VIRTUAL ELEMENT RETRIEVAL IN MIXED REALITY

M. Radanovic[1,2,*], K. Khoshelham[1,2], C. Fraser[2]

[1] Building 4.0 CRC, Caulfield East, Victoria, Australia
[2] Department of Infrastructure Engineering, The University of Melbourne, Parkville, Victoria, Australia
(m.radanovic, k.khoshelham, c.fraser)@unimelb.edu.au

KEY WORDS: Augmented reality, Mixed reality, BIM, Element retrieval, Localisation, Anchoring.

ABSTRACT:

The application of mixed reality visualisation in construction engineering requires accurate placement and retrieval of virtual models within the real world, which depends on the localisation accuracy. However, it is hard to understand what this means practically from localisation accuracy alone. For example, when we superimpose a Building Information Model (BIM) over the real building, it is unclear how well does a BIM element fit the real one and how small a BIM element are we able to retrieve. In this paper, we evaluate virtual element retrieval by designing an experiment where we attempt to retrieve a set of cubes of different sizes placed in both the real and the virtual world. Furthermore, inspired by existing camera localisation methods for indoor MR being almost exclusively image-based, we use a localisation approach based solely on 3D-3D model registration. The approach is based on the automated registration of a low-density mesh model of the surroundings created by the MR device to the existing point cloud of an indoor environment. We develop a prototype and perform experiments on real-world data which show high localisation accuracy, with average translation and rotation errors of 1.4 cm and 0.24°, respectively. Finally, we show that the success rate of virtual element retrieval is closely related to the localisation accuracy.

1. INTRODUCTION

One of the important applications of Building Information Models (BIMs) is building life cycle management where a BIM is used as a live digital database where all building information can be stored. However, the interaction between the BIM and the real world is not straightforward. The conventional approach is to inspect the building in the real world and record the information in the BIM in a separate process which requires inferring the correspondence between the virtual elements and the real world. Mixed reality (MR) provides a means to do this in one step and with inherent correspondence. In MR, virtual objects are not just overlaid on top of the real world but anchored within it. This enables interaction between them, such as a virtual object being occluded by a real-word object (Muthalif et al., 2022), and allows us to interact with both the virtual BIM and the real world simultaneously. However, the virtual content must be placed precisely within the real world for their blending to be smooth. (Çöltekin et al., 2020) The pre-requisite for this is accurate global localisation, however, the relationship between the localisation accuracy and the success of virtual element (e.g. BIM element) retrieval in MR is not well understood.

Camera localisation is one of the most researched topics in the field of MR (Kim et al., 2018). Global camera localisation, which is required for large-scale applications, is dominated by vision-based methods (Marchand et al., 2016). However, because they rely on image features, vision-based methods are susceptible to changes in light conditions, reflections, dynamic objects and other scene changes (Melekhov et al., 2017; Parisotto et al., 2018). Furthermore, they require the building of specific feature libraries which affects their performance as the area covered increases in size. These drawbacks make vision-based methods unsuitable for accurate large-scale indoor localisation. For example, Li et al. (2020a) report an average positional error for image-based pose regression localisation with *PoseNet* of 0.66 m for a roughly 10 m^2 office space, and a 0.35 m error for an ORB feature-based localisation. Moreover, industry solutions such as Microsoft's Azure Spatial Anchor[1] feature-based localisation works only in a small local area, such as a wall or a table. For large-scale applications, the documentation suggests creating individual anchors (and individual experiences) for each specific local area where a virtual object is desired and establishing their spatial relationship with the device's tracking system. Placing multiple anchors for different lighting conditions is also suggested.

These conditions clearly inhibit the development of large-scale MR applications. On the other hand, there have been significant improvements in the field of automated coarse 3D-3D model registration brought by learning-based solutions in the past few years. Although these methods have been successfully applied for the localisation of autonomous LiDAR-equipped (Light Detection and Ranging) vehicles, they have not been studied for MR localisation.

Furthermore, certain MR applications require accurate placement and retrieval of virtual models within the real world. In the above example of a real-world BIM model, an accurate anchoring of BIM elements is an obvious requirement. This becomes even more important if the virtual information is invisible until called for. For example, a BIM might be invisible in the case where it covers real-world information, or an art piece might be left unobscured by virtual data, and only after we interact with the real-world object, i.e. by clicking on it, the expected virtual information is retrieved and visualised. However, it is not well understood how the retrieval success rate relates to localisation accuracy.

Localisation accuracy is usually reported as an average distance between the estimated 3D camera position and the ground

* Corresponding author.

[1] Azure Spatial Anchors documentation: `https://docs.microsoft.com/en-us/azure/spatial-anchors/` (accessed 11 Jan. 2022).

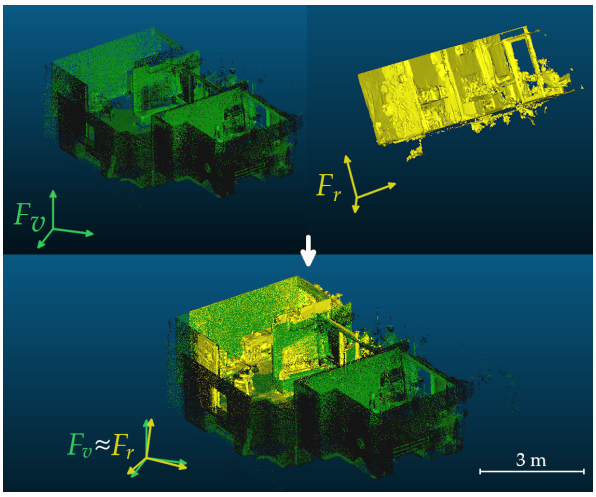


Figure 1. The registration of the real world frame (yellow) represented by the polygonal mesh model captured by the MR device (location 1 of the testing site) to the virtual world frame (green) represented by the reference point cloud.

truth 3D camera position. However, this ignores the rotation error which is equally important in evaluating the accuracy with which the virtual object is placed, i.e. anchored within the real world. Furthermore, it is unclear what is the size of the object one can reliably retrieve with the MR device from localisation error only. For example, does a localisation error of 35 cm guarantee that a 35 cm virtual object can be retrieved with a 100% success rate?

The main aim of this paper is to answer these questions by empirically testing the relationship between the localisation accuracy and the size of a virtual element one can retrieve. To this end, we use an MR localisation method based purely on learning-based 3D-3D model registration. The method performs global localisation by registering the low-density untextured 3D model of the surroundings, created by the MR device, to the existing reference 3D model. The prototype of the proposed localisation method, built with Unity and deployed on the Microsoft HoloLens (1st gen), is used to assess and compare the localisation and the retrieval success rate. The retrieval is empirically evaluated by attempting to retrieve invisible virtual element cubes of different sizes (60, 30, 15, 8, 4, 3, 2 and 1 cm) after localisation and compared to the localisation accuracy achieved at the same testing site.

2. RELATED WORK

Previous research on virtual object retrieval in mixed reality mainly focuses on camera localisation. Estimating the pose of the camera within a reference coordinate system, i.e. camera localisation, can be local or global. To be able to place virtual objects in desired predefined places within a known model of the environment, global or absolute localisation is needed. Having a pre-existing 3D model of the environment facilitates the development of large-scale indoor MR applications, such as museum exhibitions or real-world BIMs, as it enables two important functions. First, it enables the placement of virtual objects precisely where they are desired to be in the real world, for example, a painting can be placed on the wall, a BIM window can be aligned with the real window, or a virtual guide can have its pathfinding rules set. Second, a pre-existing 3D model enables global camera localisation within it, which also anchors the virtual objects in their predefined places. Global localisation approaches without external sensors, i.e. infrastructure-independent approaches, can be image-based (also called vision- or camera-based) or based on registration in 3D space.

Image-based methods are by far the most commonly used ones (Marchand et al., 2016) and they can be split into feature-based, model-based, image retrieval and image-based pose regression. Feature-based approaches localise the query image by first establishing correspondences between the extracted 2D features and their 3D coordinates in the reference 3D scene reconstruction, which has been previously generated from images. A wide variety of local feature descriptors have been proposed, such as ORB (Rublee et al., 2011). Matching keypoints are then used to estimate the camera pose with perspective from points methods (PnP) (Quan and Lan, 1999). For example, Li et al. (2020a) build a 3D feature database of the environment from image and depth data where each 3D point is tied with its ORB descriptor. Global localisation is achieved by extracting ORB features from the query image and matching them with the database. They achieve 35 cm localisation accuracy for an independent image with varying light conditions. In general, the reliance on image-based point features introduces limitations such as robustness to reflections and other changes in lighting, large variations in viewpoints, lack of texture, repetitive structures and dynamic objects, or, in other words, the robustness to changes in the scene.

Model-based approaches use lines or other types of shapes present in a CAD or a BIM model and minimise the distance between their projection and their matched detected contour points. For example, Acharya et al. (2019b) achieve a 10-centimetre accuracy of localisation by rendering the visible edges from a BIM model, sampling their points and projecting them to the image plane, where correspondences with structural edges detected in a query image are found. Similarly, Petit et al. (2012) find and use correspondences between image edges and 3D model edges after rendering for frame-by-frame tracking. However, these approaches assume a known initial pose which may be complex. A good overview of BIM-based AR and MR methods is given in (Sidani et al., 2021), where it is noted that there is no general system architecture for localisation within a BIM but a wide variety of different approaches.

Image retrieval approaches compare the query image to a database of geo-tagged images to estimate its location. The database consists of either previously captured images or those generated from a digital model. In a two-step process, the image features of both the query and database images are described, followed by a similarity association across the description vectors (Piasco et al., 2018). For example, Baek et al. (2019) propose an augmented reality (AR) facility management system that overlays BIM elements on top of the real world. The localisation is performed by comparing the query image to the database of images generated from the BIM model, but these are created only in the expected case study area and from the expected angles. This susceptibility to viewpoint changes is one of the downsides of image retrieval approaches, along with the requirement to build a large dataset of geo-tagged images (Acharya et al., 2019a).

Image-based pose regression approaches directly regress from query image to its corresponding pose (Piasco et al., 2018) and

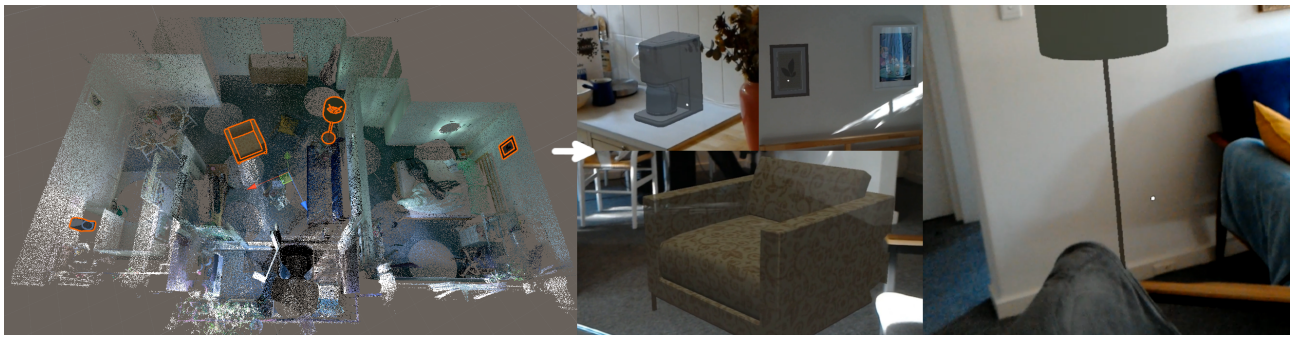


Figure 2. Anchoring of virtual objects, outlined in orange within the reference point cloud to the left, within the real world. Any number of virtual objects can be placed anywhere within the reference point cloud and they are anchored with high accuracy and without stuttering. The occlusion of objects needed for MR applications can be seen on the armchair and the lamp.

are dominated by Convolutional Neural Network (CNN) approaches that learn the mapping from images to their global poses (Herbers and König, 2019). Kendall et al. (2015) propose a CNN regressor *PoseNet* made by fine-tuning a network (originally trained for classification of images) on samples with poses obtained with Structure from Motion (SfM) algorithms. Many proposed approaches build upon *PoseNet*. For example, Acharya et al. (2019a) fine-tune *PoseNet* and train it on a synthetic dataset of computer-generated images of an indoor BIM model, to avoid the requirement of an SfM reconstruction of the scene which they see as a major limitation. However, although image-based pose regression approaches have good tolerance towards changes in the scene geometry (Piasco et al., 2018), their main downside is the localisation accuracy which is generally in the range from 0.5 to 3 m for indoor scenes (Walch et al., 2017).

Although not used as commonly as image-based methods, global camera localisation can be solved by registering the local 3D model captured by the MR device (most commonly by a depth camera) to an existing global or reference model. As registering the local model, i.e. the current frame to the global model, is equivalent to finding the six degrees of freedom (DoF) camera pose relative to it (Marchand et al., 2016), in a sense, the challenge of global MR localisation shares similarities with the 3D model registration challenge (Herbers and König, 2019). 3D-3D model rigid registration is divided into coarse registration (initial rough alignment) and fine registration (subsequent refinement and optimisation of coarse registration). The field of automated fine registration is dominated by the well-established Iterative Closest Point (ICP) algorithm (Besl and McKay, 1992) and its efficient variants (Rusinkiewicz and Levoy, 2001). On the contrary, automated coarse registration is still an open challenge (Bueno et al., 2018).

Coarse 3D model registration is based on 3D-3D correspondences between points, and central to finding these correspondences are 3D local feature descriptors. Recently, researchers have focused on the development of descriptors in a data-driven manner, especially over the past five years, as these have been shown to outperform hand-crafted descriptors (Li et al., 2020b; Guo et al., 2020; Choy et al., 2019; Zeng et al., 2017; Poiesi and Boscaini, 2021). Networks for learning local 3D descriptors can be split into patch-based or local and fully convolutional or global networks. Patch-based networks process a patch of a point cloud. For example, *3DMatch* proposed by Zeng et al. (2017) uses a 3D CNN to map to a feature vector from a local volumetric 3D voxel grid, while the rotation-invariant local deep descriptor (DIP) proposed by Poiesi and Boscaini (2021) canonicalises patches through their local reference frame. On the other hand, fully convolutional networks process the whole point cloud. Choy et al. (2019) propose *FCGF*, the first fully-convolutional single-pass network which uses convolution on sparse tensors to create rotation-invariant dense descriptors. Another example is *D3Feat* proposed by Bai et al. (2020), a fully convolutional network that jointly learns 3D feature descriptors and detectors to be able to find well-localised points. Repeatable keypoints are obtained with a density-invariant keypoint selection strategy and a detector score for each point of the cloud. In general, deep learning on point clouds is still in its infancy (Guo et al., 2020) and new methods are constantly being proposed. However, we are

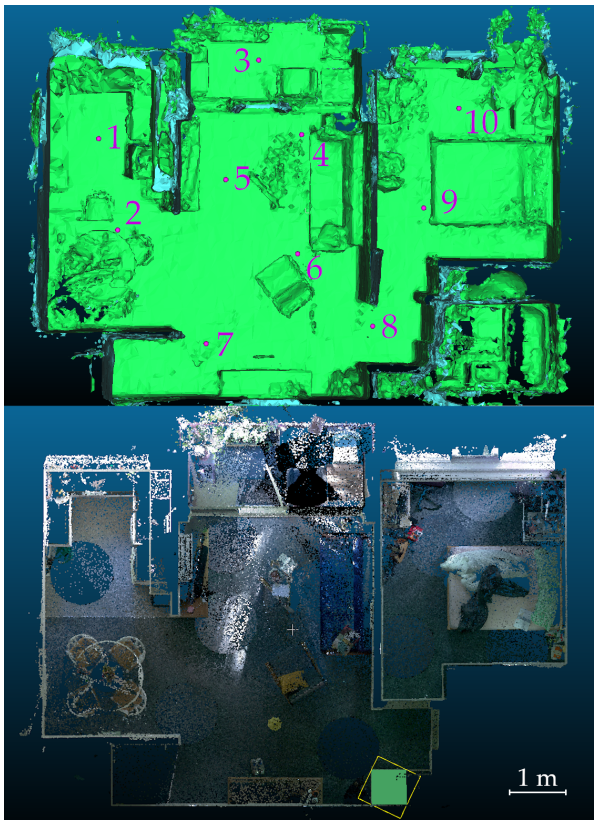
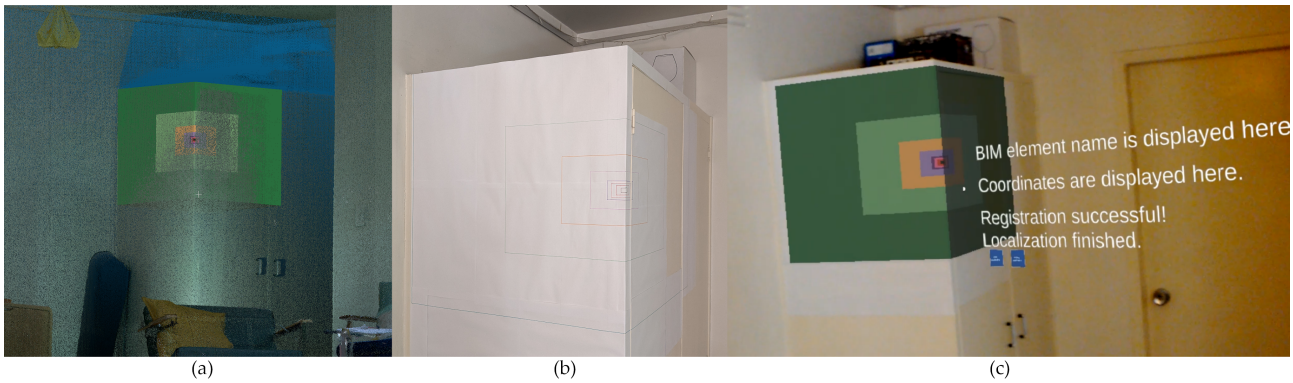


Figure 3. Models of the testing site. Top: a polygonal mesh model created with the Microsoft HoloLens (1st gen) with testing locations marked. Bottom: a LiDAR point cloud with the cubes used for retrieval success rate evaluation.



(a) (b) (c)

Figure 4. Cubes used for the empirical evaluation of retrieval success rate in (a) the virtual world, (b) the real world and (c) in mixed reality with the worlds aligned.

witnessing a successful application of the recent advances in learning-based 3D-3D model registration to the localisation of self-driving vehicles, for example in (Zhang et al., 2021). But, these are always combined with other external sensors such as global navigation systems which are not available indoors. To the best of our knowledge, there have been no attempts at global MR camera localisation systems based solely on coarse-to-fine 3D-3D model registration.

3. LOCALISATION METHOD

The MR system combines the real and the virtual world. The main idea of the proposed localisation method is to register and align the 3D model representing the real world and the 3D model representing the virtual world. An example is given in Fig. 1, where a low-density polygonal mesh of the surroundings created in real-time by the MR device (which represents the real world) is registered to an existing point cloud model of the same environment imported into a game engine (which represents the virtual world). Aligning the real and virtual frames performs localisation and anchors all the virtual objects as a result. One of the advantages of this approach is that there is no limit to the number of virtual objects that can be placed within the virtual world. The existing point cloud of the environment is used to place virtual objects in desired locations and a single registration anchors all of them, and their relative spatial relationships are accurate, as shown in Fig. 2. Virtual objects can be static, such as the objects shown in the figure, or dynamic, such as virtual guides with predefined movement rules. Maintaining this alignment requires continuous tracking of the pose of the MR device with respect to the real world, which can be performed accurately by most existing devices.

The proposed localisation method can be split into four main stages: (1) mapping of the environment, (2) map cropping, (3) 3D-3D model registration, and (4) world transformation. The first stage begins after the MR device is turned on when it starts to map the environment within reach of its sensors. With the origin of the virtual world as an anchor, the device creates a 3D map of the surroundings in the form of a mesh surface model, with the specific mapping method depending on the device used. The device used for the prototype, Microsoft HoloLens (1st gen), uses RGB-D Simultaneous Localisation and Mapping (SLAM) method based on four tracking cameras and a time-of-flight depth camera (Khoshelham et al., 2019; Hübner et al., 2020).

The second stage begins after a user initialises the localisation. First, the map of the surroundings is cropped so only the faces within an axis-aligned device-centred box 6x6x6 m in size remain. The size is determined by the range of the depth camera on the HoloLens device used for the prototype, which is around 3.5 m (Hübner et al., 2020). The mesh is converted into a point cloud of the surroundings by random sampling over the mesh surface. In the third stage, described in Section 3.1, this point cloud representing the real world is registered to the reference point cloud representing the virtual world. Due to the computing limits of MR devices, 3D-3D model registration is performed on a server. In the final stage, the resulting transformation matrix is applied to the mesh of the surroundings and the underlying virtual camera anchor to align the two worlds.

This initial localisation is maintained with continuous tracking which can be performed accurately by modern devices. However, the alignment will deteriorate with the accumulation of tracking errors as the device moves further away from the place of localisation. The misalignment can be fixed by subsequent re-localisation using the same method.

3.1 Coarse and fine 3D-3D model registration

The point cloud of the surroundings is registered to the reference point cloud by performing a coarse registration followed by a fine registration. The coarse registration is based on the extraction of 3D keypoints from the models by a deep neural network, and their matching with Random sample consensus (RANSAC).

As a backbone for keypoint detection and description we use *D3Feat* by Bai et al. (2020). The network is pretrained on *3DMatch* (Zeng et al., 2017) dataset fragments with overlap higher than 30% and a grid size of 3 cm, momentum optimiser with a base learning rate of 0.1, 0.98 momentum with exponential decay in each epoch and other parameters as described in (Bai et al., 2020). We make several modifications to increase generalisation and adapt the network to large point clouds with inherent differences, as they are captured with different sensors and there may be geometric differences between the real world and the virtual world, i.e. the reference model. First, prior to feature extraction, the models are downsampled with voxel grid subsampling with a grid size of 10 cm, which forces the network to work with larger rather than smaller objects to reduce the susceptibility to changes in the scene geometry. Furthermore, the receptive field of the descriptor and the scale of kernel points are modified accordingly, so that they cover the same

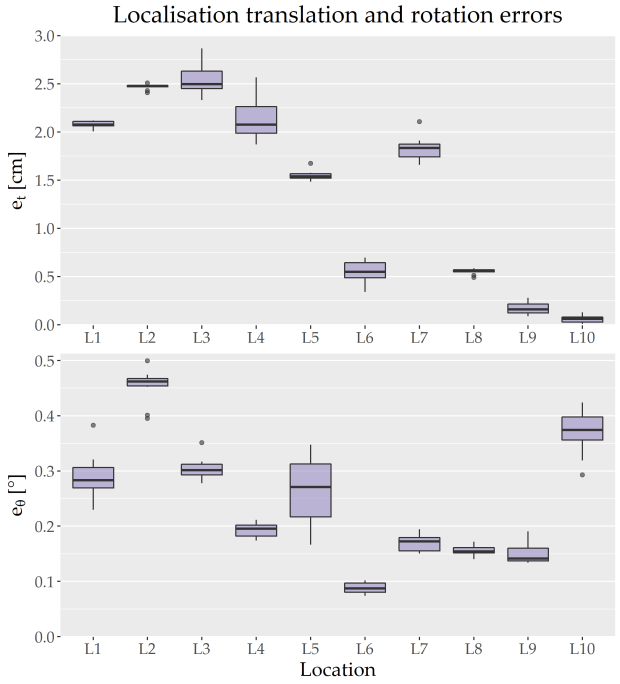


Figure 5. Localisation simulation accuracy.

size of the local patch. Finally, to secure a uniform distribution of keypoints and avoid their clustering, a hard selection of keypoints is used (Bai et al., 2020).

After the keypoint extraction comes keypoint matching with RANSAC (Fischler and Bolles, 1981) based on feature matching with the distance threshold of 5 cm, no change in scale and a maximum number of iterations of 40 million. A large number of iterations is required to identify and eliminate the outliers, which are inevitable due to the difference in the size of models being registered, the difference in scene geometry between them, and the fact that they are created with different sensors.

The resulting registration can be considered coarse as the rotation errors were experimentally found to be between 0.6 to 1.4°, which can result in an increasing misalignment between the real and the virtual world as the distance from the point of localisation increases. For this reason, RANSAC registration is refined by subsequent fine registration with ICP (Besl and McKay, 1992). The distance threshold is set to 5 cm with 30 maximum iterations.

4. EXPERIMENTS AND RESULTS

We developed a prototype to perform testing and analysis of the proposed method. The prototype has a client side and a server side. The client side is developed in Unity 2020.4 and uses a Mixed Reality Toolkit (MRTK), an open-source toolkit for the development of MR applications on a wide range of devices. The server side requires a device with a CUDA enabled GPU and a Tensorflow framework in a Linux environment. For experiments, we deploy the prototype on a Microsoft HoloLens (1st gen) and use a Linux laptop with a 6-core Intel Core i9-8950HK CPU, 16GB NVIDIA Quadro P5200 GPU and 64GB of memory.

To determine the relationship between the localisation accuracy and the size of the virtual model that can be retrieved, we first evaluate the localisation accuracy in Section 4.1. Then, in Section 4.2, we first calculate the expected success rate, i.e. the theoretical retrieval success rate based on the achieved localisation accuracy, followed by the evaluation of the empirical retrieval success rate with a field experiment.

4.1 Localisation accuracy

As the localisation is based solely on 3D-3D model registration, its accuracy can be assessed through the accuracy of registration. In turn, the accuracy of registration can be evaluated by calculating the translation and rotation errors from the differences between the true and the estimated transformation matrices. To avoid a poorly conditioned solution, before estimating the transformation the point coordinates are normalised and then the final translation and rotation parameters are obtained from the estimated transformation using the following equation (Khoshelham, 2016):

$$\mathbf{t}'_{i-j} = \mathbf{t} - \bar{\mathbf{p}}^{j} + \mathbf{R}_{i-j}\bar{\mathbf{p}}^{i}, \tag{1}$$

where $\bar{\mathbf{p}}^{\mathrm{i}}$ and $\bar{\mathbf{p}}^{\mathrm{j}}$ represent means of the source and destination models *i* and *j*, and $\mathbf{R}_{\mathrm{i\text{-}j}}$ is the rotation matrix.

Then, the translation error e_t is defined as the absolute difference between the lengths of the true and estimated translation vectors:

$$e_t = |\,||\mathbf{t'}^{t}_{i-j}|| - ||\mathbf{t'}^{e}_{j-i}||\,|. \tag{2}$$

Similarly, the rotation error is defined as the absolute difference between the true and the estimated rotation matrices, $\mathbf{R} = \mathbf{R}_{\mathrm{i\text{-}j}}{}^{\mathrm{t}} \cdot \mathbf{R}_{\mathrm{j\text{-}i}}{}^{\mathrm{e}}$. The rotation angles can be converted into one Euler angle for simplicity (Khoshelham, 2016):

$$e_\theta = cos^{-1}\big(\,(r_{11} + r_{22} + r_{33} - 1)\,/2\big), \tag{3}$$

where r_{11}, r_{22} and r_{33} are diagonal elements of $\mathbf{R}$ composed of camera rotation angles in camera pose estimation (yaw, pitch and roll angles).

As described in Section 3, the localisation is performed by registering a 6x6x6 metre box part of the mesh model of the surroundings created by the device to the existing point cloud of the environment. Thus, we evaluate the localisation accuracy by performing 100 such registrations and calculating the translation and rotation errors. We perform these simulated localisations on a testing site, which is a fully furnished 50 m^2 apartment. Within the site, 10 locations are evenly distributed and on each of them 10 registrations are performed. Fig. 3 shows the testing site and the models used. On top is the untextured polygonal mesh model created with a Microsoft HoloLens (1st gen) by making several closed loops and on the bottom is a reference point cloud surveyed with a Leica BLK360 terrestrial Light Detection and Ranging (LiDAR) device. The 10 testing locations are marked in the mesh.

As true transformation is required to calculate the translation and rotation errors, the mesh model and the reference point cloud model are aligned with the combination of a manual coarse alignment and a fine alignment with ICP. The localisation is simulated by cropping the mesh model with the 6x6x6

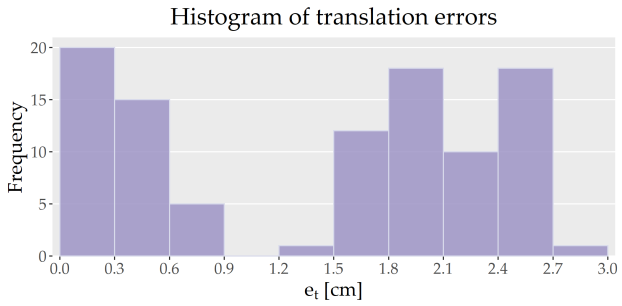


Figure 6. The empirical distribution of translation errors.

metre box centred on a location, applying a random transformation to this cropped part, and attempting an automated registration back to the reference model. The random transformation gives us the true transformation, i.e. the ground truth, which makes it possible to assess the registration accuracy from the differences between the ground truth and estimated registrations.

All 100 attempts at registration were successful and they took on average 24 seconds. The results, i.e. the calculated translation and rotation errors of localisation simulations, are shown in Fig. 5. It can be noticed that both translation and rotation errors are very consistent on each individual location with rather small spreads. However, the translation errors are less consistent between different locations, with locations 6, 8, 9 and 10 having noticeably smaller errors than the rest. This may be explained by the differences between the scene and the reference model. More precisely, roughly 6 months had passed from the capturing of the reference model, resulting in countless changes in the environment. All of the smaller furniture, such as chairs, dining table, coffee table and armchair, changed their location to some extent. Furthermore, other small- to large-sized objects, such as cushions, plants, PC screens or music instruments, may be in completely different locations or present in the reference model but absent from the scene model and vice versa. The differences in the number and the extent of these changes throughout the apartment may explain the observed difference in translation errors. Locations 6, 8, 9 and 10 are within or close to the simply furnished bedroom where the differences between the scene and reference model are less pronounced. Furthermore, the translation and rotation errors at no location exceed 3 cm and 0.5°and their averages are 1.4 cm and 0.24°, respectively. Based on these indicators, we can conclude that the overall accuracy of localisation is high.

4.2 Retrieval success rate

We assess the retrieval success rate through the probability of correct retrieval of invisible virtual elements of different sizes. We construct eight cubes with edge lengths of 60, 30, 15, 8, 4, 3, 2 and 1 cm, and mark their centre point, as shown in Fig. 4. We ensure the cubes are in the same spot in the real and in the virtual world by aligning them with a closet present in both, which provides one corner, three edges, and three sides that can be used for anchoring. In the virtual world, Fig. 4 (a), the cubes are modelled and aligned with the point cloud of the closet with a combination of manual coarse registration and ICP. In the real world, Fig. 4 (b), the cubes are constructed along the edges using the corner as an anchor and drawn on the surface of the closet.

After localisation and the alignment of the real and the virtual world, and in the absence of errors, the cubes should be perfectly aligned. Fig. 4 (c) shows this overlay of virtual cubes on top of the real cubes after localisation, although it should be noted that for evaluation the cubes are not rendered visible. To test if the cubes are where we expect them to be, after localisation we attempt to retrieve them by aiming at the centre of their real-world representations. To independently perform this test the virtual cubes must be invisible, as if visible they may obstruct the real world and influence the tester. In other words, there is little point in testing the retrieval of a visible virtual object as one can just point the cursor towards it and retrieve it, even if the localisation is poor and the object is at a certain distance from its desired predefined location.

4.2.1 Theoretical evaluation The expected retrieval probability of cubes of different sizes can be estimated from the achieved localisation accuracy, i.e. from the translation and rotation errors (Section 4.1). As the retrieval is performed a short distance from the point of localisation (within 2 m), the effect of the rotation error is expected to be minor. Hence, we use the 100 translation errors and their empirical distribution, shown in Fig. 6, to calculate the percentile of translation errors. The first, second and third quartile percentiles, as well as percentiles for the cube sizes of 1, 2 and 3 cm, are listed in Table 1. Cube sizes larger than 3 cm are not included as the 3 cm error is at the 100$^{\text{th}}$ percentile, which means all errors are below 3 cm.

Percentile	e_t [cm]
25$^{\text{th}}$ percentile	0.5
40$^{\text{th}}$ percentile	1.0
50$^{\text{th}}$ percentile	1.7
63$^{\text{th}}$ percentile	2.0
75$^{\text{th}}$ percentile	2.1
100$^{\text{th}}$ percentile	3.0

Table 1. Percentiles of localisation translation errors according to the empirical distribution.

The percentile of localisation translation errors in Table 1 are indicative, but not equivalent to the expected retrieval probability of cubes of different sizes because the retrieval is influenced by other errors, such as the rotation error or the calibration error of the HoloLens's projector and cameras. However, we hypothesise that the translation error is dominant amongst these so we use the calculated percentiles as an approximate indicator for the expected probability of retrieval.

4.2.2 Empirical evaluation To evaluate the empirical retrieval probability, we perform a localisation with the developed prototype followed by an attempt at the retrieval of an invisible virtual cube by aiming at the centre of that cube in the real world. The centre is clearly marked in the real world, aimed at with a virtual crosshair and retrieved with a gesture (so-called air tap gesture). The localisation and the retrieval are performed at a distance of 1.5-2 m from the cubes. The cubes are roughly at the same height as the user and the user looks directly at one of the cubes' vertical edges (at roughly 45° angle to 2 vertical faces). We repeat this process 10 times for each of the 8 cubes, resulting in 80 localisations and 80 retrieval attempts.

The results of this experiment are shown in Fig. 7. The first thing to notice is the consistency of results, with a steady 100% retrieval success rate for cube sizes from 60 down to 4 cm followed by a consistent drop in the success rate for 3, 2 and 1

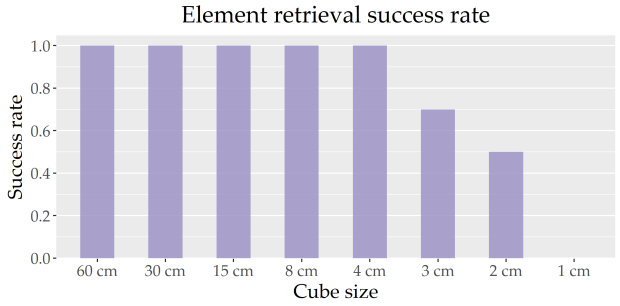


Figure 7. The empirical element retrieval success rate for elements of different sizes.

cm cube sizes. The second thing to notice is that the empirical probabilities of correct retrieval for 3, 2 and 1 cm cubes are lower than the expected probabilities (Section 4.2.1). The empirical retrieval success rate for the 3 cm cube was 70% as compared to the expected 100%, for the 2 cm cube the empirical success rate was 50% as compared to the expected 63%, while for the 1 cm cube there were no successful retrievals whereas a success rate of 40% was expected. As the theoretical values are derived solely from localisation translation errors, the discrepancy between the theoretical and the empirical values is likely caused by one or several of the additional influencing factors:

1. Localisation rotation error.
2. Aiming error, i.e. imperfect alignment of the crosshair with the middle of cubes upon retrieval.
3. Positional offset between the real and the virtual object, i.e. imperfect alignment of real and virtual cubes with the closet before localisation.
4. Calibration error of the cameras and the projector of the device.
5. The angle at which the user aims at the virtual object, i.e. the 2D projection of the virtual cube will have different offsets from its 3D centre depending on the centre of projection.

Based on these results we can conclude that the retrieval success rate closely follows the localisation accuracy (and the corresponding confidence interval) which means that other sources of error are rather negligible. Furthermore, we can conclude that the localisation method enables accurate anchoring and reliable retrieval of elements of 4 cm in size and larger. Furthermore, it enables a somewhat reliable retrieval of 3 cm large elements, with the retrieval of elements smaller than that becoming unreliable.

5. CONCLUSION AND FUTURE WORK

In this paper, we investigated the relationship between the localisation accuracy and the size of a virtual object one can retrieve after localisation. Furthermore, we presented a global MR localisation approach based only on 3D-3D model registration. By relying on 3D geometry, the proposed approach offers certain advantages over image-based localisation methods, most importantly the high resilience to scene changes such as lighting conditions or the movement of objects, as illustrated by a 100% localisation success rate in an apartment six months after the mapping of the reference model. Furthermore, it relies on a low-density untextured mesh model of the surroundings which means that there is no need to build specific maps or databases and there are no privacy issues.

The results of the localisation accuracy evaluation showed an average translation and rotation errors of 1.4 cm and 0.24°, respectively. We empirically tested the retrieval success rate and showed that one can, by aiming at a real-world object, retrieve the invisible underlying virtual element, such as a BIM element, with a 100% reliability if the object is 4 cm in size or larger. The success rate drops to 70, 50 and 0% for objects of 3, 2 and 1 cm, respectively, which is in line with expected retrieval rates influenced mainly by the translation error, but also by other influences listed in Section 4.2.

Overall, the results of our experiments suggest that MR localisation based on 3D-3D registration offers considerable potential and high accuracy. The results on the current 50 m^2 testing site indicate that the method performs well in a medium-scale environment, but larger environments should be tested in future research. Other future work directions include testing different networks, investigating a reduction of the complexity of the RANSAC search space by assuming gravity up, and adding an outlier rejection step between the feature extraction and RANSAC steps. Finally, the retrieval success rate of objects of different shapes and in different locations within the testing site should be explored.

6. ACKNOWLEDGEMENTS

This research is supported by Building 4.0 CRC. The first author acknowledges the financial support from the University of Melbourne through Melbourne Research Scholarship.

References

Acharya, D., Khoshelham, K., Winter, S., 2019a. BIM-PoseNet: Indoor Camera Localisation Using a 3D Indoor Model and Deep Learning from Synthetic Images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150, 245–258.

Acharya, D., Ramezani, M., Khoshelham, K., Winter, S., 2019b. BIM-Tracker: A Model-Based Visual Tracking Approach for Indoor Localisation Using a 3D Building Model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150, 157–171.

Baek, F., Ha, I., Kim, H., 2019. Augmented Reality System for Facility Management Using Image-Based Indoor Localization. *Automation in Construction*, 99, 18–26.

Bai, X., Luo, Z., Zhou, L., Fu, H., Quan, L., Tai, C.-L., 2020. D3Feat: Joint Learning of Dense Detection and Description of 3D Local Features. *arXiv:2003.03164 [cs]*.

Besl, P. J., McKay, N. D., 1992. Method for registration of 3-D shapes. *Sensor Fusion IV: Control Paradigms and Data Structures*, 1611, International Society for Optics and Photonics, 586–606.

Bueno, M., Bosché, F., González-Jorge, H., Martínez-Sánchez, J., Arias, P., 2018. 4-Plane Congruent Sets for Automatic Registration of as-Is 3D Point Clouds with 3D BIM Models. *Automation in Construction*, 89, 120–134.

Choy, C., Park, J., Koltun, V., 2019. Fully Convolutional Geometric Features. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, Seoul, Korea (South), 8957–8965.

Çöltekin, A., Lochhead, I., Madden, M., Christophe, S., Devaux, A., Pettit, C., Lock, O., Shukla, S., Herman, L., Stachoň, Z., Kubíček, P., Snopková, D., Bernardes, S., Hedley, N., 2020. Extended Reality in Spatial Sciences: A Review of Research Challenges and Future Directions. *ISPRS International Journal of Geo-Information*, 9(7), 439.

Fischler, M. A., Bolles, R. C., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6), 381–395.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., Bennamoun, M., 2020. Deep Learning for 3D Point Clouds: A Survey. *arXiv:1912.12033 [cs, eess]*.

Herbers, P., König, M., 2019. Indoor Localization for Augmented Reality Devices Using BIM, Point Clouds, and Template Matching. *Applied Sciences*, 9(20), 4260.

Hübner, P., Clintworth, K., Liu, Q., Weinmann, M., Wursthorn, S., 2020. Evaluation of HoloLens Tracking and Depth Sensing for Indoor Mapping Applications. *Sensors*, 20(4), 1021.

Kendall, A., Grimes, M., Cipolla, R., 2015. PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization. *2015 IEEE International Conference on Computer Vision (ICCV)*, IEEE, Santiago, Chile, 2938–2946.

Khoshelham, K., 2016. Closed-Form Solutions for Estimating a Rigid Motion from Plane Correspondences Extracted from Point Clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 114, 78–91.

Khoshelham, K., Tran, H., Acharya, D., 2019. INDOOR MAPPING EYEWEAR: GEOMETRIC EVALUATION OF SPATIAL MAPPING CAPABILITY OF HOLOLENS. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W13, 805–810.

Kim, K., Billinghurst, M., Bruder, G., Duh, H. B.-L., Welch, G. F., 2018. Revisiting Trends in Augmented Reality Research: A Review of the 2nd Decade of ISMAR (2008–2017). *IEEE Transactions on Visualization and Computer Graphics*, 24(11), 2947–2962.

Li, J., Wang, C., Kang, X., Zhao, Q., 2020a. Camera Localization for Augmented Reality and Indoor Positioning: A Vision-Based 3D Feature Database Approach. *International Journal of Digital Earth*, 13(6), 727–741.

Li, L., Zhu, S., Fu, H., Tan, P., Tai, C.-L., 2020b. End-to-End Learning Local Multi-View Descriptors for 3D Point Clouds. *arXiv:2003.05855 [cs]*.

Marchand, E., Uchiyama, H., Spindler, F., 2016. Pose Estimation for Augmented Reality: A Hands-On Survey. *IEEE Transactions on Visualization and Computer Graphics*, 22(12), 2633–2651.

Melekhov, I., Ylioinas, J., Kannala, J., Rahtu, E., 2017. Relative Camera Pose Estimation Using Convolutional Neural Networks. *arXiv:1702.01381 [cs]*.

Muthalif, M. Z. A., Shojaei, D., Khoshelham, K., 2022. A Review of Augmented Reality Visualization Methods for Subsurface Utilities. *Advanced Engineering Informatics*, 51, 101498.

Parisotto, E., Chaplot, D. S., Zhang, J., Salakhutdinov, R., 2018. Global Pose Estimation with an Attention-Based Recurrent Network. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, IEEE, Salt Lake City, UT, 350–35009.

Petit, A., Marchand, E., Kanani, K., 2012. Tracking complex targets for space rendezvous and debris removal applications. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4483–4488.

Piasco, N., Sidibé, D., Demonceaux, C., Gouet-Brunet, V., 2018. A Survey on Visual-Based Localization: On the Benefit of Heterogeneous Data. *Pattern Recognition*, 74, 90–109.

Poiesi, F., Boscaini, D., 2021. Distinctive 3D local deep descriptors. *2020 25th International Conference on Pattern Recognition (ICPR)*, IEEE, Milan, Italy, 5720–5727.

Quan, L., Lan, Z., 1999. Linear N-point Camera Pose Determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8), 774–780.

Rublee, E., Rabaud, V., Konolige, K., Bradski, G., 2011. ORB: An efficient alternative to SIFT or SURF. *2011 International Conference on Computer Vision*, Ieee, 2564–2571.

Rusinkiewicz, S., Levoy, M., 2001. Efficient variants of the ICP algorithm. *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, 145–152.

Sidani, A., Matoseiro Dinis, F., Duarte, J., Sanhudo, L., Calvetti, D., Santos Baptista, J., Poças Martins, J., Soeiro, A., 2021. Recent Tools and Techniques of BIM-Based Augmented Reality: A Systematic Review. *Journal of Building Engineering*, 42, 102500.

Walch, F., Hazirbas, C., Leal-Taixé, L., Sattler, T., Hilsenbeck, S., Cremers, D., 2017. Image-Based Localization Using LSTMs for Structured Feature Correlation. *arXiv:1611.07890 [cs]*.

Zeng, A., Song, S., Niessner, M., Fisher, M., Xiao, J., Funkhouser, T., 2017. 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Honolulu, HI, 199–208.

Zhang, J., Khoshelham, K., Khodabandeh, A., 2021. Seamless Vehicle Positioning by Lidar-GNSS Integration: Standalone and Multi-Epoch Scenarios. *Remote Sensing*, 13(22), 4525.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-V-4-2022-227-2022 | © Author(s) 2022. CC BY 4.0 License.