blocks in 3DS file format and provided by The City of Hamburg. These 8 virtual objects are 1 entity which depends on each other. The virtual model was imposed to the object target just like a scenario of urban planning which already explained on Section 1 and pictured in fig. 2. Two building blocks of the virtual model correspond to 2 building blocks of the object target. It is expected that these 2 virtual building blocks are completely projected to the corresponding object target building blocks, while the other 6 virtual building blocks are projected to their corresponding map location. The result and demo video of this experiment is available at the following URL: https://vimeo.com/221005614. Some images extracted from the video are presented in this paper as well (Figure 7).

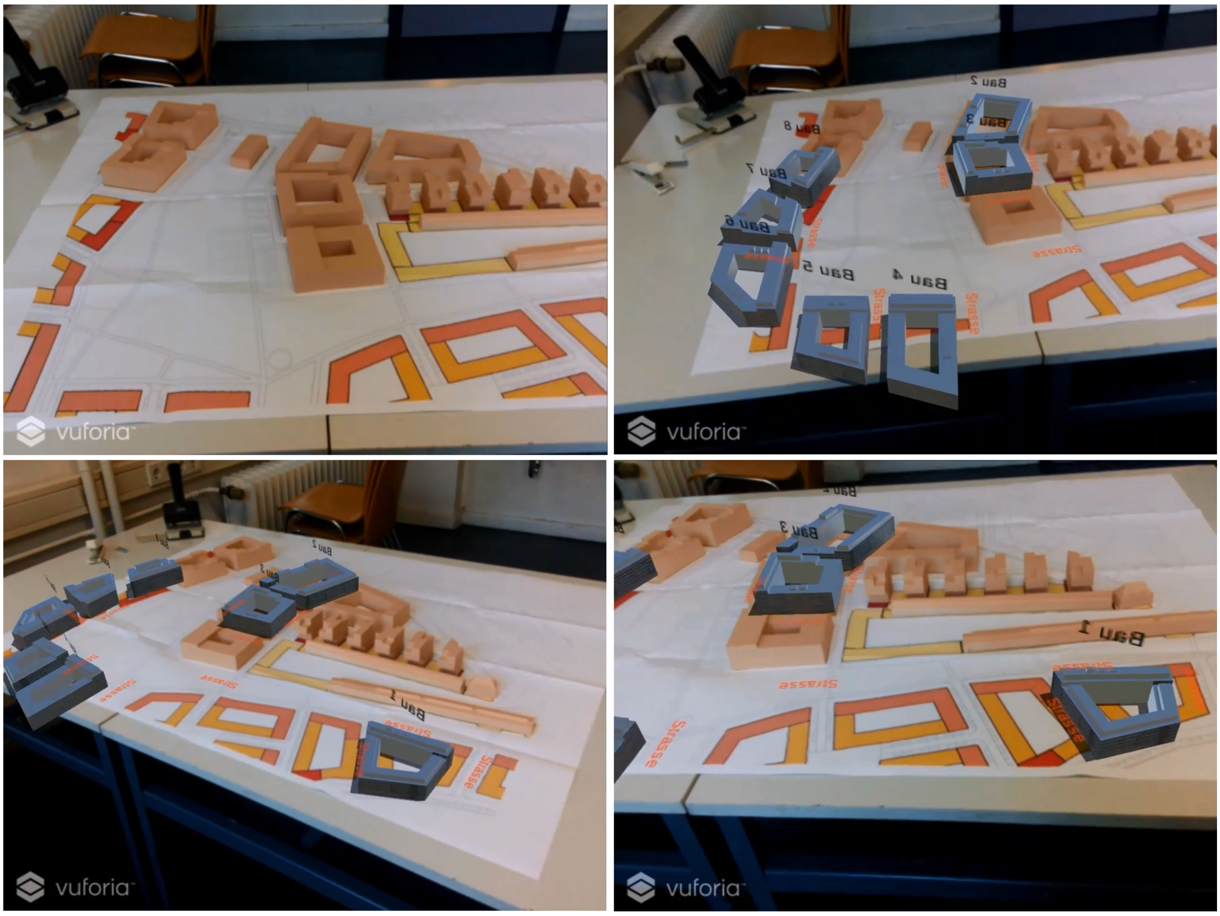

Fig. 7: (top-left) Mobile application is getting started. (top-right) The virtual is imposed to the object target as soon as the target is recognized. (bottom-left) Tracking the object target from another viewpoint. (bottom-right) Getting closer to the object target.

3.1.2 Evaluation

As shown in the video, the experiment shows the untextured 3D model can be recognised as the object target and the virtual model can be imposed to the target. The tracking performance was convincing enough since the virtual model can be viewed from side to side without flickering or being disappeared during the tracking. This performance gives the users more flexibility in observing the virtual model since they can move around to see it from different viewpoints. However, while 2 virtual buildings in the middle can be projected accurately, the other 6 virtual buildings were slightly misplaced.

To inspect this accuracy issue, we manually map the virtual model to the 90cm x 110cm 2D map of the target object location using a GUI tool in order to see the desired location of each virtual building on the map and how accurate the corresponding location is. The mapping (Figure 8) shows 6 virtual buildings do not overlap perfectly with their corresponding map location, while the other 2 virtual buildings in the middle match very well. Therefore, this accuracy issue is still acceptable since the issue lies on the virtual model rather than an issue with the tracking or locating the virtual model.

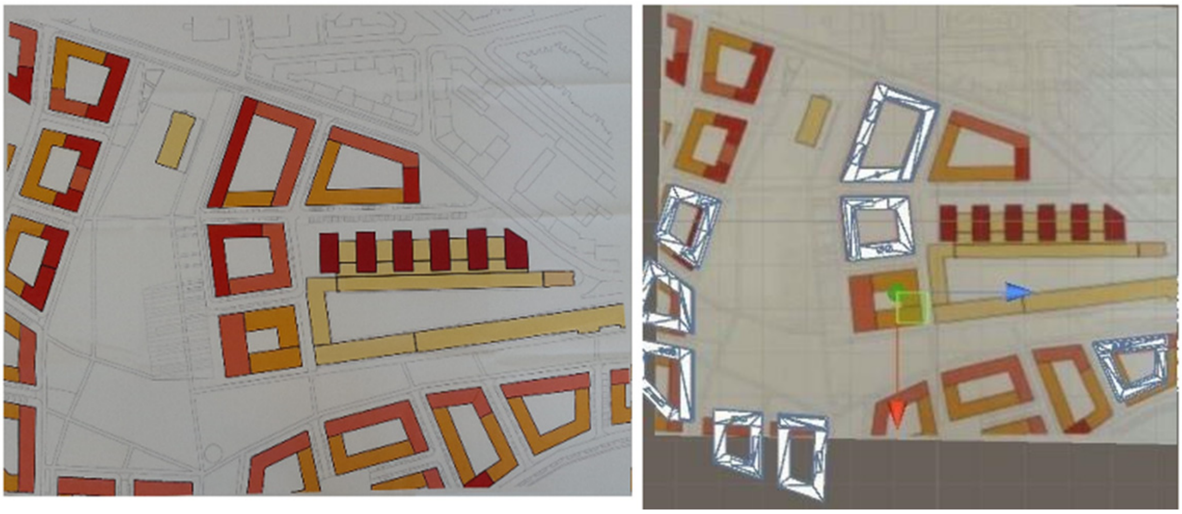

Fig. 8: (left) The location map of the 3D model. (right) Six virtual model do not correspond accurately to their location, while the other 2 can be located accurately.

3.2 Shadow Mapping

3.2.1 Experiment

Shadow Mapping was implemented by using Unity3D. On the previous experiment, the shadow of the virtual model was already casted to the ground. However, the shadow was not so visible due to the height of the virtual model. To make it more obvious, we used a cube and a cylinder imported from Unity3D library as the virtual models. The cube and the cylinder are higher than the actual virtual model and were placed next to each other in the AR scene. These addition virtual models were assigned as both shadow caster and shadow receiver (Figure 9). There was no particular point registration needed in this experiment since the objective is only to see how well the shadow mapping works when the virtual models are placed on the ground (x, y, 0).

3.2.2 Evaluation

As pictured in fig. 9, the shadow is clearly visible and mapped realistically to the invisible ground and to the visible virtual object. This promising performance is also shown in the tracking experiment, in which the shadows of the virtual buildings can be seen as well. The shadow appearances are different which reflect the shadow behaviour in the real world. Furthermore, the shadow also helps to emphasize the spatial relationship between the objects.

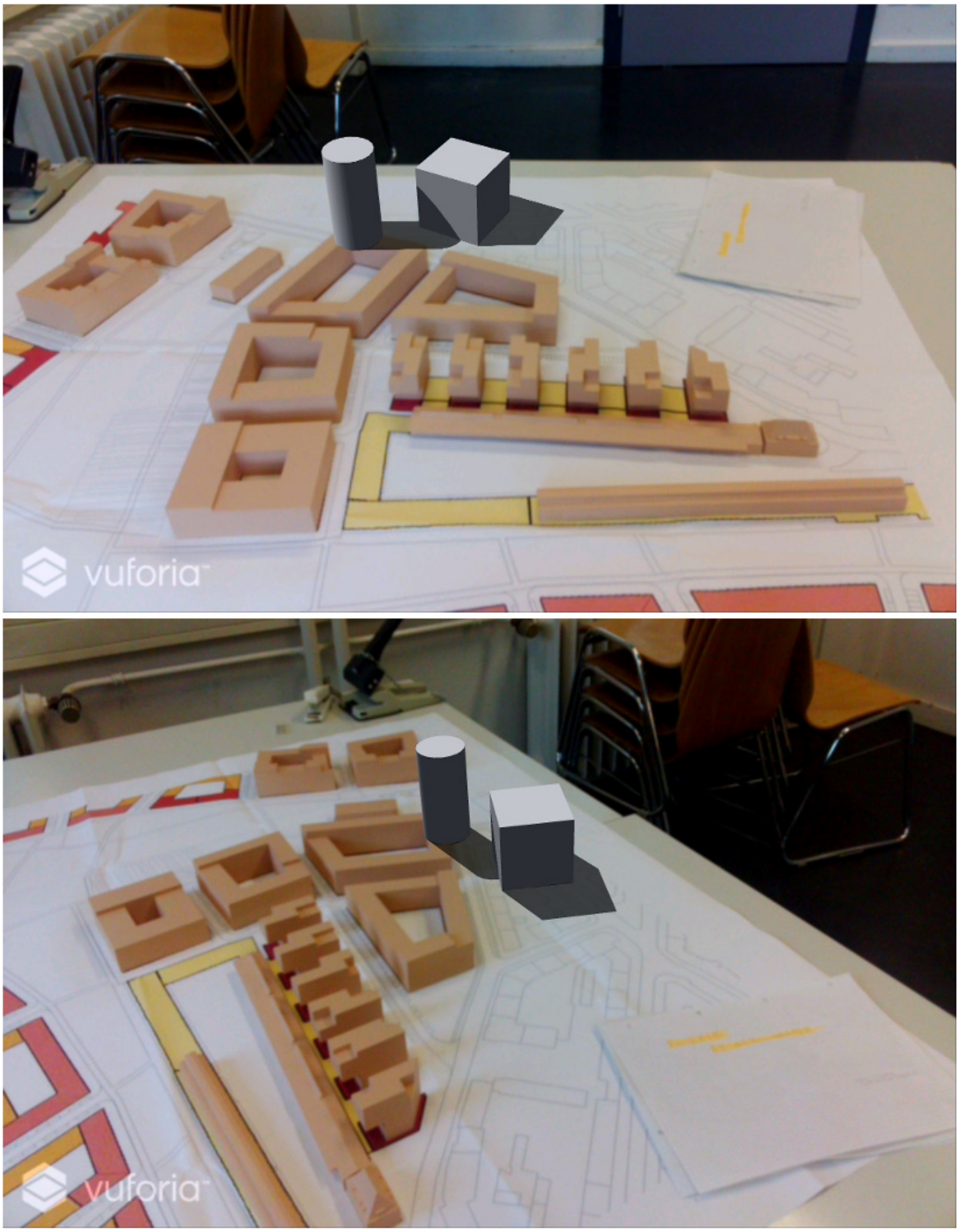

Fig. 9: A cylinder and a cube are assigned as both shadow caster and receiver, so each can also receive shadow from another virtual object.

4 Conclusion and Future Work

Object-based mobile AR for a 3D model and shadow mapping have been implemented. The 3D object recognition approach gives a convincing performance and result when working with 3D model. It is able to track the 3D model as the object target from broader viewpoint. As the result, the augmented 3D model can be viewed without being disappeared. The augmented shadow made the AR scene more realistic by emphasizing the spatial relationship between the object target, the virtual object and the ground, which is essential for urban planning.

As already mentioned, locating the virtual model on this paper were done manually. Therefore, in our further research we would like to make this process automatic so that it would be easier to add another virtual models to the AR scene. It is also our interest to have a real-time shadow and map it to the real objects so that the interaction between real and virtual object can be more realistic.

5 Acknowledgement

We would like to thank Dipl.-Ing. Walter Sieh and his team at Landesbetrieb Geoinformation und Vermessung of Hamburg city for their support and also for providing the 3D model, so that it was possible to have the experiments in Stuttgart.

6 References

AZUMA, R., BAILOT, Y., BEHRINGER, R., FEINER, S. K., JULIER, S. & MACINTYRE, B., 2001: Recent Advances in Augmented Reality. IEEE Computer Graphics and Applications, **21**(6), 34-37.

BILLINGHURST, M., CLARK, A. & LEE, G., 2015: A Survey of Augmented Reality. Foundations and Trends® in Human–Computer Interaction, **8**(2–3), 73–272.

CASTRO, T. K. D., FIGUEIREDO, L. H. D. & VELHO, L., 2012: Realistic shadows for mobile augmented reality. In Proceedings - 2012 14th Symposium on Virtual and Augmented Reality, SVR 2012, 36–45.

HALLER, M., DRAB, S. & HARTMANN, W., 2003: A real-time shadow approach for an augmented reality application using shadow volumes. Virtual Reality Software and Technology, 56.

PTC INC., 2018: Vuforia Object Scanner. Accessed August 2017. https://library.vuforia.com/articles/Training/Vuforia-Object-Scanner-Users-Guide.

SIHOMBING, R., 2016: Development of Image-Based Mobile Augmented Reality Application For City Map and City Model. MSc Thesis, Hochschule für Technik Stuttgart.

SUGANO, N., KATO, H. & TACHIBANA, K., 2003: The effects of shadow representation of virtual objects in augmented reality. In Proceedings - 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR 2003, 76–83.

UNITY, 2018: Unity Game Engine. Accessed August 2017. https://unity3d.com/de.

WILLIAMS, L., 1978: Casting Curved Shadows on Curved Surfaces. ACM SIGGRAPH Computer Graphics, **12**(3), 270-274.