Change Detection and Model Update Framework for Accurate Long-Term Localization

Stefan Larsen[1], Ezio Malis[1], El Mustapha Mouaddib[2], Patrick Rives[1]

***Abstract*— The ability to perform long-term robotic operations in dynamic environments remains a challenge in fields such as surveillance, agriculture and autonomous vehicles. For improved localization and monitoring over time, this paper proposes a novel model update framework using image-based 3D change localization and segmentation. Specifically, shallow image data is used to detect and locate significant geometric change areas in a pre-made 3D model. The main contribution of this paper is the ability to precisely segment and locate both new and missing objects from few observations, and to provide consistent model updates. The applied method for geometric change detection is robust to seasonal, viewpoint, and illumination differences that may occur between operations. Qualitative and quantitative tests with both our own and publicly available datasets show that the model update framework improves on previous methods and facilitates long-term localization.**

I. INTRODUCTION

Model update is a crucial part to enable long-term robotic autonomy in dynamic environments. Many methods for localization and monitoring rely on stable and accurate maps, and struggle over time in outdoor (peri)-urban environments. Periodic changes from weather, vegetations and illuminations may cause errors for most vision-based methods. Dynamic objects like cars and pedestrians may change a scene during or in-between operations, causing uncertainty also for scanner-based approaches. Without precise and consistent model updates, scene changes may have significant impacts on the long-term performance of autonomous systems.

Research on update of environmental representation is increasing in popularity, due to the many demands from real-world applications. Autonomous vehicles need accurate maps to localize and maneuver safely and efficiently over time [1]. Monitoring systems for surveillance or environmental purposes rely on change detection (CD) for precise scene understanding [2]. Robots in agriculture must deal with changing vegetation and scenery on a regular basis [3].

The ability to detect changes and update a 3D model depends on the available sensor data. Typically during exploration, robots are equipped with systems of multiple high-end sensors like IMU, GPS-RTK, RGBD and LiDAR. The robots often perform thorough mapping of the scene, via teleoperation or pre-defined trajectories, until the entire scene is captured. Although this enables dense and advanced CD and model update methods, it is not a realistic scenario for all autonomous, low-cost robotic systems in the near future. For example, an autonomous robot equipped with only a subset of low-cost sensors is asked to deliver a package or maneuver towards its target location. Although the sensors enable it to navigate using a pre-build 3D map or image database, it will not take the time to monitor in details the entire scene around it, and its sensors are not able to update the model with the same quality in any case. However, the robot might still get glimpses of scene changes through a sensor, like a monocular RGB camera. This data could be used for continuous and consistent model updates, without the need of specific mapping operations.

The main objective of this paper is to address the problem of using shallow image data to detect changes and update a dense, pre-made 3D model. The data can be acquired during a generic mission, not necessarily dedicated to model update. An existing image-based method for 3D change detection based on geometrical ambiguities is used to detect scene change areas [4]. Then, this paper proposes to combine mesh segmentation and voxel grids to remove missing objects, and add new objects using an image-to-mesh network [5].

II. RELEVANT WORK

Existing works with simultaneous localization and mapping (SLAM) already show impressive performance in challenging environments. Short-term dynamics are typically dealt with by ignoring dynamic objects or image pixels suceptible to change. However, less work have looked into the problem of long-term spatio-temporal representations [6], and how it can facilitate model-based localization. At the same time, it is seen that many pose estimation methods struggle over time from changes in viewing conditions between reference and query models [7], which motivates the need for consistent and accurate reference model updates.

To avoid costly remappings of the entire scene, an intuitive approach for model update starts with detecting changes in the scene. While some recent CD works try to detect significant geometric change areas directly in 2D images using deep neural networks (DNNs) [8], methods incorporating 3D data show improved robustness to viewpoint and illumination differences [2]. Some of these works find inconsistencies between images and 3D models [4][9], while others use grids for point cloud comparisons [10]. Unfortunately, most methods only give rough estimates of change locations, like change ellipsoids [4][9], instead of updating the scene model.

Some works present new mapping or classification techniques to better locate, segment and track important changes. One initial approach used map differencing and object segmentation learning [11]. Signed distance functions can

[1]Stefan Larsen, Ezio Malis and Patrick Rives are with ACENTAURI team at Inria, Sophia-Antipolis, France stefan.larsen@inria.fr, ezio.malis@inria.fr, patrick.rives@inria.fr

[2]El Mustapha Mouaddib is with MIS lab, Amiens, France mouaddib@u-picardie.fr

reconstruct dense scenes from point clouds and images [12][13][14], while neural descriptor fields can perform robust object-level CD indoors [15]. Others attempt to classify or segment maps using random forest classifiers or DNNs [16]. While they show promising results, it remains a challenge to precisely distinguish new, moved and missing objects, both indoors and outdoors, regardless of object sizes and types. Additionally, these methods require streams of RGBD data to construct and update the environment, which is not feasible if only a few images are available.

Other methods show how only sparse change information can be enough to improve long-term localization. One technique uses dynamic occupancy grids to estimate the robot pose and environment state [17], while some others update feature-based maps to reduce localization error over time [1][18]. Semantic scene graphs can relocate the robot in changing indoors environments [19], which seems like a promising direction, but would require more available change-data for training.

Contrary to most existing approaches, this paper looks to update a dense, pre-made model using only shallow query image data. While previous work generally have dense query input data from RGBD video streams or LiDAR scanners, this paper considers the case where only the original environmental model is collected with high-resolution sensors. The query data comes from few samples of inexpensive sensors like monocular RGB camera. In this case, it is counterproductive to continuously update the entire model to incorporate changes. Instead, timely and precise local change updates will provide consistent model updates without propagations of errors. This work shows how recent methods for image-based 3D CD can assist visual localization, and provides the following main contributions:

- Segmentation and localization of changes in 3D model using shallow image query data
- Experiments showing how consistent model updates improve long-term accuracy of vision based localization

III. Proposed Framework

The proposed 3D model update framework, with steps to locate and segment changes in the scene, is described in Fig. 1. The initial 3D model is reconstructed as a surface mesh, to enable query images to detect significant change areas for both new and missing objects. For more precise change estimation, this paper proposes 3D reconstruction from change masks to generate newly discovered objects, and mesh segmentation and voxel grids to remove missing objects. The framework is designed to run offline, in-between robotic tasks. The updated model can then be used to localize the robot online during the subsequent operations.

A. Input Data

The initial 3D point cloud of an urban or peri-urban scene is typically acquired using cameras, possibly with depth estimates, or using LiDAR scanners. The proposed scheme for model update is designed for flexibility, and works for any dense point clouds. Then, the point cloud must be reconstructed into a mesh model with surfaces, preferrably into two different models: dense for CD and sparse for segmentation. The CD method used in this paper relies on accurate image to model correspondences, and gets more accurate image reprojections with a dense mesh. Meanwhile, the proposed segmentation method utilizes distinct shape differences, which are more apparent in sparse meshes.

Model updates are given by images from later inspections of the same scene, using RGB cameras. The camera intrinsic parameters and poses relative to the initial 3D model are needed, either directly from measurements, or from estimates using techniques like VO or SLAM with initial registration. For the proposed CD scheme using shallow data, a minimum of three images are needed from different viewpoints around each change area.

B. Image-Based 3D Change Detection

While there exists different approaches for 3D CD, as presented earlier, this paper choses to apply the methods for

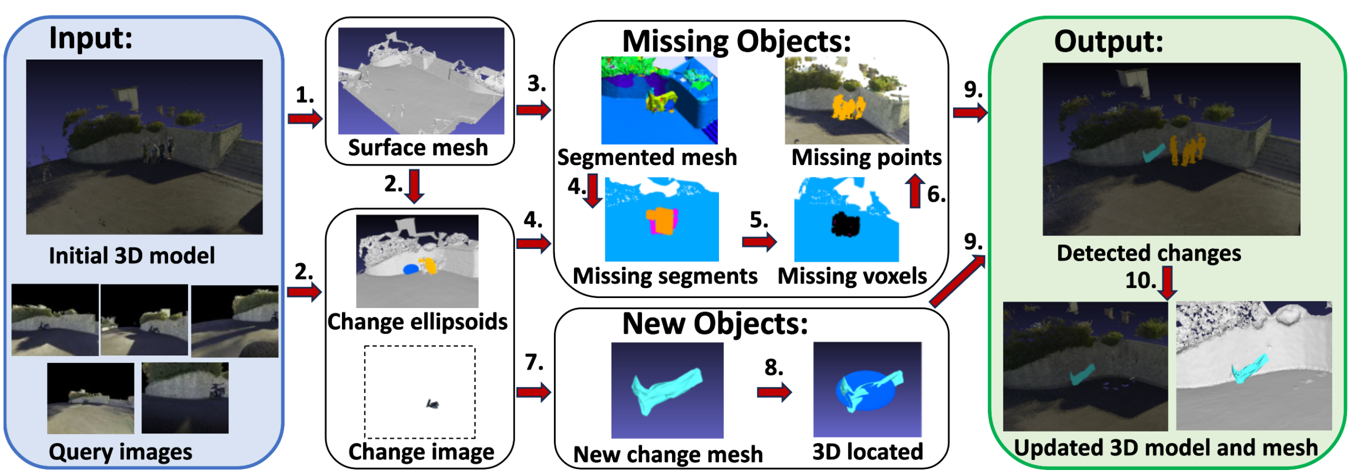

Fig. 1. Complete model update framework. 1. Surface mesh reconstruction. 2. Image based 3D change detection. 3. Geometric mesh segmentation. 4. Change ellipsoid and segment pairing for missing objects. 5. Voxelization of change segments. 6. Point occupancy collection. 7. Change mesh projection for new objects. 8. 3D change localization. 9. Detected changes. 10. Model update.

finding geometrical ambiguities [4][9]. These methods use pairs of query images to detect significant changes in a scene represented by an outdated 3D mesh. The images should be acquired within a narrow time frame to avoid structural or lighting changes between them. Objects that are moving during the acquisition should be filtered out using other methods, as the proposed framework only considers objects that have changed in-between the tasks. Typical examples are static pedestrians, parked cars and scooters, and changes to vegetation and construction sites.

For detecting inserted objects, 2D change masks are estimated for each reference image [9]. Each reference image is compared with reprojections of every other reference image from its own point of view, via the outdated 3D model. Changes in the foreground scene causes color ambiguities between each reference image and the reprojected images. The differences are stored in delta maps, and combined to reduce noise and improve accuracy of detected changes. Change location in 3D is estimated by triangulating the image changes based on the delta maps, resulting in 3D change ellipsoids, up to scale with the model.

This method was extended for removal detection [4]. In a sense, the problem was reverted, to focus on areas that should be occluded during the image reprojection process. Areas that result in the least change signify a previously occluded area, meaning foreground objects that are missing. By combining estimates from multiple images, missing objects are detected in images and localized as 3D ellipsoids.

While these approaches succeed in locating changes, the 3D change ellipsoids are not directly useful for consistent long-term model updates and localization. After a few iterations, the updated model would become crowded with overlapping ellipsoids. Instead, the proposed framework goes further, to use the image change masks and change ellipsoids to segment and update the changed objects in the model.

C. Locating and Segmenting New Objects

Significant changes may occur from new objects in the scene, like vehicles or construction work. Assuming new objects will remain in the scene for the near future, they must be added to the model. While it is possible to obtain dense 3D reconstructions directly from images, this usually requires multiple high quality images. However, the proposed scheme should be flexible to a limited amount of images depicting each change, and to less accurate change images.

Using the image data available, segmented images depicting only the new objects are obtained from the 2D change masks created during the CD method [9], as seen in Fig. 1.

For new-object localization, this paper proposes to directly insert the cut-out change images into a image-to-mesh network [20]. This end-to-end deep learning architecture produces a 3D mesh model directly from a few RGB images. Using a graph-based DNN with a coarse-to-fine strategy, further improved by leveraging cross view information, this network shows high 3D shape estimation accuracy. As with all DNN-based methods, this requires training data of similar objects. The quality of the results also depends on the quality of the object image, in this case from the size and accuracy of the change mask produced during the CD step.

Another key step for locating the new objects is to obtain the correct scale and 3D position of the newly created object mesh. Without available dense maps, this must be done through triangulation from the camera viewpoints. As the proposed framework already estimates the position of 3D change ellipsoids, the newly created object mesh is centered at the corresponding ellipsoid, and scaled to the size of the ellipsoid, as seen at step 8. in Fig. 1.

D. Locating and Segmenting Missing Objects

Other typical scene changes occur from objects missing from the initial model. Assuming missing objects will not return to the same location in the scene again, they must be removed from the model. Here, the solution is to infer which missing objects the removed-change ellipsoids have detected, and extract them. This is done by segmenting the mesh model, and pairing change ellipsoids with candidate mesh segments. While well trained DNNs show impressive semantic segmentation performance [21] geometry-based methods are often easier to implement with equivalent results. The implementation shown in Fig. 1 uses a geometric, clustering-based approach [22]. For change pairing, each removed-change ellipsoid is converted into dense 3D voxelgrids. Mesh segments who are overlapping with a voxelized ellipsoid, correspond to missing objects. This step requires an accurate mesh segmentation and few close objects, to avoid extracting nearby, unchanged parts of the scene.

To complete this change localization step, the corresponding changed points in the initial 3D model must be determined. Many surface reconstruction methods simplify the model, either by filtering nearby points, or by interpolating new points for smoother surfaces. Thus, deleting the change segment vertex points directly from the 3D model may not be a straightforward task, at least not to remove all the changed points. Here, from the idea of voxelization for occupancy checks, the change segments are also voxelized, and the initial 3D points contained within them can be marked as change, as seen as the yellow points in Fig. 1.

E. Long-Term Model Update

Finally, the initial 3D model can be updated with any new and missing objects. The new objects can be added as vertex points, with or without triangle faces, and the points of missing objects from the previous occupancy check can be subtracted. However, removing points directly from a 3D model may create holes, depending on the model and accuracy of the segmentation. If an application requires a solid and refined update model, any small gaps could be filled instantly with interpolation or hole filling algorithms [22], or with new data during the next acquisitions.

For long-term tasks with multiple model update iterations, the proposed framework must also be robust to initial models with gaps or various densities. By design, any small gap in the initial 3D model will be covered during the first mesh reconstruction step, to provide a solid model for the following detections using image reprojections, seen in Fig. 1, step 10. on the right. This effect motivates the use of surface meshes for image reconstruction, compared to image rendering from point clouds which are more prone to gaps or sparse data. For image-based localization, the gaps can simply be ignored using depth masks. Alternatively, Gaussian splatting techniques [23] may create dense images for image reprojections. However, accurate results would require optimization with multiple images of the scene.

IV. Experiments and Discussion

Results are presented to illustrate how the proposed model update framework performs on three different datasets, with precise model updates for new and missing objects. Simulations demonstrate how the proposed model update improves localization in changing outdoor environments over time.

A. Experimental Setup and Datasets

Surface reconstruction and mesh segmentation is done with CGAL [22]. 3D CD is done using the aforementioned image-based method [4]. Reconstruction and mesh wrapping is done with a pretrained model of the Pixel2Mesh++ network [20]. Colors for mesh segmentation and new objects are set arbitrarily in the figures. Feature based localization [24] is combined with dense camera pose optimization [25].

B. Results with 2017 Palazzolo and Stachniss Dataset

This dataset contains five different urban scenes, where the query images depict new structural elements not present in the models. To demonstrate detection and update of missing objects, 3D objects are inserted manually to the initial model.

The results with the statue scene are illustrated in Fig. 2. The proposed framework is able to detect the new statue and the missing foreground objects, segment them and finally update the model, see nr. 8 and nr. 9 in Fig. 2. This provides a more detailed model update with respect to the initial change ellipsoids [4][9] seen in nr. 3 in Fig. 2.

Three masks where used to recreate the new statue object using [20]. Since the pre-trained network was not trained on human-shapes, the obtained object shape is not accurate. However, it is stored as a floating, watertight mesh within the boundaries of the change ellipsoid. The intention is that this added object provides more accurate information of the current scene. Later, additional acquisitions around the statue can be used to further refine the reconstruction.

Deleting foreground objects creates gaps, causing trouble for some applications. Here, the holes were covered by a remeshing of the updated model, as seen in nr. 9 in Fig. 2, however hole filling techniques could also be applied [22].

C. Results with Fehr et al Dataset

This dataset consists of three indoor scenes mapped by a handheld RGB-D camera. There are about 50-100% overlap between each scene, where the changes mostly come from moved furnitures. Due to the close-up camera viewpoints, the changes are only partially visible in the images.

Results from the living room scene are shown in Fig. 3. The framework detects two of the moved furnitures, a pouf and a table, as missing objects, and updates the model. The removal of the pouf left a hole, which was repaired using a mesh hole filling technique [22], as seen in nr. 5 in Fig. 3.

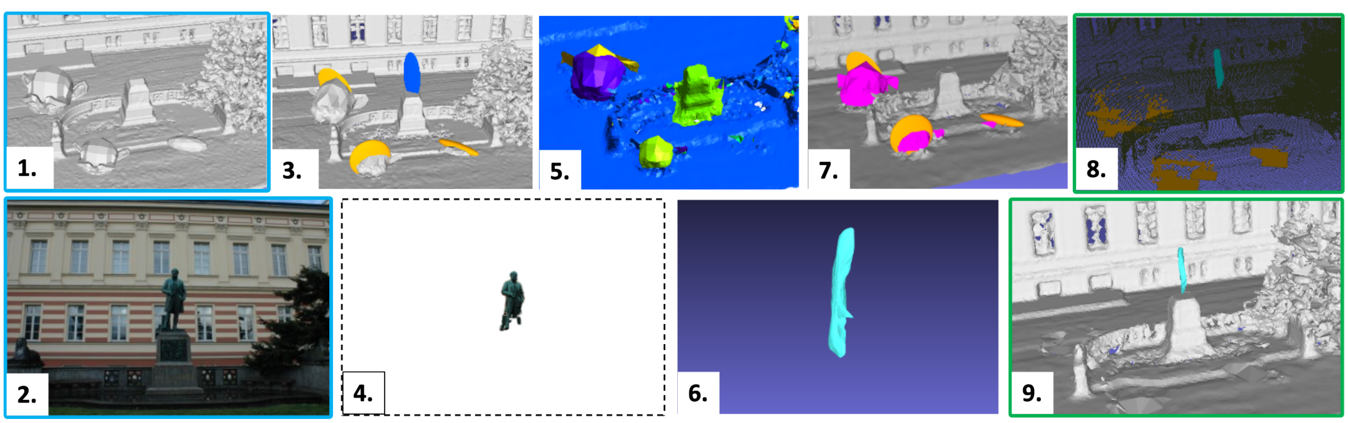

Fig. 2. Results from statue data. 1. Initial mesh model. 2. A query image. 3. Mesh model with change ellipsoids. 4. A change mask. 5. Segmentation. 6. New object mesh. 7. Pairing of missing objects. 8. Updated model point cloud. 9. Updated mesh reconstruction.

Fig. 3. Results from Fehr data. 1. A query image. 2. Initial mesh reconstruction with detected change ellipsoids. 3. Mesh segmentation. 4. Pairing of missing objects. 5. Updated mesh model with hole filling reconstruction applied.

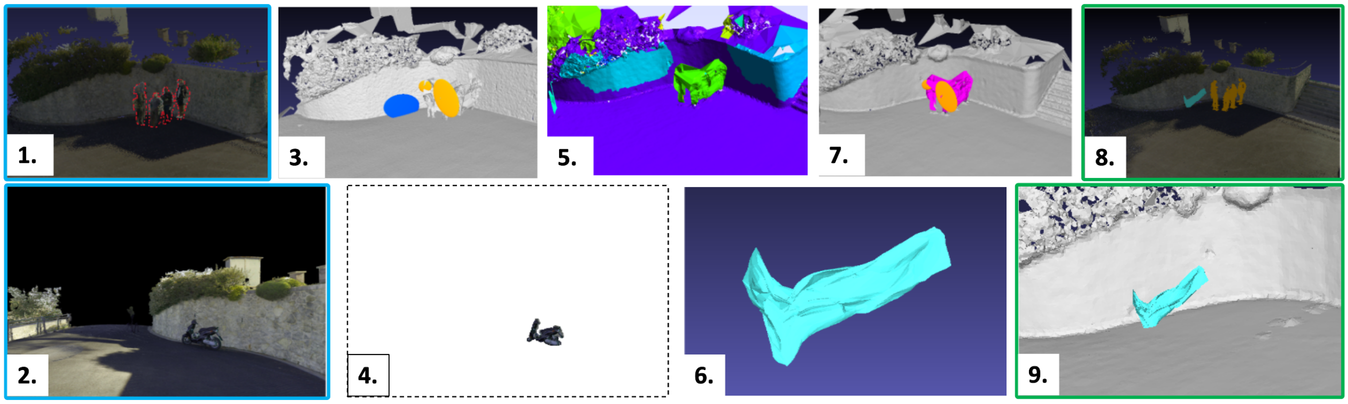

Fig. 4. Results from OCA data. 1. Initial model with pedestrians outlined in red. 2. A query image. 3. Mesh reconstruction with change ellipsoids. 4. A change mask. 5. Segmentation. 6. New object mesh. 7. Pairing of missing objects. 8. Updated model point cloud. 9. Updated mesh reconstruction.

While it is accurate to say the objects were missing from their original locations, it would be even more accurate to describe them as moved. However, this would require exploitation of semantic information, wheras this framework is based on geometric appearences. The new positions of the table and pouf could also be detected as new objects, but the objects where not sufficiently visible in the available images.

D. Results with 2023-OCA Dataset

Due to the lack of datasets for long-term change detection in outdoor scenes, the 2023-OCA dataset has been created in the context of long-term shareable mapping for collaborating multi-robot systems. Specifically, this dataset contains two dense point clouds with rendered images. It was obtained at the Observatory of Nice (OCA), using a Leica RTC 360 LiDAR scanner with internal RGB camera. The acquired models include object changes in the scene, which are depicted in manually rendered query images. For better demonstration in this paper, a group of pedestrians was manually moved closer to the location of a scooter.

The results in Fig. 4 show how the framework detects the changed objects and updates the model. The initial model contains pedestrians (outlined in red), while the query images show that the pedestrians are gone and a scooter has appeared. The pedestrian segments are detected and removed from the model, while the scooter is reconstructed and placed in the scene, see nr. 8 and nr. 9 in Fig. 4.

While the pedestrians are properly detected by the framework, the recreated scooter object is not very accurate. This is because only one of the change masks produced by the 3D CD method [4] were of adequate quality. The masks could be improved by previously mentioned DNN based CD approaches. However, that requires registered images before the change occured, and will be affected by illumination variations between the new and old images. Since the focus of the paper is mainly on CD for model update, only a pretrained reconstruction model [20] was used, and it was not able to recognize the scooter object type. Obviously, training the network on more similar object types would improve the results of this object reconstruction step.

E. Results from Long-Term Localization

Experiments have been conducted to show how the proposed model update framework improves long-term localization. A typical localization setup consists of feature based localization for initial pose estimates, followed by pose optimization. For this experiment, a hierarchical method using Structure-from-Motion (SfM) and feature-based DNN creates the initial estimates [24]. They are then optimized using a dense, iterative method [25], to find the camera transformation between the query and reference images, using corresponding reference depth maps. Each estimated query pose is compared to the true query pose to determine a localization error. Changes between the reference model and the model contained in the query images affect the results.

The data has been rendered from the OCA dataset. The query images are chosen similar to nr. 2 in Fig. 4, such that they depict the scooter area from different viewpoints.

To showcase the effects of model update for localization in changing environments, references are rendered from three different models. One set of references is taken from the original model with pedestrians, as in Fig 4 nr. 1, denoted $Original$. Another is taken from the original model with pedestrians, including the blue ellipsoid from Fig 4 nr. 3 representing the detected scooter, denoted $Roupin$[4]. The third set is taken from the updated model using the proposed framework, which includes the reconstructed scooter and removes the pedestrians, as in Fig. 4 nr. 9, denoted $Ours$.

In total, 10 reference images per model and 100 query images are rendered around the change area, with splits of 10 queries per reference. There is a difference in translation of 0-1 meters and rotation of 0-10 degrees between each pair of query and reference pose. For each reference model, Table I shows the total number of images that converged, number of iterations for convergence, and the mean error of translation E_T and rotation E_R and their standard deviations. The averages are taken from when all models converged, which was only 23 of the 100 query images.

The updated model $Ours$ reduces the pose error compared to the others, and converges faster and almost every time. The residuals for all models can be explained by the imperfect

TABLE I
LOCALIZATION RESULTS FROM THREE MODELS OF OCA DATASET

Model	Converged	Iterations	Mean pose errors	
			$E_T[cm]$	$E_R[deg]$
$Original$	31 / 100	903	21.4 ± 11.4	4.11 ± 2.67
$Roupin[4]$	69 / 100	948	19.6 ± 11.2	3.62 ± 1.68
$Ours$	**97 / 100**	**414**	**18.7 ± 10.2**	**1.23 ± 1.03**

reference models used, and outliers and artifacts from vegetation. While each query and reference image are initially well aligned by the feature-based localization [24], within 1 meter from the true pose, the changes contained in the queries with respect to the outdated references cause significant errors for the optimization step. Particularly, removal of the pedestrians is influential, as they occlude and cause divergence for the outdated models. $Original$ generally performs worse than $Roupin[4]$, since the added ellipsoid helps to cover artifacts and potentially mismatched features from the scooter.

The differences between the ellipsoid and estimated scooter mesh also affect the pixel comparisons of the pose estimation. Neither of the shapes are accurate, and their colors are not obtained, but manually painted black when rendering images. However, the estimated scooter shape is generally more accurate than the ellipsoid and creates less occlusions, for improved pose estimation.

V. CONCLUSIONS AND FUTURE WORK

A novel framework for 3D model update is proposed for long-term autonomous robotic operations like localization and monitoring in (peri)-urban environments. It demonstrates how surface reconstructions, model segmentations, voxel grids and image-based reconstructions can be chained together to succesfully update a model with both new and missing objects. It can be seen as an extension on previous change detection methods [4][9]. While they only indicate change locations in the scene, the proposed framework provides precise segmentation and localization of changes for consistent model updates. The usefulness of this approach is demonstrated through localization experiments.

Improvements for more robust and precise model updates could be to include semantic classification for mesh segmentation and object tracking. Future work could also improve the 3D CD step, to better handle reflections and occlusions. Extensions could include pose estimation for missing camera parameters, or LiDAR or depth images as query data.

ACKNOWLEDGMENT

This project is funded by the ANR SAMURAI project (ANR-21-CE33-0014).

REFERENCES

[1] J. S. Berrio, S. Worrall, M. Shan, and E. Nebot, "Long-term map maintenance pipeline for autonomous vehicles," *IEEE T-ITS*, 2021.

[2] A. Kharroubi, F. Poux, Z. Ballouch, R. Hajji, and R. Billen, "Three dimensional change detection using point clouds: A review," *Geomatics*, 2022.

[3] M. Hussain, D. Chen, A. Cheng, H. Wei, and D. Stanley, "Change detection from remotely sensed images: From pixel-based to object-based approaches," *ISPRS JPRS*, 2013.

[4] O. Roupin, M. Fradet, C. Baillard, and G. Moreau, "Detection of removed objects in 3d meshes using up-to-date images for mixed-reality applications," *Electronics*, 2021.

[5] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2mesh: Generating 3d mesh models from single rgb images," in *Proceedings of the European conference on computer vision (ECCV)*, 2018.

[6] L. Schmid, M. Abate, Y. Chang, and L. Carlone, "Khronos: A unified approach for spatio-temporal metric-semantic slam in dynamic environments," in *Proc. of Robotics: Science and Systems*, 2024.

[7] C. Toft, W. Maddern, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, T. Pajdla, *et al.*, "Long-term visual localization revisited," *IEEE PAMI*, 2020.

[8] W. Shi, M. Zhang, R. Zhang, S. Chen, and Z. Zhan, "Change detection based on artificial intelligence: State-of-the-art and challenges," *Remote Sensing*, 2020.

[9] E. Palazzolo and C. Stachniss, "Fast image-based geometric change detection given a 3d model," in *IEEE ICRA*, 2018.

[10] W. Ma, Z. Song, Y. He, and S. Shen, "Change detection and update of 3d sparse map by merging geometry and appearance," *International Journal of Applied Earth Observation and Geoinformation*, 2023.

[11] R. Finman, T. Whelan, M. Kaess, and J. J. Leonard, "Toward lifelong object segmentation from change detection in dense rgb-d maps," in *IEEE European Conference on Mobile Robots*, 2013.

[12] M. Fehr, F. Furrer, I. Dryanovski, J. Sturm, I. Gilitschenski, R. Siegwart, and C. Cadena, "Tsdf-based change detection for consistent long-term dense reconstruction and dynamic object discovery," in *IEEE ICRA*, 2017.

[13] J. Fu, C. Lin, Y. Taguchi, A. Cohen, Y. Zhang, S. Mylabathula, and J. J. Leonard, "Planesdf-based change detection for long-term dense mapping," *IEEE RA-L*, 2022.

[14] L. Schmid, J. Delmerico, J. L. Schönberger, J. Nieto, M. Pollefeys, R. Siegwart, and C. Cadena, "Panoptic multi-tsdfs: a flexible representation for online multi-resolution volumetric mapping and long-term dynamic scene consistency," in *IEEE ICRA*, 2022.

[15] J. Fu, Y. Du, K. Singh, J. B. Tenenbaum, and J. J. Leonard, "Robust change detection based on neural descriptor fields," in *IROS*, 2022.

[16] E. Langer, T. Patten, and M. Vincze, "Robust and efficient object change detection by combining global semantic information and local geometric verification," in *IROS*, 2020.

[17] G. D. Tipaldi, D. Meyer-Delius, and W. Burgard, "Lifelong localization in changing environments," *IJRR*, 2013.

[18] E. Derner, C. Gomez, A. C. Hernandez, R. Barber, and R. Babuška, "Change detection using weighted features for image-based localization," *Robotics and Autonomous Systems*, 2021.

[19] J. Kabalar, S.-C. Wu, J. Wald, K. Tateno, N. Navab, and F. Tombari, "Towards long-term retrieval-based visual localization in indoor environments with changes," *IEEE RA-L*, 2023.

[20] C. Wen, Y. Zhang, Z. Li, and Y. Fu, "Pixel2mesh++: Multi-view 3d mesh generation via deformation," in *ICCV*, 2019.

[21] J. Wald, K. Tateno, J. Sturm, N. Navab, and F. Tombari, "Real-time fully incremental scene understanding on mobile platforms," *IEEE RA-L*, 2018.

[22] The CGAL Project, *CGAL User and Reference Manual*, 5.1 ed. CGAL Editorial Board, 2020.

[23] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, 2023.

[24] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine: Robust hierarchical localization at large scale," in *ICCV*, 2019.

[25] A. I. Comport, E. Malis, and P. Rives, "Real-time quadrifocal visual odometry," *IJRR*, 2010.