Pers Ubiquit Comput (2015) 19:1275–1294
DOI 10.1007/s00779-015-0892-6

ORIGINAL ARTICLE

Fast and scalable 3D cyber-physical modeling for high-precision mobile augmented reality systems

Hyojoon Bae[1] · Jules White[2] · Mani Golparvar-Fard[3] · Yao Pan[2] · Yu Sun[2]

Received: 8 May 2015 / Accepted: 18 November 2015 / Published online: 1 December 2015

Abstract Mobile augmented reality is an emerging technique which allows users to use a mobile device's camera to capture real-world imagery and view real-world physical objects and their associated cyber-information overlaid on top of imagery of them. One key challenge for mobile augmented reality is the fast and precisely localization of a user in order to determine what is visible in their camera view. Recent advances in Structure-from-Motion (SfM) enable the creation of 3D point clouds of physical objects from an unordered set of photographs taken by commodity digital cameras. The generated 3D point cloud can be used to identify the location and orientation of the camera relative to the point cloud. While this SfM-based approach provides complete pixel-accurate camera pose estimation in 3D without relying on external GPS or geomagnetic sensors, the preparation of initial 3D point cloud typically takes from hours to a day, making it difficult to use in mobile augmented reality applications. Furthermore, creating 3D cyber-information and associating it with the 3D point cloud is also a challenge of using SfM-based approach for mobile augmented reality. To overcome these challenges in 3D point cloud creation and cyber-physical content authoring, the paper presents a new SfM framework that is optimized for mobile augmented reality and rapidly generates a complete 3D point cloud of a target scene up to 28 times faster than prior approaches. Key improvements in the proposed SfM framework stem from the use of (1) state-of-the-art binary feature descriptors, (2) new filtering approach for accurate 3D modeling, (3) optimized point cloud structure for augmented reality, and (4) hardware/software parallelism. The paper also provides a new image-based 3D content authoring method designed specifically for the limited user interfaces of mobile devices. The proposed content authoring method generates 3D cyber-information from a single 2D image and automatically associates it with the 3D point cloud.

Keywords Structure-from-Motion · Image-based modeling · Mobile augmented reality

✉ Yao Pan
panyao98@gmail.com

Hyojoon Bae
hjbae@vt.edu

Jules White
julesw@vanderbilt.edu

Mani Golparvar-Fard
mgolpar@illinois.edu

[1] Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA

[2] Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, USA

[3] Department of Civil and Environmental Engineering, Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, USA

1 Introduction

Mobile augmented reality is a technique for overlaying cyber-information, such as 3D CAD models of a building, on top of real-world imagery captured with a mobile device's camera so that users can interpret their surrounding contexts at any place. For example, mobile augmented reality for construction can precisely overlay the planned 3D model of a building on top of real-world imagery of what is being constructed in order to check for deviations in the field. Two key components of mobile augmented reality are (1) localizing the user's mobile

camera in order to determine what should be visible in the photograph and where and (2) authoring and associating cyber-information with real-world 3D objects so that it can be visualized with a mobile augmented reality system. The key challenge is to deliver relevant cyber-information precisely and quickly. In order to work regardless of users' location and environment, it would be best for the localization to be performed without pre-deployed external infrastructure for location tracking.

Over the past decade, many research-related mobile augmented realities (e.g., [4, 24, 39, 44, 45]) have focused on techniques for accurate user localization, which is used to determine the user's current viewpoint and derive what real-world physical objects are currently visible in the scene and what cyber-information should be rendered over the digital camera imagery. Prior localization approaches have primarily used global positioning systems (GPS), wireless local area networks (WLAN), or indoor GPS for accurately positioning the user [10, 26, 27, 46]. The main drawback of these radio frequency (RF)-based location tracking technologies is their high degree of dependency on pre-installed infrastructure, such as GPS satellites or wireless sensors and susceptibility to noise in commodity mobile device hardware [19], which makes them either highly inaccurate or impractical to use in many cases. Some research has focused on developing infrastructure-independent location tracking systems [2, 36]. These systems are typically based on inertial measurements and make use of highly accurate accelerometers and gyroscopes attached to users. However, these sensor-based approaches suffer from accumulated drift errors, which grow with the distance traveled by the users.

To remove the dependency on pre-installed infrastructure, inertial measurers, and/or geomagnetic sensors, image-based localization has gained significant attention in the computer vision community, as well as in the augmented reality community [5, 14, 28, 43]. In addition, recent advances in Structure-from-Motion (SfM) [40] enable the creation of large-scale 3D point clouds from an unordered set of images, which can be used to localize mobile device camera imagery and provide extremely accurate augmented reality systems [6–8]. Using a 3D point cloud for user localization permits mobile augmented reality systems to estimate the 3D position and 3D orientation of the new photograph purely based on the image captured by mobile device [18, 25, 31, 38], and therefore it does not have any hardware constraints on mobile devices, such as stereo cameras, GPS sensors, or highly accurate motion tracking sensors.

However, most of the recent image-based localization methods using 3D point clouds assume that those point clouds are already available at the beginning of the localization process. The 3D point cloud generation process, also called as 3D reconstruction, is often separated from the localization process and the 3D reconstruction is done in an offline preparation step. Although there is a majority of great works on visual SLAM (simultaneous localization and mapping) technique [13, 28, 37, 42], which simultaneously constructs a sparse 3D map and localizes a device using generated map, the visual SLAM mostly focuses on small-scale environment, such as indoor office room, and suffers from inconsistent loop closure problem when the scale becomes larger, such as outdoor buildings on the street. In addition, in the context of augmented reality, the visual SLAM is difficult to associate arbitrary 3D cyber-information with physical objects as the 3D coordinates of the map are varying from the devices and their initial location of calibration. As a consequence, the SLAM also requires either an offline-learned 3D model or manual association of 3D cyber-information whenever users start SLAM with different devices.

To realize high-precision mobile augmented reality system that supports multi-scale environment as well as any kinds of commodity mobile devices with monocular camera, we also target the offline SfM-based 3D point cloud preparation for localization and cyber-information association. Despite the scalability of recent approaches in SfM [1, 15, 40], however, collecting image data and processing them to prepare a 3D point cloud still takes considerable amount of time. The Bundler [40], a widely used SfM software package, takes from hours to a day to generate a 3D point cloud even with small numbers of images. This time-consuming preparation of 3D point cloud prevents using model-based localization in mobile augmented reality, especially when users want to model a daily changing scene such as construction site.

In order to easily and rapidly prepare 3D point clouds for mobile augmented reality, a new parallelized 3D reconstruction framework optimized for mobile augmented reality is designed and verified in this paper. The proposed framework, called HD^{4}AR-SFM, makes use of (1) the combination of different binary feature descriptors, (2) filtering approach for reducing noise in the final point cloud, (3) optimized point cloud structure with 3D point descriptor for augmented reality, and (4) hardware/software parallelism. The framework is based on our previous work, Hybrid 4-Dimensional Augmented Reality (HD^{4}AR) [6–8], which utilizes 3D point clouds for fast and robust mobile augmented reality. Specifically, this paper presents a complete analysis of 3D reconstruction framework of HD^{4}AR and discusses 3D reconstruction for different use cases—from small indoor objects to large buildings at outdoor. The paper also discusses the impacts of binary descriptors used in the proposed framework on the quality and performance of 3D reconstruction. The new contributions in this paper

therefore include: (1) a new client–server architecture supported by cloud computing resources for 3D reconstruction and content authoring, (2) algorithm details of each stage—from feature extraction to incremental bundle adjustment—in the proposed framework, and (3) analysis of the resulting 3D reconstruction, e.g., memory consumption, elapsed time, mean re-projection error, and viewing direction comparison, and (4) extensive experimentations on both indoor and outdoor image data sets obtained by actual users.

Another important capability in mobile augmented reality is being able to author and associate content with the real-world physical objects around the user. Prior work has assumed that this content is already available and focused on approaches for fast and accurate user localization with 3D point clouds. Creating and associating cyber-information with physical objects on the fly, however, is challenging due to the complexity of spatially associating cyber-information with the geometry of arbitrary real-world objects, such as engine parts, in a 3D space and using a small 2D mobile device interface. Very little research has examined 3D cyber-physical information association for mobile augmented reality, which is critical in order to create applications where users can create and share cyber-information with each other through mobile augmented reality interfaces. As described by Arth et al. [5], the question of how to conveniently and accurately register even simple 3D content using a mobile device and 2D interface is still an open problem. To address this challenge, the paper provides a new image-based 3D content authoring method designed specifically for mobile augmented reality using 3D point clouds. This content authoring technique not only provides a method for creating 3D cyber-information within the confines of limited 2D mobile device user interfaces, but also provides automatic association of user-driven cyber-information with physical objects through the 3D point cloud generated from our framework.

The remainder of this paper is organized as follows: Sect. 2 discusses gaps in research related to mobile augmented reality using 3D point clouds; Sect. 3 presents the details of the proposed SfM framework; Sect. 4 discusses single-image-based 3D content authoring using 3D point clouds; Sects. 5 and 6 present empirical results from experiments with the proposed approaches; and Sect. 7 presents lessons learned and concluding remarks.

2 Related work and gaps in research

Many augmented reality applications that make use of 3D point clouds assume that target 3D point clouds and their associated 3D cyber-information already exist. Therefore, the 3D reconstruction is often done in an offline preparation step. Corresponding 3D cyber-information, such as 3D drawings of buildings, is then manually aligned to the generated 3D point clouds and these fused 3D cyber-physical models are used for augmented reality applications [16, 17]. Although many research projects have shown that 3D point clouds can be used to precisely overlay cyber-information on top of each photograph and can be used even when visual obstructions are present, they have not focused on: (1) the speed of point cloud creation; (2) the preparation of the requisite 3D cyber-physical models; or (3) the user interfaces of 3D cyber-information creation/association. If we limit our scope to SfM-based 3D reconstruction, i.e., generating a 3D point cloud from only 2D camera images without using any sensor information or geometric information, the time taken for 3D reconstruction cannot be ignored even though this step is done in an offline process [8].

2.1 Research gap 1: fast algorithmic pipelines for point cloud creation

Computer vision researchers have proposed several methods, separately from augmented reality applications, to accelerate the speed of SfM-based 3D reconstruction. First, the Bundler package has been developed by Snavely et al. [40]. Snavely et al. have created the first structured pipeline for 3D point cloud modeling from an unordered set of large-scale internet photo collections. However, the Bundler still takes from several hours to a day to generate a single 3D point cloud due to exhaustive computations in pair-wise feature matching and nonlinear multi-dimensional optimization processes on single-thread CPU. In addition, it uses the SIFT (scale-invariant feature transformation) descriptor [32] for feature extraction, which has good invariance properties but requires multiple layers of computation for each spatial scale, and thus is time-consuming. More recently, a cloud computing scheme has been introduced to accelerate the entire SfM procedure [1]. A cloud computing has achieved a remarkable performance gain on very large-scale 3D reconstruction by distributing tasks over several hundreds of cores. However, using several hundreds of cores is often not feasible and the system is still based on CPU-based SIFT descriptor. Another approach uses both GPU-based SIFT and an image clustering scheme on a cloudless system [15]. The proposed system, however, limits the number of feature points per image due to memory bandwidth of the GPU and its purpose is estimating the pose of base cameras to recover the surface of the scene rather than creating an accurate 3D point cloud for user localization or augmented reality. Finally, Strecha et al. [41] have proposed a dynamic and scalable 3D reconstruction method. The proposed method uses image meta-data, such as geo-tags, to overcome the

fragmentation and speed problems of 3D reconstruction. However, using image meta-data is not appropriate for our target applications which assume that given input images are unordered and do not have any geo-tags.

2.2 Research gap 2: validation of the speed and robustness of varying feature descriptors for 3D reconstruction

One of the key components of SfM-based 3D reconstruction is to use image feature descriptors, e.g., SIFT, to formulate the correspondence search problem as a descriptor matching and triangulate 3D geometry of each correspondence. By considering a fact that input images for 3D reconstruction are unordered and typically taken at random location, the feature descriptors used in SfM should provide consistent detection and description of image regardless of image rotation and scaling.

The most widely used image feature descriptors are vector-based real-number descriptors, such as SIFT or SURF (Speeded-Up Robust Features) [9]. In particular, SIFT is used in many recent works on 3D reconstruction [1, 15, 40] and image-based localization [25, 29, 38, 43]. Although there are many variances of SIFT, such as GPU-based or quantized SIFT, the significant memory requirements and time-consuming computation of multiple layers typically make SIFT descriptor difficult to use in fast 3D reconstruction or real-time location recognition. Therefore, many attempts have been made to achieve faster or real-time computation by replacing SIFT descriptor to other vector-based descriptors such as SURF or DAISY descriptors (e.g., [12, 31]).

On the other hand, some research projects have used binary descriptor, which consists of a binary bit-string rather than a vector of real-numbers, to reduce memory consumption and computational complexity of image processing in localization (e.g., [23]). The advantages of using binary descriptors are that (1) it requires much less memory than real-number descriptors and (2) it can use Hamming distance for descriptor matching, which is faster than Euclidian distance. However, binary descriptors are typically considered as a trade-off, providing less robustness against image rotation or scaling. While some researches have compared the robustness of binary descriptors against 2D image rotation and scaling, no research has argued the impact of binary descriptors on 3D reconstruction and compared different feature descriptors using a single unified SfM framework. Through the extensive experiments, we realize that recently proposed binary descriptors, such as BRISK (Binary Robust Invariant Scalable Keypoint) [30] or FREAK (Fast REtinA Keypoint) [3], have a strong potential for accurate 3D reconstruction. The details of these descriptors are discussed in Sect. 3.

2.3 Research gap 3: mobile cyber-physical content authoring for augmented reality applications

In terms of 3D content authoring for mobile augmented reality, a number of methods have been presented based on 3D drawing tools and manual association [2, 11, 16, 20, 24]. All of these works used existing commercial 3D drawing tools to create 3D cyber-information and manually aligned cyber-information to real-world physical objects or 3D point clouds. The main problem with this approach is that it requires specific 3D design frameworks (e.g., CAD) and tools (e.g., mouse, pen, etc.), which are not available on mobile devices. Furthermore, the 3D point clouds from SfM approach are typically sparse and it is difficult to align the 3D cyber-information to the target point clouds even with manual processes [11, 16].

A specific aim of this paper was to overcome the aforementioned challenges, speeding up overall time of 3D reconstruction and providing new algorithmic methods for creating and aligning 3D cyber-information against generated 3D point clouds. The details of each framework and method are discussed in the remainder of this paper.

3 A new parallelized SfM framework for mobile augmented reality

As described in Sects. 1 and 2, an initial 3D point cloud must be created to serve as a reference model for the model-based localization and/or mobile augmented reality. Creating this 3D point cloud requires collection of an initial set of base images of the target scene, and processing these images using the SfM algorithm estimates the 3D positions of 2D image feature points. To speed up 3D reconstruction task, new types of feature descriptors are first investigated to replace the time-consuming SIFT descriptor. As a consequence, GPU-based SURF, CPU-based BRISK, and CPU-based FREAK are comprehensively analyzed and compared within the proposed framework. A new filtering approach is also developed for accurate 3D reconstruction and the structure of point cloud is optimized for further application, such as mobile augmented reality and image-based localization. In addition, an entire framework exploits hardware/software parallelism including parallelized nearest neighbor searching to scale the performance of 3D reconstruction. The proposed SfM framework, called HD^{4}AR-SFM, follows some of the original algorithmic steps in [40], but significantly alters others in order to vastly accelerate the process and improve robustness and accuracy. As aforementioned, the key modifications that make the most substantial impact on performance are: (1) the combination of different feature detectors and descriptors to optimize the 3D reconstruction performance,

(2) new filtering approach for reducing noise in the 3D point clouds and improving localization accuracy, (3) memory-efficient point cloud structure for mobile augmented reality, and (4) a parallelized multicore CPU and GPU hardware implementation for faster processing. Figure 1 illustrates the overview of the HD^{4}AR-SFM, consisting of four algorithmic stages. The details of each algorithmic stage are further discussed in the following subsections.

3.1 Feature Detector/Extractor stage

The first stage of the HD^{4}AR-SFM is the *Feature Detector/Extractor* process, which extracts image keypoints and feature descriptors for each base image. Figure 2 shows the overall structure of the *Feature Detector/Extractor* stage. To find a set of image keypoints, a feature detection algorithm is first run on each input image. CPU-based SIFT and GPU-based SURF are implemented and used in the *Detector* module. Both SIFT and SURF are invariant to image scale and rotation and thus appropriate for 3D reconstruction from unordered photographs. However, SIFT and SURF algorithms use slightly different ways of detecting feature points. SIFT builds a set of image pyramids, filtering each layer with Difference of Gaussians (DoG) [32]. On the other hand, SURF creates a stack without downsampling for higher levels in the pyramid and it filters the stack using a box filter approximation of second-order Gaussian partial derivatives to speed up the processing time [9].

Next, the *Extractor* module extracts feature descriptors at the detected image keypoints. These extracted feature descriptors will be used as the basis for pair-wise image matching. CPU-based SIFT, GPU-based SURF, CPU-based FREAK, and CPU-based BRISK are implemented and used in this module. In contrast to SIFT and SURF, FREAK uses retinal sampling patterns to compare image intensities and produces a cascade of binary strings [3]. BRISK also assembles a bit-string descriptor from intensity comparisons retrieved by dedicated sampling of each keypoint neighborhood [30]. These resulting binary descriptors consume much less disk space compared to vector-based real-number descriptors, such as SIFT and SURF, and use Hamming distance instead of Euclidian distance for descriptor matching. After extracting feature descriptors, the pixel color information of detected keypoints is read by the *Color Reader* module. This information will be used later to assign colors to 3D points for visualization purpose. Then, all outputs are stored as binary files for faster input/output (I/O) tasks.

To investigate how feature detector and feature descriptor affect the performance and quality of 3D reconstruction, we have tested four different detector–descriptor combinations in our experiments, i.e., SIFT-SIFT, SURF-SURF, SURF-FREAK, and SURF-BRISK. To simplify the name of these combinations, we refer to them as SIFT, SURF, FREAK, and BRISK, respectively. Figure 3 shows invariant properties of each combination against 2D image rotation and scaling. From this simple test result, we can infer that all these combinations will work well for 3D reconstruction. The detailed experimental results of 3D reconstruction are presented and fully discussed in Sects. 5 and 6.

3.2 Robust matcher stage

The next step is finding correspondences between all image pairs (i.e., pair-wise matching) using extracted feature descriptors. For binary feature descriptors (FREAK and BRISK), the *FANN Matcher* module first creates hierarchical clustering trees of each image descriptors and runs the Fast Approximate Nearest Neighbors (FANN) searching algorithm [34] to rapidly find the two nearest neighbors of each descriptor in the image. For vector-based real-number descriptors (SIFT and SURF), it runs randomized k-d tree searching algorithm with four parallel trees to

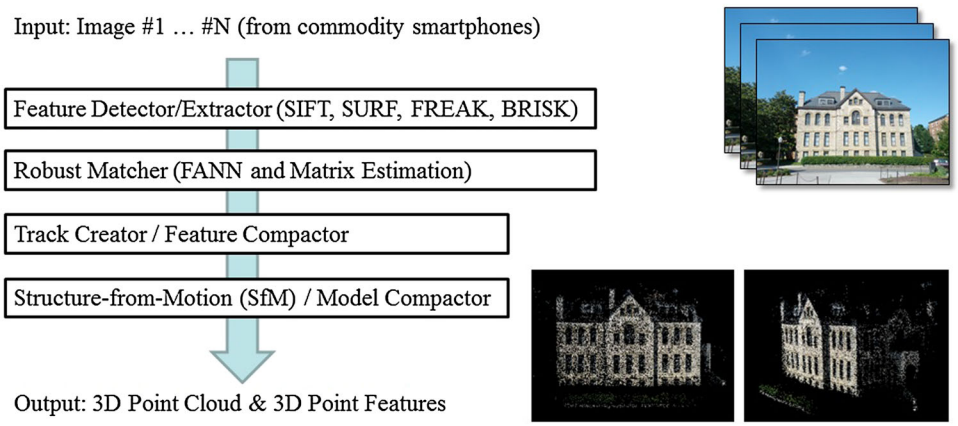


Fig. 1 A new parallelized SfM framework for mobile augmented reality

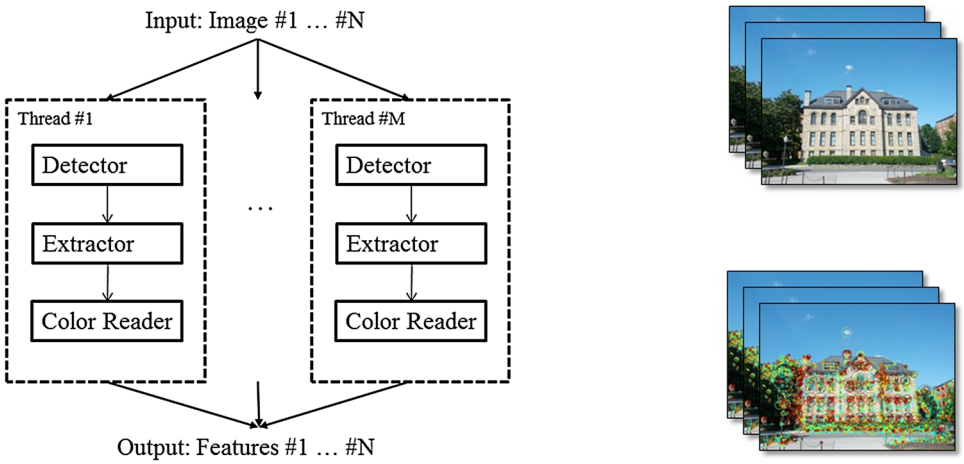


Fig. 2 Overall structure of *Feature Detector/Extractor* stage

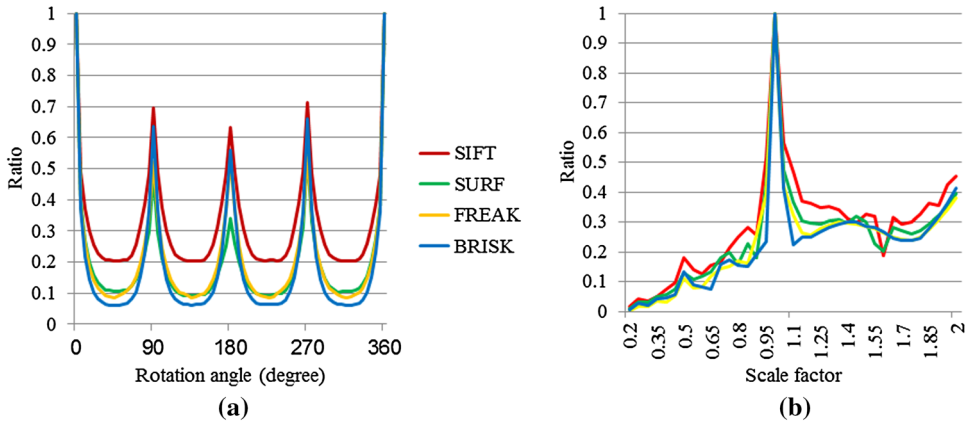


Fig. 3 Descriptor invariance test on real-world imagery. **a** Rotation invariance test, **b** scaling invariance test

improve the search speed [33]. With all recovered nearest neighbor results, the *FANN Matcher* module then performs a distance ratio-test [32] with threshold 0.5 to remove suspicious matches. In addition, if more than one feature descriptor matches the same feature in the opposite image, it removes all of the matches for that pair.

After the distance ratio-test, the *F-matrix* module robustly estimates a fundamental matrix and further removes outlier for every image pair using the RANSAC (RANdom SAmple Consensus) algorithm with the eight-point algorithm [22]. This filtering process removes false matches using an epipolar geometry constraint given by the estimated fundamental matrix. In other words, the maximum allowed distance from a keypoint to an epipolar line is σ_F pixels, beyond which the point is considered as an outlier. This outlier constraint can be expressed as:

$$||\mathbf{x}_i^T \mathbf{F}_{ij} \mathbf{x}_j|| \geq \sigma_F = \max(w_i, h_i, w_j, h_j) \cdot 0.006 \quad (1)$$

where $x_i = [u_i, v_i, 1]^T$ and $x_j = [u_j, v_j, 1]^T$ are homogenous coordinates of the matched keypoints in image i and j, respectively, F_{ij} is the estimated fundamental matrix from RANSAC iteration, and (w_i, h_i) and (w_j, h_j) are the dimension of image i and j, respectively.

Upon receiving the inliers from the *F-matrix* module, the *H-matrix* module finds a homography matrix using the RANSAC with normalized Direct Linear Transform [22] for every image pair. The outlier constraint is in the form of

$$||\mathbf{x}_i^T - \mathbf{H}_{ij} \mathbf{x}_j|| \geq \sigma_H = \max(w_i, h_i, w_j, h_j) \cdot 0.004 \quad (2)$$

where $x_i = [u_i, v_i, 1]^T$ and $x_j = [u_j, v_j, 1]^T$ are homogenous coordinates of the inliers after fitting to fundamental matrix, and H_{ij} is the estimated homography matrix from RANSAC iteration, and (w_i, h_i) and (w_j, h_j) are the dimension of image i and j, respectively. Then, the percentage of number of inliers with homography matrix, *H-score*, is calculated and recorded. The *H-score* will be used in *Structure-from-Motion* stage and image-based content authoring method to select the proper image sets.

Since the pair-wise matching is the most performance bottleneck in 3D reconstruction, each image pair is processed on different threads with lock-free parallelization, in addition to FANN searching, to shorten the overall processing time. Figure 4 shows the overall structure of the *Robust Matcher* stage. Due to FANN matching and multi-threading of the tasks, the performance of pair-wise

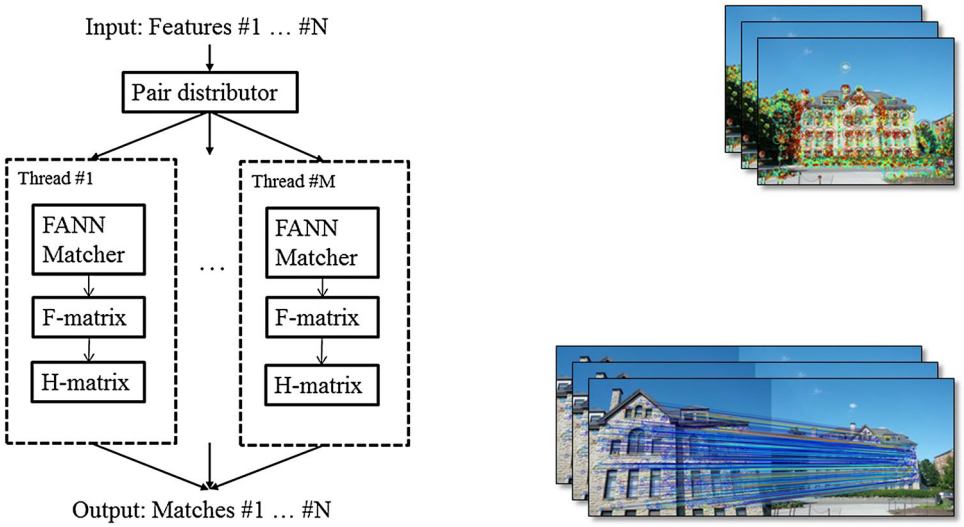


Fig. 4 Overall structure of *Robust Matcher* stage

matching is significantly improved compared to existing SfM package, e.g., the Bundler.

3.3 Track Creator/Feature Compactor stage

The *Track Creator/Feature Compactor* stage first creates tracks from matching results, where a track is a connected set of matching keypoints across multiple images. Figure 5 illustrates the overall procedures of this stage. Through extensive experiments, we found that some false matches can still survive in matching stage even after robust tests, such as distance ratio-test and fitting to the fundamental matrix, were performed. This situation is likely to happen when the target scene has repeated patterns such as multiple similar windows in the building. If these surviving false matches are organized into tracks, the SfM procedure may generate a very noisy 3D point cloud.

Therefore, we have designed and included a track ratio-test in this stage to remove false matches from each track by comparing all the matching distances of the keypoints inside the track. If one of the matching keypoints connected to a track has very high distance than others, that keypoint is erased from the track. In other words, the *Cleaner* module removes a keypoint from the track if

$$d_m/d_k < \sigma_{TR} \tag{3}$$

where d_m is the minimum matching distance among all keypoints in the track and d_k is the matching distance of each keypoint in the track. We call this procedure as a track ratio-test and the σ_{TR} is typically set to 0.3. In addition to a track ratio-test that removes the inconsistent keypoints for each track, the *Cleaner* module also removes inconsistent tracks by observing the length of each track. If the length of a track is less than σ_{TL}, which means that the track is seen by only $\sigma_{TL} - 1$ cameras, the track will not be considered in 3D reconstruction. The σ_{TL} can be set to 3–4 for very accurate 3D modeling if the input photographs were taken with specific purpose and have numerous overlapping images of target scene. However, the σ_{TL} is typically set to 2 since we target an unordered set of photographs taken at random locations.

Finally, the *Feature Compactor* module extracts and merges the feature descriptors of keypoints that are remaining in the set of consistent tracks. This process significantly reduces the disk space consumption as well as the speed of I/O task in the next stage.

3.4 Structure-from-Motion (SfM)/Model Compactor stage

The final stage of the HD^{4}AR-SFM is the *Structure-from-Motion (SfM)/Model Compactor* stage that estimates a set of camera parameters, such as focal length, rotation matrix, and translation vector, for each image and a 3D location for each track. Similar to the Bundler, the component uses an incremental approach, i.e., recovering a few cameras at a time. Once the 3D point cloud is reconstructed, the component also extracts and imposes a representative feature descriptor for each 3D point, making 3D point clouds ready for direct 2D-to-3D matching used in image-based localization. Figure 6 shows overall structure of the stage and Fig. 7 shows an example of 3D point clouds generated by our framework using real-world construction element and static building photos.

The *SfM* stage first starts with initial image pair to recover camera parameters using Nistér's five-point algorithm [35] and triangulates their feature points using polynomial method [21]. As discussed in [40], this initial pair should have a large number of matched feature points, but also have a long separation distance between the cameras to

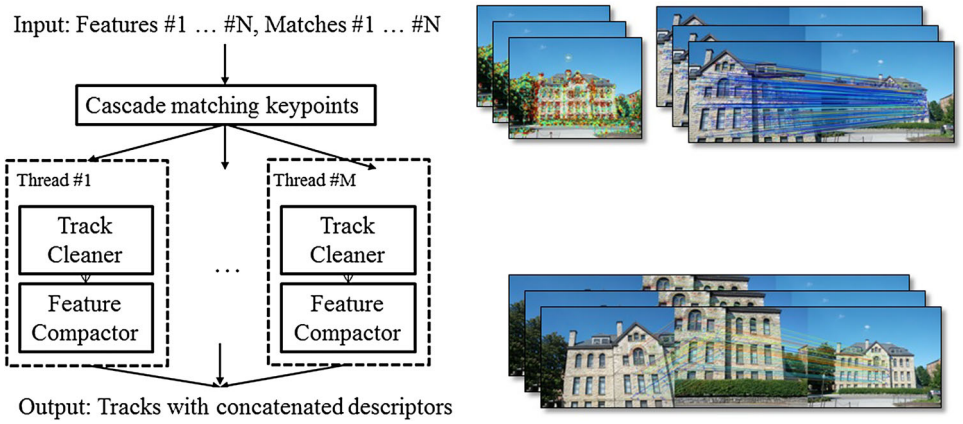


Fig. 5 Overall structure of *Track Creator/Feature Compactor* stage

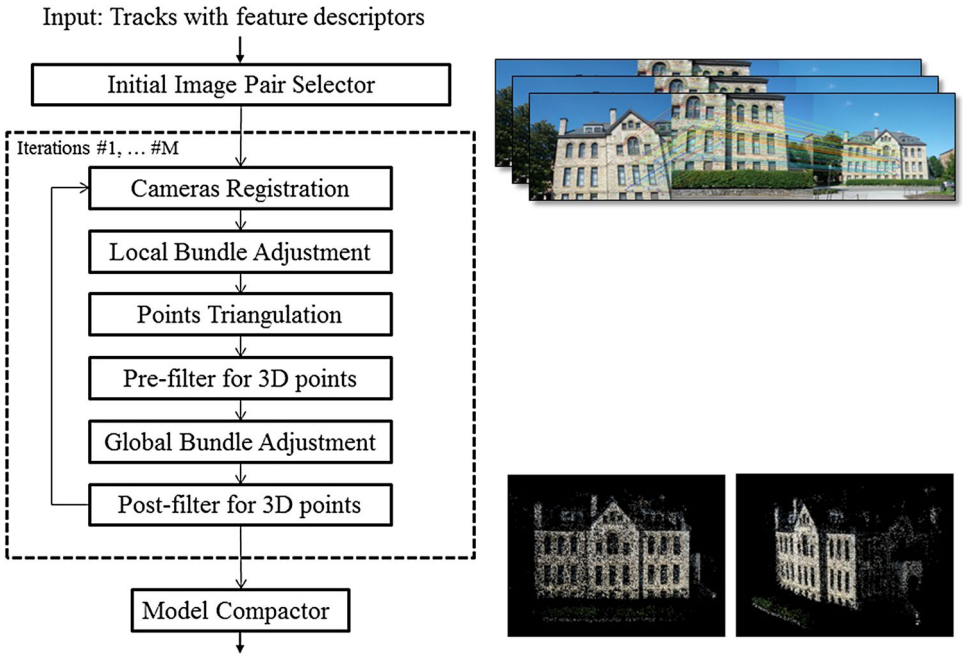


Fig. 6 Overall structure of *SfM/Model Compactor* stage

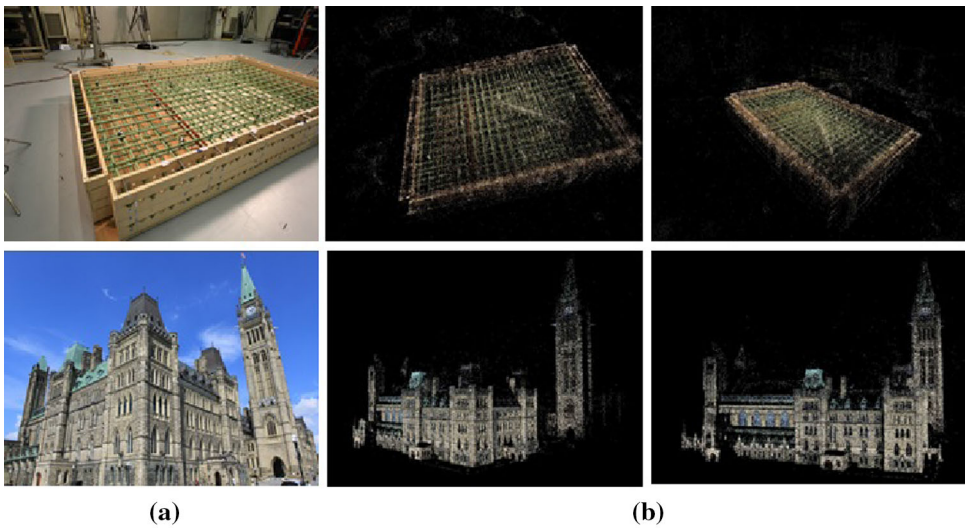


Fig. 7 Example of 3D point clouds from the proposed framework. Resulting 3D point clouds well represent the target construction element and building. **a** Initial input images, **b** 3D point clouds from the proposed framework

avoid getting stuck in a local minima during the optimization process. To fulfill this requirement, the component selects an initial image pair which has the lowest *H-Score* among all possible pairs of images. However, our experiments have shown that the *H-score* should be greater than 0.25 and the number of matches between selected pairs should be greater than 200 to generate the most accurate 3D point cloud. Therefore, these conditions are also taken into account the initial image pair selection. After calibrating the camera parameters and triangulating feature points of initial image pair, bundle adjustment optimization is run to minimize the overall mean re-projection error, i.e., the difference between predicted 2D positions of the feature points in the photographs given their triangulated 3D positions and the locations of where the feature points are actually extracted in the images. To significantly enhance the speed of this optimization, we adopt GPU-based parallel bundle adjustment approach [47].

Then, the SfM algorithm goes through iterations to calibrate camera parameters of each additional input image using the already triangulated 3D points and matching information between the images. This calibration is done using PnP (Perspective-n-Point) camera estimation method with RANSAC and Levenberg–Marquardt optimization [22]. If the component successfully recovers camera parameters of an additional base image, it registers the new camera and runs local bundle adjustment, i.e., optimizing only the newly added cameras. This camera registration fails in the event that an additional input image does not have any matched feature points against the previously registered images. After local bundle adjustment, the component triangulates the 3D points seen by the newly registered cameras and pre-filters 3D points which have high re-projection error. Through extensive experiments, we realized that this pre-filtering step is vital for accurate 3D modeling. Very little number of high-error 3D points can destroy an entire shape of 3D point cloud even with the bundle adjustment which tries to minimize overall mean re-projection error. The outlier threshold for this pre-filtering based on re-projection error is set to the same value used in the *F-matrix* module of the *Robust Matcher* stage.

Finally, global bundle adjustment is run to optimize entire 3D points currently retrieved and all parameters of currently registered cameras. During this optimization, however, it is possible that some 3D points have a high re-projection error while other 3D points have a very small re-projection error, resulting in a small mean re-projection error. The ultimate purpose of the HD^{4}AR-SFM is user localization and/or mobile augmented reality, not the visual representation of target scene, and it is very important to reduce such noise in the 3D point cloud by removing 3D points with a high re-projection error. To achieve this, the SfM algorithm in the HD^{4}AR-SFM uses a double-threshold scheme for the post-filtering stage. The first threshold is for controlling the target mean squared error (MSE) of bundle adjustment. This threshold value is set to be 0.25 pixel^2 so that the target average re-projection error of entire 3D point cloud is equal to 0.5 pixels. Another threshold, which called an absolute re-projection threshold, is for removing individual 3D points from a 3D point cloud. This threshold is adaptively calculated based on the current distribution of re-projection errors of each base image. Nevertheless, the maximum value of this threshold is set to be 4.0 pixels so that no 3D points in the final 3D point cloud have a re-projection error greater than 4.0 pixels. After post-filtering stage, if the registered camera has number of visible 3D points less than 16, that camera is removed from 3D reconstruction as it will not provide an accurate estimation of camera parameters due to small number of points. The entire SfM procedure including global bundle adjustment and post-filtering is iteratively executed until there are no more cameras to register. Due to the algorithmic enhancements and parallelization, the HD^{4}AR-SFM is up to 30 times faster than the Bundler package. In Sects. 5 and 6, experimental results of this new SfM algorithm are discussed in detail.

Once the 3D points and camera parameters of input images are successfully recovered, the *Model Compactor* module finally collects image feature descriptors for all triangulated tracks and creates a representative descriptor for each 3D point to enable direct 2D-to-3D matching. As described in [8, 38], a direct 2D-to-3D matching method has a considerable potential for fast and accurate user localization. We propose to use minimum-distance criteria, rather than averaging image descriptors proposed by Sattler et al. [38], as the framework should be able to handle binary descriptors, such as FREAK or BRISK. This procedure can be summarized as follows: for each 3D point $\mathbf{X}_n$ in the 3D point cloud,

1. Find a list of base images $(I_1, \ldots, I_k)$ and their corresponding 2D image points $(\mathbf{x}_1, \ldots, \mathbf{x}_k)$ that participated in triangulation of the 3D point during the 3D reconstruction.
2. Collect image feature descriptors $(\mathbf{d}_1, \ldots, \mathbf{d}_k)$ at discovered 2D image points $(\mathbf{x}_1, \ldots, \mathbf{x}_k)$, where each descriptor is typically a 64-dimensional (SURF, FREAK, BRISK) or 128-dimensional (SIFT, SURF) vector.
3. For each feature descriptor $(\mathbf{d}_1, \ldots, \mathbf{d}_k)$, sum Hamming (BRISK, FREAK) or Euclidean (SIFT, SURRF) distances to all other descriptors in the set.
4. Select the descriptor, which has the minimum summation value, as a representative descriptor of the 3D point.

Due to this representative descriptors approach, the localization time will depend on the number of 3D points in the point cloud, not on the number of input images used in 3D reconstruction. In addition, this approach does not only create representative descriptors of 3D points, but also provides higher probability of finding 2D-to-3D correspondences as it selects the descriptor, which has the minimum distance across all input images, as a representative descriptor for each 3D point.

4 3D cyber-physical content authoring from a single 2D image

To realize the augmented reality system with 3D point cloud, all deliverable cyber-information should have 3D positional information and be associated with given point cloud. The most straightforward method for this 3D content authoring is preparing a 3D drawing of target object or building and manually aligning it to given point cloud [17], as shown in Fig. 8. Although this approach can deliver a plenty of information to end users, it always requires manual association and a 3D drawing generated with specific 3D design frameworks, such as Computer-Aided Design (CAD) tools.

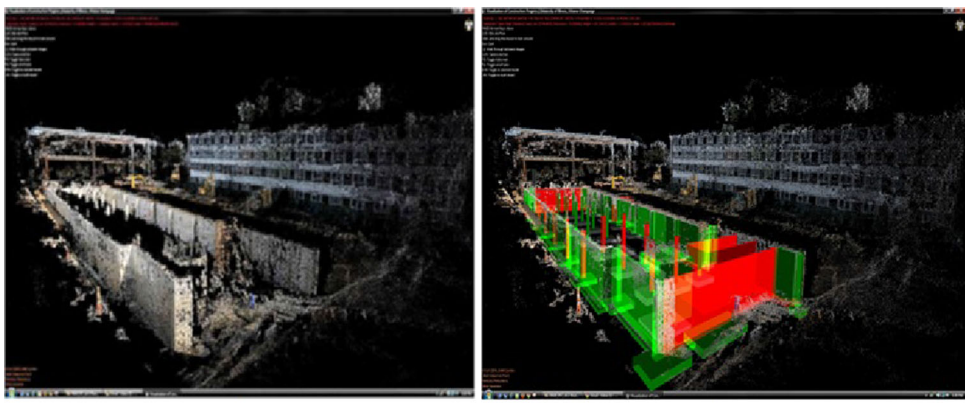
Therefore, a new approach, which can create 3D cyber-information and associate them with the point cloud using a single 2D image, is proposed in this paper. With this approach, a user can easily create and associate new 3D cyber-information by simply drawing a polygon on the photograph, and thus can work with commodity smartphones which typically have 2D user interfaces. This 3D content authoring method is based on plane image transformation, i.e., homography matrix. By its definition, the homography is an invertible transformation in a projective space that maps an image plane to another image's plane. For example, each pixel in image plane 1 can be transformed to another image plane 2 via homography matrix, as shown in Fig. 9:

Fig. 9 Example of homography transformation. Image 1 is transformed to image plane 2

$$s\begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} = \mathbf{H}_{12}\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \tag{4}$$

where $\mathbf{H}_{12}$ is an estimated homography matrix between image 1 and 2, (x_1, y_1) is a pixel coordinates in image plane 1, and (x_2, y_2) is a transformed pixel coordinates of (x_1, y_1) in the image plane 2.

Since the 3D reconstruction framework discussed in Sect. 4 keeps all estimated homography matrices between base images, we utilize these matrices to find correspondences of a user-created 2D element on each base image. For example, windows drawn by the user can be correctly found on other base images using homography matrices, as illustrated in Fig. 10a, b. To increase the accuracy of correspondences, we only investigate base images in which *H-Score* is greater than 0.85. Using this correspondence information as well as intrinsic and extrinsic camera parameters recovered during 3D reconstruction, our



(a) **(b)**

Fig. 8 Example of 3D cyber-physical model. **a** 3D point cloud of construction site, **b** 3D building plan model aligned with the point cloud (adopted from [17])

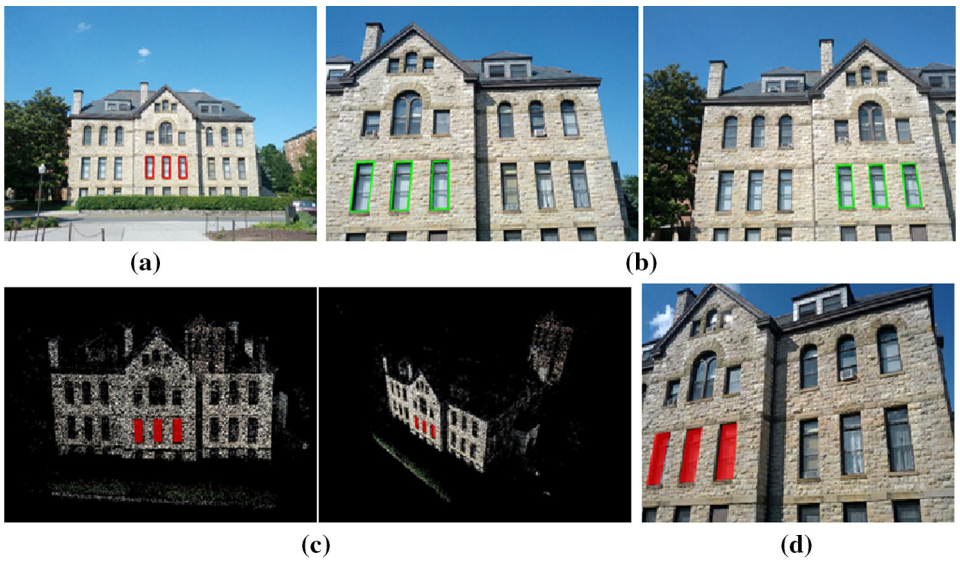


Fig. 10 Example of 3D cyber-physical content authoring. **a** A user marks windows on the photograph, **b** using the estimated homographies, the system automatically finds correspondences of windows for each base image, **c** the system triangulates window elements using camera information of base images (which is recovered during the 3D reconstruction), **d** mobile augmented reality: user-created window contents can be precisely overlaid on other photographs from different viewpoint

method then triangulates each vertex of the user-created polygon to impose 3D positional information to user-created 2D element. After fixing camera parameters and running bundle adjustment to minimize a re-projection error of the triangulated polygon, the resulting 3D element is well aligned with the existing 3D point cloud as shown in Fig. 10c. Once this user-created element has 3D positional information, it can be precisely overlaid on other photographs from different viewpoints using model-based localization, as shown in Fig. 10d.

This simple and robust 3D cyber-physical content authoring method can help users create 3D cyber-contents easily by drawing a simple polygon on a single 2D image. In addition, the approach automatically associates user-driven cyber-information with the 3D point cloud and therefore users do not have to manually position and associate 3D cyber-information in 3D geometry. Therefore, this approach can be used in any commodity smartphones which typically have a capability of showing an image on their displays and tracking user's touch points to draw the polygon. Figure 11 shows an example of 3D cyber-physical models, i.e., 3D cyber-contents with point cloud, generated from the proposed method.

5 Experimental results

This section presents experimental results of the proposed 3D reconstruction framework, i.e., HD^{4}AR-SFM, and 3D cyber-physical content authoring approaches. The details of the data set specifications and validation metrics are discussed in the following subsections.

5.1 3D reconstruction

In order to assess the improvements provided by HD^{4}AR-SFM for point cloud creation, the proposed SfM framework was compared against the Bundler package, the most widely used SfM package using incremental approach. Specifically, the speed, accuracy, and memory consumption of the Bundler package were measured and compared against HD^{4}AR-SFM to demonstrate the performance gains resulting from track compression, double-threshold filtering, parallelized matching, etc. In addition, in order to fill the key research gap in Sect. 2.2, we tested the four feature detector–descriptor combinations described in Sect. 3, i.e., SIFT, SURF, FREAK, and BRISK, to investigate the impact of feature descriptors on the performance of 3D reconstruction for mobile augmented reality.

The 3D reconstruction experiments were conducted on a single Amazon EC2 instance with 22.5 GB memory and two Intel Xeon X5570 processors running Ubuntu version 12.04. An NVIDIA Tesla M2050 graphic card was used for GPU computations. The image data sets used to create the 3D point clouds can be categorized as: (1) outdoor: existing buildings on the street and (2) indoor: car interior, kitchen, etc. Table 1 shows the summary of data sets that were used for our experimentation. These data sets were obtained from real-world mobile augmented reality photo sets provided by PAR Works Inc. An entire 3D reconstruction procedure of the HD^{4}AR-SFM was run on each data set to produce the 3D point clouds. To compare the generated 3D point clouds against those from the Bundler, the following metrics were measured:

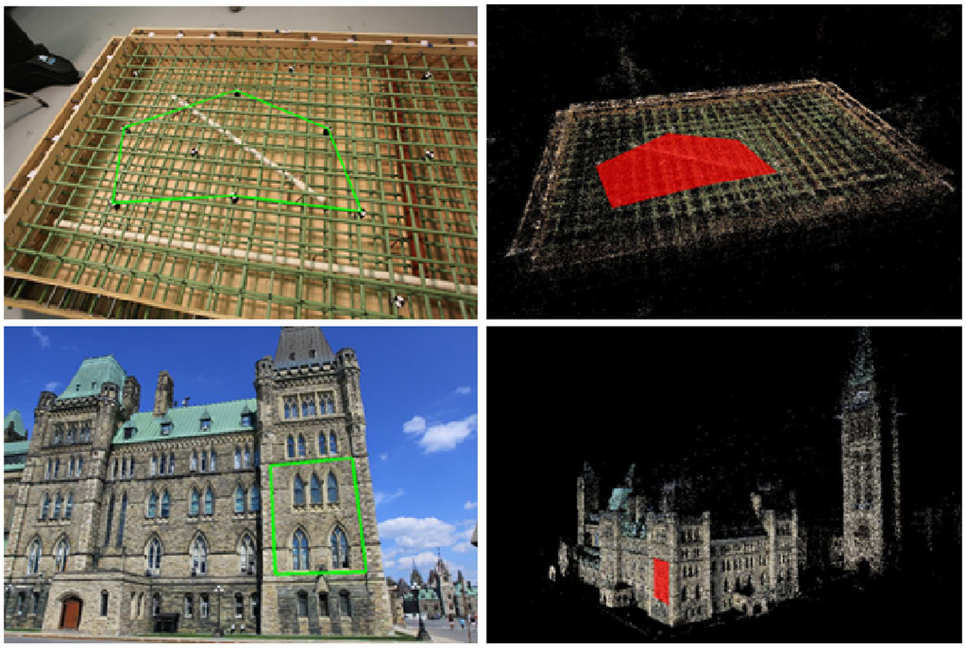


Fig. 11 Example of generated 3D cyber-physical models using the proposed method, **a** user-input on 2D image, **b** backprojected 3D cyber-information. It is well aligned to the 3D point clouds

Table 1 Data set specification for 3D reconstruction

Environment	Name	Number of images	Image resolution	Camera model
Indoor	Dashboard	27	2592 × 1944	Samsung Galaxy Nexus
	Ikea	44	3265 × 2448	Apple iPhone 4S
	Kitchen	47	2048 × 1536	Samsung Galaxy Nexus
	Knu	50	2592 × 1458	Samsung Galaxy Nexus
Outdoor	Patton	40	2592 × 1944	Samsung Galaxy Nexus
	Rh	155	2144 × 1424	Nikon D300

1. *Number of registered images* how many pre-collected photographs were calibrated. This metric measures the completeness of the 3D reconstruction process. Higher numbers of calibrated cameras will increase the reliability of the positional information of 3D points triangulated during the 3D reconstruction.
2. *Number of 3D points* how many 3D points were successfully triangulated. Larger numbers of 3D points increase the probability of direct 2D-to-3D matching and 3D localization success for mobile augmented reality.
3. *Mean re-projection error* overall mean re-projection error is computed by projecting each 3D point into each calibrated camera of the base images in order to measure the positional error of generated 3D point clouds. This metric measures the robustness and accuracy of a 3D point cloud and affects the accuracy of 3D localization for mobile augmented reality.
4. *Point cloud size* how much disk space is consumed by a single 3D point cloud. To directly use a point cloud on a mobile device, memory consumption is a key concern.
5. *Elapsed time* how long does it take to generate a single 3D point cloud. A specific aim of our framework was reducing this time in order to rapidly enable mobile augmented reality using 3D point cloud models.

Tables 2, 3, and 4 compare the overall results of 3D reconstruction on the indoor data sets. Although there are many factors that influenced the performance, such as the number of base images, the image sizes, and the texture of the target scenes, the HD^{4}AR-SFM is 661–1558 % faster than the Bundler in all indoor data sets we study. Regardless of used feature descriptors, the HD^{4}AR-SFM generated all 3D point clouds for indoor data sets within 3 min, while the Bundler took up to 25 min. For "dashboard" and "ikea" data sets, HD^{4}AR-SFM with the SIFT combination achieved the fastest point cloud creation and the SURF combination was the fastest one for "kitchen" data set. However, the SIFT combination (also used in the

Table 2 Performance comparison of 3D reconstruction for "dashboard" data set

Package	Bundler	The proposed SfM framework (HD^{4}AR-SFM)			
Detector–descriptor	SIFT	SIFT	SURF	FREAK	BRISK
Number of registered images	27/27	27/27	27/27	27/27	27/27
Number of 3D points	5210	5806	9179	7962	5962
Mean re-projection error (pixels)	0.881	0.677	0.967	0.767	0.755
Point cloud size (MB)	34.64	12.10	8.83	2.80	2.17
(memory gain)	(1×)	(2.86×)	(3.92×)	(12.37×)	(15.96×)
Elapsed time (s)	736	93.031	111.330	104.675	60.373
(performance gain)	(1×)	(7.911×)	(6.611×)	(7.031×)	(12.191×)

Table 3 Performance comparison of 3D reconstruction for "ikea" data set

Package	Bundler	The proposed SfM framework (HD^{4}AR-SFM)			
Detector–descriptor	SIFT	SIFT	SURF	FREAK	BRISK
Number of registered images	34/44	43/44	39/44	40/44	36/44
Number of 3D points	3013	7375	6350	14,868	9043
Mean re-projection error (pixels)	2.308	0.781	1.284	0.788	0.790
Point cloud size (MB)	24.69	16.30	5.98	5.35	3.37
(memory gain)	(1×)	(1.52×)	(4.13×)	(4.62×)	(7.33×)
Elapsed time (s)	1533	98.420	145.863	167.802	126.222
(performance gain)	(1×)	(15.576×)	(10.510×)	(9.136×)	(12.145×)

Table 4 Performance comparison of 3D reconstruction for "kitchen" data set

Package	Bundler	The proposed SfM framework (HD^{4}AR-SFM)			
Detector–descriptor	SIFT	SIFT	SURF	FREAK	BRISK
Number of registered images	47/47	47/47	47/47	47/47	46/47
Number of 3D points	9091	8159	11,441	8852	7517
Mean re-projection error (pixels)	1.047	0.855	1.020	0.890	0.893
Point cloud size (MB)	27.02	19.00	12.20	3.50	3.22
(memory gain)	(1×)	(1.42×)	(2.22×)	(7.72×)	(8.39×)
Elapsed time (s)	922	59.522	57.249	68.164	76.288
(performance gain)	(1×)	(15.490×)	(16.105×)	(13.526×)	(12.086×)

Bundler) produced the less number of 3D points for indoor data sets.

Next, HD^{4}AR-SFM significantly reduces the memory consumption of 3D point clouds as it only records the representative descriptors of each 3D point, while the Bundler stores all feature descriptors from the entire set of base images. In addition, the Bundler only uses the SIFT descriptor, which is 128-dimensional real-number vector, so it consumes a lot of disk space to store information related to 3D point clouds for localization (called registration in the Bundler) and mobile augmented reality. Memory consumption is important when multiple mobile clients perform online localization simultaneously with different 3D physical models. Large file sizes prevent from pre-loading multiple point clouds into memory and reduce server-side localization speed due to increased disk I/O and memory swapping. Large file sizes prevent from mobile clients from pre-loading multiple point clouds into memory and reduce server-side localization speed due to increased disk I/O. In our experience, file I/O for reading 3D point could for localization takes about 6 s when the 3D point cloud size excesses 300 MB, and it is about 70 % of the entire model-based localization process if the server does not cache the point cloud in the memory.

Finally, the mean re-projection errors show that the HD^{4}AR-SFM generates more accurate 3D point clouds for the indoor data sets. The HD^{4}AR-SFM achieved mean re-projection errors less than 1.3 pixels for all cases, while the Bundler resulted up to 2.3 pixels of error. The mean re-projection error represents how accurate the resulting 3D point cloud and the calibrated camera parameters are, as the re-projection error is calculated by projecting each 3D

point into each calibrated camera of the base images and computing the distance to the position of original image feature point. The results illustrate that the generated 3D point clouds with HD^{4}AR-SFM have only 1-pixel mean re-projection error and well represent the target scenes. Therefore, we can conclude that the generated 3D point clouds can be indeed used for accurate model-based image localization and the mobile augmented reality applications targeting 3D localization. Figure 12 shows the generated 3D point clouds from all data sets using BRISK combination.

Tables 5, 6, and 7 compare the overall results on outdoor data set and Fig. 13 shows the generated 3D point clouds using BRISK combination. Again, the HD^{4}AR-SFM outperformed the Bundler and is 304–2875 % faster for outdoor data sets. In addition, the HD^{4}AR-SFM achieved the memory gain up to 2759 % and all generated 3D point clouds have mean re-projection error smaller than 0.703 pixels. While binary descriptors, i.e., FREAK and BRISK, achieved a huge gain on both reconstruction speed and memory consumption on the outdoor data sets, they produced little less dense 3D point clouds. The outdoor images typically have a plenty of textures, and therefore, the invariance properties of feature descriptors shown in Fig. 3 affect the number of true matches between photographs taken at random location and orientation. A key question is whether or not the reduction in point cloud density impacts mobile client localization. Based on visual analysis of the point clouds, we believe that the reduced density of the 3D point clouds would not affect model-based 6-DOF localization since all 3D point clouds well represent the target scene, as shown in Fig. 13. Rather, the smaller number of 3D points accelerates the direct 2D-to-3D matching by focusing on the most significant feature points and therefore improves localization speed.

In order to present the accuracy of camera calibration, we also compared the recovered camera parameters of the "patton" 3D point cloud to the reference camera positions which were manually measured. The reference positions and orientations were derived by manually selecting 2D-to-3D correspondences and computing the rotational matrix and translation vector from those correspondences. Although this approach does not represent the ground truth comparison, it is useful to demonstrate the accuracy of 3D reconstruction and calibrated camera parameters. Furthermore, we aligned camera center positions of the 3D point cloud to the geotags of photographs using RANSAC algorithm to derive real-world distance. Table 8 summarizes results of the comparison, including viewing direction and distances of corresponding cameras. For the camera orientation, the mean angle difference of viewing direction is within 0.45° and the SIFT combination produced the best estimates of viewing direction. For the camera position, it is important to note that the geo-information was measured by noisy GPS sensor installed in Galaxy Nexus smartphone, and therefore the translational difference in meter is not highly credential. Nevertheless, HD^{4}AR-SFM calibrates the camera position within 1 meter, except the BRISK combination. Compared to the Bundler, we can

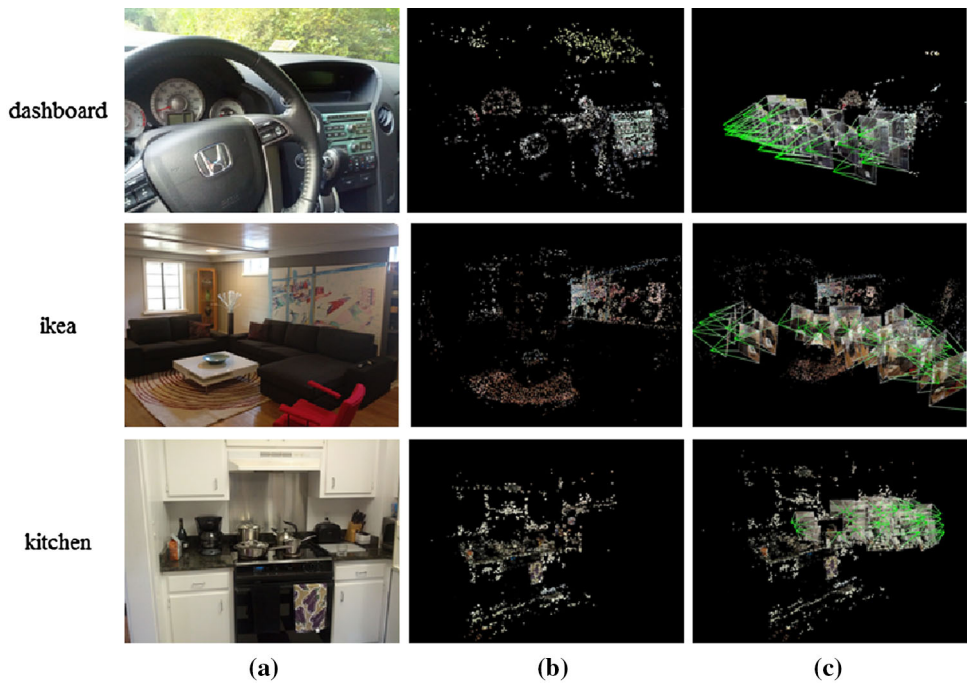


Fig. 12 3D reconstruction results with BRISK combination (SURF-BRISK). **a** Initial input images (indoor), **b** 3D point clouds from the proposed framework, **c** 3D point clouds with estimated camera positions of input images

Table 5 Performance comparison of 3D reconstruction for "knu" data set

Package	Bundler	The proposed SfM framework (HD^{4}AR-SFM)			
Detector–descriptor	SIFT	SIFT	SURF	FREAK	BRISK
Number of registered images	50/50	49/50	50/50	49/50	49/50
Number of 3D points	37,356	51,730	40,858	32,827	33,122
Mean re-projection error (pixels)	0.681	0.504	0.673	0.595	0.552
Point cloud size (MB)	223.16	104.00	41.38	12.02	11.97
(memory gain)	(1×)	(2.15×)	(5.39×)	(18.57×)	(18.64×)
Elapsed time (s)	4424	469.687	314.944	321.040	378.303
(performance gain)	(1×)	(9.419×)	(14.047×)	(13.78×)	(11.694×)

Table 6 Performance comparison of 3D reconstruction for "patton" data set

Package	Bundler	The proposed SfM framework (HD^{4}AR-SFM)			
Detector–descriptor	SIFT	SIFT	SURF	FREAK	BRISK
Number of registered images	40/40	40/40	40/40	40/40	40/40
Number of 3D points	129,693	147,798	72,000	47,163	46,318
Mean re-projection error (pixels)	0.661	0.578	0.596	0.502	0.498
Point cloud size (MB)	446.90	331.00	72.80	16.30	16.20
(memory gain)	(1×)	(1.35×)	(6.14×)	(27.42×)	(27.59×)
Elapsed time (s)	8571	2824.424	923.932	300.358	298.095
(performance gain)	(1×)	(3.035×)	(9.277×)	(28.536×)	(28.753×)

Table 7 Performance comparison of 3D reconstruction for "rh" data set

Package	Bundler	The proposed SfM framework (HD^{4}AR-SFM)			
Detector–descriptor	SIFT	SIFT	SURF	FREAK	BRISK
Number of registered images	155/155	155/155	155/155	149/155	151/155
Number of 3D points	59,533	27,247	36,854	31,738	41,097
Mean re-projection error (pixels)	0.818	0.603	0.703	0.567	0.600
Point cloud size (MB)	247.08	60.00	38.80	14.20	18.10
(memory gain)	(1×)	(4.12×)	(6.37×)	(17.40×)	(13.65×)
Elapsed time (s)	16,070	980.450	2474.513	1329.698	1371.612
(performance gain)	(1×)	(×16.390)	(6.494×)	(12.085×)	(11.716×)

conclude that HD^{4}AR-SFM is also accurately calibrated cameras during the 3D reconstruction, in terms of viewing direction and position, even with significant performance and memory gains.

5.2 3D cyber-physical content authoring from a single 2D image

As described in Sect. 4, we developed a plane transformation-based 3D cyber-physical content authoring method using a single 2D image. Since it is impractical to measure the ground truth position of every objects on the 3D point cloud, which often consists of sparse 3D points, we focused on demonstrating the capability of generating 3D cyber-information from 2D interface using commodity smartphones and empirically made a decision whether cyber-information is accurately associated with physical objects or not. In addition to visual analysis, however, we also measured the mean re-projection error of triangulated 3D driven element with base images that were participated in back-projection. The experimentation for 3D cyber-physical content authoring is performed in following procedure: (1) let users draw polygons on interesting objects on the single image with smartphones, (2) perform the proposed content authoring method and visualize generated 3D cyber-information with 3D point cloud to see the accuracy of 3D cyber-information triangulation and back-projection, and (3) test localization/augmentation on different location and viewpoint to verify that created cyber-contents are indeed well associated in 3D geometry. The test tool for augmentation was based on our previous work on HD^{4}AR [6–8].

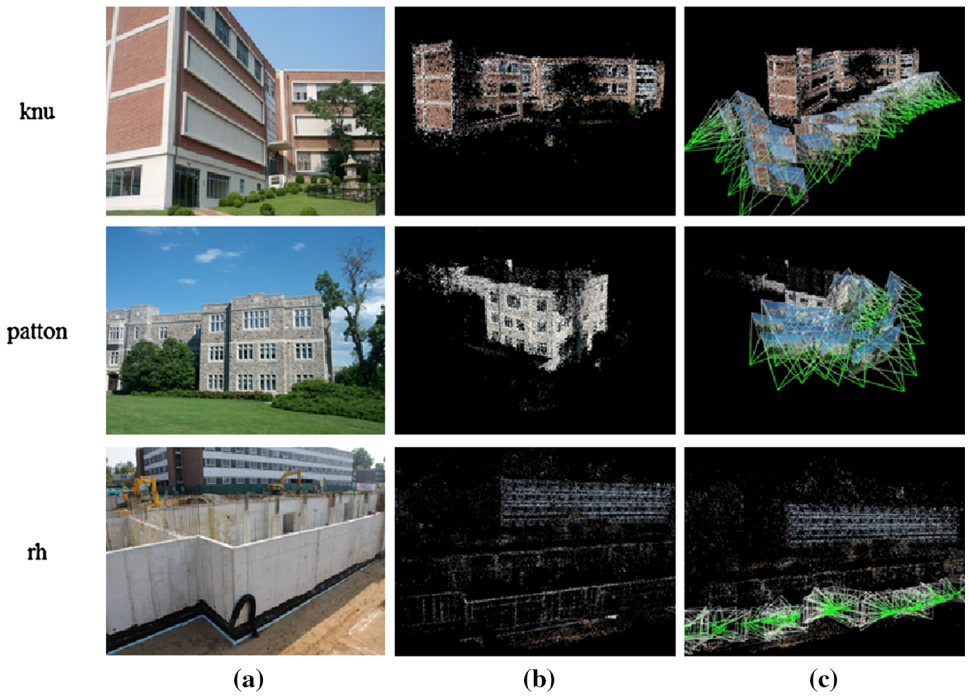


Fig. 13 3D reconstruction results with BRISK combination (SURF-BRISK). **a** Initial input images (outdoor), **b** 3D point clouds from the proposed framework, **c** 3D point clouds with estimated camera positions of input images

Table 8 Mean differences between camera pose estimates in "patton" 3D point clouds and the manually measured references

Package	Bundler	The proposed SfM framework (HD^{4}AR-SFM)			
Detector–descriptor	SIFT	SIFT	SURF	FREAK	BRISK
Rotational difference	0.398°	0.410°	0.449°	0.419°	0.426°
Translational difference (in 3D coordinates[a])	0.0277	0.0115	0.0258	0.0362	0.016
Translational difference (m[b])	8.242	0.941	0.194	0.214	9.878

[a] Each point cloud has its own 3D coordinates formed during the SfM

[b] Aligned to geotags measured by noisy sensors in Galaxy Nexus

Table 9 and Figs. 14 and 15 show the results of 3D cyber-physical content authoring with the proposed method. When selecting the base image to find correspondence using homographies, we only use the images in which *H-Score* is greater than 0.85. In all cases, the proposed method successfully generated 3D contents from user inputs on a single 2D image. For example, a user drew several polygons on the dashboard buttons for "dashboard" image and the proposed method precisely triangulated and associated them with corresponding objects in the "dashboard" 3D point cloud, as shown in Fig. 14. Similarly, user-driven windows on patton (outdoor) image were successfully associated with windows in the patton 3D point cloud, as shown in Fig. 15. Once these user-driven elements were successfully attached and aligned to 3D point clouds, users can see this cyber-information precisely overlaid on the photograph taken from different locations (see Figs. 14c, 15c). Based on experimental results, we can conclude that the proposed method purely creates 3D cyber-information using user inputs from a single 2D image. By using plane transformation to automatically find correspondences of user-driven elements and triangulating all of those correspondences against the 3D point cloud, the proposed method automatically associates user-driven cyber-information with corresponding physical objects in 3D geometry. As a result, users do not require any priori knowledge of the coordinates of point clouds and the generated 3D cyber-physical model from our method can be used in mobile augmented reality to deliver any arbitrary cyber-information created by users using a single 2D image. In addition, the proposed method can be used with any mobile devices if mobile devices have a capability of showing an image on the screen.

6 Discussion on results

Based on experimental results discussed in the previous section, we illustrate the potential of the HD^{4}AR-SFM for rapidly creating 3D point clouds from real-world data sets.

Table 9 3D cyber-physical content authoring results with 3D point clouds generated by the BRISK combination

Environment	Name	Number of vertices for user-driven elements	Number of base images participated in triangulation	Mean re-projection error (pixels)
Indoor	dashboard	20	4	0.432
	Ikea	4	3	0.686
	Kitchen	4	2	0.205
	Knu	4	4	0.268
Outdoor	Patton	15	8	2.619
	Rh	4	5	0.914

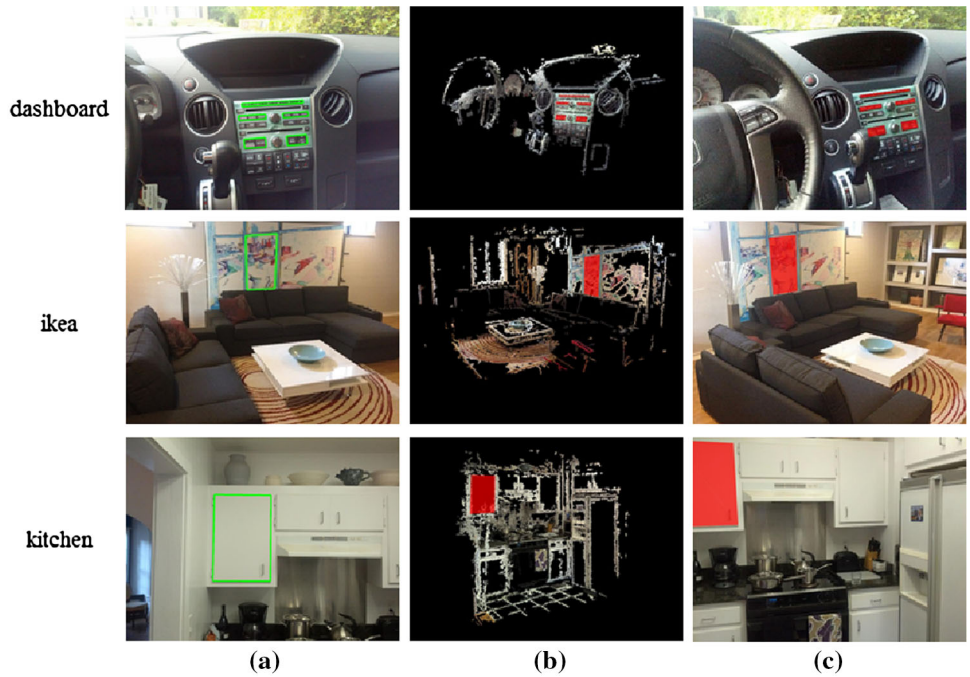


Fig. 14 Results of 3D cyber-physical content authoring with the proposed method on indoor data sets. **a** User-created information on the 2D image, **b** 3D elements driven from the user-created 2D elements and correspondences found by homography. Here the dense 3D point cloud is used for visualization purpose, and **c** augmentation results of the user-created elements on another smart device on the site

Due to enhancements presented in this paper, such as combination of binary feature descriptor, post-filtering during the SfM, and hardware/software parallelism, the HD^{4}AR-SFM took at most 3 min to generate a 3D point cloud for indoor images. Compared to the Bundler, the most widely used SfM package with incremental bundle adjustment, the HD^{4}AR-SFM achieved the performance gain up to 2875 %. By considering all the results shown in Tables 2, 3, 4, 5, 6, and 7, we can conclude that the HD^{4}AR-SFM works well with both indoor and outdoor data sets and achieves significant gains on both speed and accuracy. The binary feature descriptors, such as FREAK and BRISK, are appropriate for fast 3D reconstruction and still generate accurate 3D point clouds with less memory consumption. Furthermore, the HD^{4}AR-SFM successfully generates 3D point clouds purely based on images and does not require any constraints on photographs, such as geo-tag and ordered sequence. In all cases, the maximum re-projection error is few image pixels, and therefore, generated 3D point clouds well represent target scene and can be used for mobile augmented reality.

In addition, the proposed homography-based 3D content authoring method purely creates 3D cyber-contents using user inputs from a single 2D image. By automatically finding correspondences of user-driven elements and triangulating all of those correspondences against the 3D point cloud, the proposed method supports from 3D content generation to automatic association of generated cyber-contents (e.g., product manual, history, website) using commodity mobile devices. As discussed in Sect. 5.2, the cyber-information generated by the proposed method can be precisely overlaid on other mobile devices at different location through model-based localization. The interface of the proposed method only requires a capability of drawing a polygon on the image and thus is intuitive and straightforward. The convenient method for cyber-contents creation is especially important when designing the mobile augmented reality applications, as the 3D cyber-physical

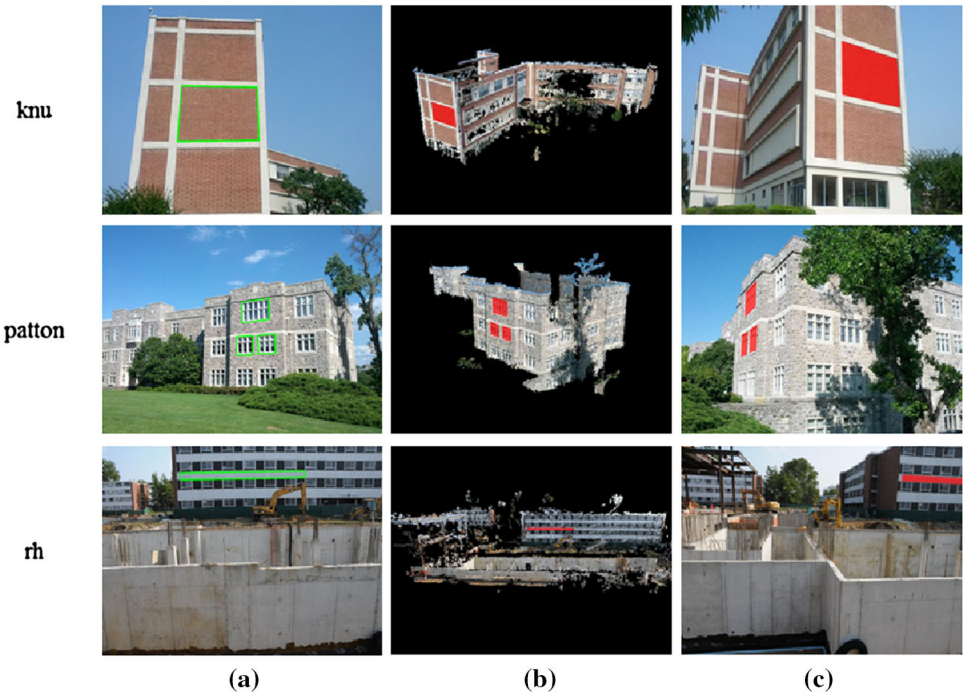


Fig. 15 Results of 3D cyber-physical content authoring with the proposed method on outdoor data sets. **a** User-created information on the 2D image, **b** 3D elements driven from the user-created 2D elements and correspondences found by homography. Here the dense 3D point cloud is used for visualization purpose, and **c** augmentation results of the user-created elements on another smart device on the site

information association is critical in order to create applications where users can create and share cyber-information with each other through mobile augmented reality interfaces.

While this paper presented the extensive experimental results with remarkable performance gain on 3D reconstruction as well as 3D content authoring, several challenges remain. Some of the open research problems that we will address in our future work include:

- Develop a metric which can guide user to take minimal number of images for accurate 3D reconstruction.
- Cluster the 3D point cloud using supplemental information such as mobile GPS information available in smartphones to further speed up 3D reconstruction.
- Analyze actual use cases of our framework on mobile augmented reality applications.
- Develop a scheme for mobile augmented reality that works with multiple 3D point clouds on the system. Most of SfM-based augmented reality assumes that users and systems know which 3D point cloud should be used for localization and augmentation, which is not practical in many cases.

7 Conclusion

In this paper, a new Structure-from-Motion (SfM)-based 3D cyber-physical modeling for mobile augmented reality is proposed and developed. By introducing HD^{4}AR-SFM, a new parallelized SfM framework which accelerates an existing 3D reconstruction pipeline by a factor of 28, we make model-based localization feasible in mobile augmented reality which provides much shorter point cloud preparation time compared to existing work. To speed up the 3D reconstruction, our framework has used four approaches: (1) the combination of several state-of-the-art feature detectors and feature descriptors including binary descriptors, (2) new filtering procedure on both track creation and SfM to reduce ambiguous matches and the noise of a final 3D point cloud, (3) extracting representative 3D descriptors to optimize the memory consumption, and (4) a scheme for use of multi-core CPU and GPU. We have also demonstrated that binary feature descriptors are suitable for fast 3D reconstruction and still generate accurate 3D point clouds with much less memory consumption compared to real-number descriptors, such as SIFT or SURF.

Along with the proposed 3D reconstruction framework, a new plane transformation-based 3D cyber-physical content authoring approach is proposed and validated. The proposed approach purely creates 3D cyber-information using user inputs from a single 2D image and automatically associates user-driven cyber-information with corresponding physical objects in 3D geometry. Validation results show that all user-driven elements on 2D images can be accurately triangulated and associated with objects in 3D point cloud and generated 3D cyber-information can be precisely overlaid on other photographs taken at completely different locations. By considering a fact that the 3D content authoring from 2D interface is still an open

problem, the proposed approach can address the open problem and make 3D cyber-physical content authoring feasible on any mobile devices. In our future work, we will study the applicability of the 3D cyber-physical models generated by the proposed method on mobile augmented reality and address the open research challenges discussed in this paper, such as real-time model-based localization, by developing point cloud clustering or cached matching scheme, etc.

References

1. Agarwal S, Snavely N, Simon I, Seitz S, Szeliski R (2009) Building rome in a day. In: 2009 IEEE 12th international conference on computer vision, pp 72–79. IEEE
2. Akula M, Dong S, Kamat V, Ojeda L, Borrell A, Borenstein J (2011) Integration of infrastructure based positioning systems and inertial navigation for ubiquitous context-aware engineering applications. Autom Constr 25(4):640–655
3. Alahi A, Ortiz R, Vandergheynst P (2012) Freak: fast retina keypoint. In: Proceeding of the 2012 IEEE conference on computer vision and pattern recognition (CVPR), pp 510–517
4. Allen M, Regenbrecht H, Abbott M (2010) Smart-phone augmented reality for public participation in urban planning. In: Proceeding of the 23rd Australian computer-human interaction conference, pp 11–20
5. Arth C, Schmalstieg D (2011) Challenges of large-scale augmented reality on smartphones. In: Proceeding of the 10th international symposium on mixed and augmented reality (ISMAR), vol 1
6. Bae H, Golparvar-Fard M, White J (2011) Enhanced HD^{4}AR (hybrid 4-dimensional augmented reality) for ubiquitous context-aware aec/fm applications. In: Proceeding of the 12th international conference on construction applications of virtual reality (CONVR), pp 253–262
7. Bae H, Golparvar-Fard M, White J (2013) High-precision and infrastructure-independent mobile augmented reality system for context-aware construction and facility management applications. In: Proceeding of the 2013 ASCE international workshop on computing in civil engineering (IWCCE), pp 637–644
8. Bae H, Golparvar-Fard M, White J (2013) High-precision vision-based mobile augmented reality system for context-aware architectural, engineering, construction and facility management (aec/fm) applications. Vis Eng 1(3):1–13. doi:10.1186/2213-7459-1-3
9. Bay H, Ess A, Tuytelaars T, Gool L (2008) Speeded-up robust features (surf). Comput Vis Image Underst 110(3):346–359
10. Behzadan A, Kamat V (2007) Georeferenced registration of construction graphics in mobile outdoor augmented reality. J Comput Civil Eng 21(4):247–258
11. Carozza L, Tingdahl D, Bosché F, Van Gool L (2014) Markerless vision-based augmented reality for urban planning. J Comput Aided Civil Infrastruct Eng. doi:10.1111/j.1467-8667.2012.00798.x
12. Chen W, Xiong Y, Gao J, Gelfand N, Grzeszczuk R (2007) Efficient extraction of robust image features on mobile devices. In: Proceeding of the 6th IEEE and ACM international symposium on mixed and augmented reality (ISMAR), pp 1–2
13. Davison A, Reid I, Molton N, Stasse O (2007) Monoslam: real-time single camera slam. IEEE Trans Pattern Anal Mach Intell 29(6):1052–1067
14. Dong Z, Zhang G, Jia J, Bao H (2009) Keyframe-based real-time camera tracking. In: Proceeding of the 12th IEEE international conference on computer vision (ICCV), pp 1538–1545
15. Frahm J, Fite-Georgel P, Gallup D, Johnson T, Raguram R, Wu C, Jen Y, Dunn E, Clipp B, Lazebnik S, Pollefeys M (2010) Building Rome on a cloudless day. In: Proceedings of the 11th European conference on Computer vision (ECCV 2010). Springer, Berlin
16. Golparvar-Fard M, Peña Mora F, Savarese S (2011) Integrated sequential as-built and as-planned representation with d^{4}ar tools in support of decision-making tasks in the aec/fm industry. J Constr Eng Manag 137(12):1099–1116
17. Golparvar-Fard M, Peña-Mora F, Savarese S (2015) Automated progress monitoring using unordered daily construction photographs and IFC-based building information models. J Comput Civil Eng. doi:10.1061/(ASCE)CP.1943-5487.0000205
18. Gordon I, Lowe D (2006) Toward category-level object recognition. Springer, Berlin
19. Gotow J, Zienkiewicz K, White J, Schmidt D (2010) Mobile wireless middleware, operating systems, and applications. Springer, Berlin Heidelberg
20. Hakkarainen M, Woodward C, Billinghurst M (2008) Augmented assembly using a mobile phone. In: Proceeding of 7th IEEE/ACM international symposium on mixed and augmented reality (ISMAR 2008), pp 167–168
21. Hartley R, Sturm P (1997) Triangulation. Comput Vis Image Underst 68(2):146–157
22. Hartley R, Zisserman A (2004) Multiple view geometry in computer vision. Cambridge University Press, Cambridge
23. Hartmann J, Forouher D, Litza M, Klssendorff JH, Maehle E (2012) Real-time visual slam using fastslam and the microsoft kinect camera. In: Proceeding of the 7th German conference on robotics (ROBOTIK), pp 1–6
24. Irizarry J, Gheisari M, Williams G, Walker B (2013) Infospot: a mobile augmented reality method for accessing building information through a situation awareness approach. Autom Constr 33:1–6
25. Irschara A, Zach C, Frahm J, Bischof H (2009) From structure-from-motion point clouds to fast location recognition. In: Proceeding of the IEEE conference on computer vision and pattern recognition (CVPR), pp 2599–2606
26. Izkara JL, Pérez J, Basogain X, Borro D (2007) Mobile augmented reality, an advanced tool for the construction sector. In: Proceedings of the 24th W78 conference, Maribor, Slovenia, pp 190–202. Citeseer
27. Khoury H, Kamat V (2009) High-precision identification of contextual information in location-aware engineering applications. Adv Eng Inform 23(4):483–496
28. Klein G, Murray D (2007) Parallel tracking and mapping for small ar workspaces. In: Proceeding of the 6th IEEE and ACM international symposium on mixed and augmented reality (ISMAR), pp 225–234
29. Lee T, Höllerer T (2008) Hybrid feature tracking and user interaction for markerless augmented reality. In: Proceeding of the IEEE virtual reality conference (VR), pp 145–152
30. Leutenegger S, Chli M, Siegwart R (2011) Brisk: binary robust invariant scalable keypoints. In: Proceeding of the 13th IEEE international conference on computer vision (ICCV), pp 2548–2555
31. Lim H, Sinha S, Cohen M, Uyttendaele M (2012) Real-time image-based 6-dof localization in large-scale environments. In: Proceeding of the 2012 IEEE conference on computer vision and pattern recognition (CVPR), pp 1043–1050
32. Lowe D (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110

33. Muja M, Lowe D (2009) Fast approximate nearest neighbors with automatic algorithm configuration. In: Proceeding of the international conference on computer vision theory and applications (VISAPP), pp 331–340
34. Muja M, Lowe D (2012) Fast matching of binary features. In: Proceeding of the 9th IEEE conference on computer and robot vision (CRV), pp 404–410
35. Nistér D (2004) An efficient solution to the five-point relative pose problem. IEEE Trans Pattern Anal Mach Intell 26(6):756–777
36. Ojeda L, Borenstein J (2007) Personal dead-reckoning system for gps-denied environments. In: Proceeding of the 2007 IEEE international workshop on safety, security and rescue robotics (SSRR): 27–29 September; Rome, Italy, pp 1–6
37. Salas-Moreno R, Newcombe R, Strasdat H, Kelly P, Davison A (2013) Slam++: simultaneous localisation and mapping at the level of objects. In: Proceeding of the 2013 IEEE international conference on computer vision and pattern recognition (CVPR), pp 1352–1359
38. Sattler T, Leibe B, Kobbelt L (2011) Fast image-based localization using direct 2d-to-3d matching. In: Proceeding of the 13th IEEE international conference on computer vision (ICCV), pp 667–674
39. Shin D, Dunston P (2008) Identification of application areas for augmented reality in industrial construction based on technology suitability. Autom Constr 17(7):882–894
40. Snavely N, Seitz S, Szeliski R (2007) Modeling the world from internet photo collections. Int J Comput Vis 80(2):189–210
41. Strecha C, Pylvanainen T, Fua P (2010) Dynamic and scalable large scale image reconstruction. In: Proceeding of the 2010 IEEE international conference on computer vision and pattern recognition (CVPR), pp 406–413
42. Ufkes A, Fiala M (2013) A markerless augmented reality system for mobile devices. In: Proceeding of the 2013 IEEE international conference on computer and robot vision (CRV), pp 226–233
43. Wagner D, Reitmayr G, Mulloni A, Drummond T, Schmalstieg D (2010) Real-time detection and tracking for augmented reality on mobile phones. IEEE Trans Vis Comput Graphics 16(3):355–368
44. Wang X (2008) Improving human–machine interfaces for construction equipment operations with mixed and augmented reality. In: Robotics and automation in construction: new development. I-Tech Education and Publishing, pp 349–362
45. Woodward C, Hakkarainen M (2011) Mobile augmented reality system for construction site visualization. In: Proceeding of the international symposium on mixed and augmented reality (ISMAR), pp 1–6
46. Woodward C, Hakkarainen M, Korkalo O, Kantonen T, Aittala M, Rainio K, Kähkönen K (2010) Mixed reality for mobile construction site visualization and communication. In: Proceeding of the 10th international conference on construction applications of virtual reality (CONVR), pp 35–44
47. Wu C, Agarwal S, Curless B, Seitz S (2011) Multicore bundle adjustment. In: Proceeding of the 2011 IEEE conference on computer vision and pattern recognition (CVPR), pp 3057–3064