Long-term localization with map compression based on solar information

Youssef Bouaziz, Eric Royer, Guillaume Bresson, Michel Dhome

Long-term localization with map compression based on solar information

Youssef Bouaziz[1,2], Eric Royer[1], Guillaume Bresson[2] and Michel Dhome[1]

***Abstract*— In this paper we address visual based localization in outdoor environments where the appearance changes dramatically. Such environmental changes result in a substantial transformation of the visual information of the scene, producing a significant impact on the visual based localization performance. Hence, these changes can lead to major difficulties when associating data between the current image and the landmarks in the map. One solution for this problem is to keep adding landmarks to the map in order to cover various environmental conditions. However, this solution leads to a continued growth of the map, which in turn, will result in a costly and resource-intensive localization. In this paper we present a map management approach in which we exploit information related to the sun's position to compare resemblance between the traversals in the map and maintain a diverse map that incorporates a minimum amount of data and ensures a reliable localization in different environmental conditions. We evaluated our approach on a dataset that incorporates more than 100 sequences with different environmental conditions and we compared the obtained results with a state of the art approach.**

I. Introduction

Visual based localization (VBL) has become a fundamental field in robotic applications and it represents an interesting alternative to laser-based systems since it can provide an accurate localization with inexpensive setup requirements. Therefore, VBL has drawn the interest of many researchers over the past few years and has seen a growing number of real-time applications such as autonomous driving, augmented reality, robot navigation.

In this paper, we are interested in real-time VBL in outdoor environments for autonomous shuttles. In such applications, shuttles are repeatedly traversing the same path at different times. This means that they are very likely to experience many different environmental conditions which can deteriorate the localization performance even when revisiting familiar places. Day-night transition is one of the most challenging cases for VBL in outdoor environments as it engenders enormous variations in luminosity and brightness that lead to a significant difference between images captured at different times of the day. These variations can result in major difficulties when associating data between the current image and the landmarks in the map. Autonomous shuttles must deal with such environmental changes in order to provide reliable long-term localization.

Building a map that covers all environmental conditions by continuously adding landmarks to this map can help improving localization performance in changing conditions. However, this will result, also, in a ceaseless growth of the map's size that is relative to the number of traversals (experiences) performed by the shuttle. This means that after several traversals, localization will require an immense storage space to store the map and high-end CPU to find matching points between the current image and the corresponding landmarks in a vast database. In other words, real-time long-term localization will be impossible after a certain number of localization sessions. Thus, a map update strategy is required to prevent such cases and to ensure reliable real-time long-term localization.

In our previous work [1], we employed an autonomous shuttle for three months, totaling nearly 1500 km of autonomous travel on an industrial site. During this operation, we experienced some difficulties for long-term navigation. One of the most challenging difficulties identified in this previous work is lighting changes over the day, where the tests demonstrated that variations in lighting (caused by changes in the direction of the sunlight and the sun elevation) have more impact than long-term changes of the environment on the localization performance. This has motivated us to exploit information related to the sun position to design a map management approach with the purpose of preventing the continuous growth of the map. The sun position information are used to determine which traversals to keep and which ones to remove from the map. Traversals with correlated sun positions coordinates are considered to have similar environmental conditions. Thus, our approach consists, in a first time, in computing the sun positions related to all the traversals in the map, then, in exploiting all those computed values to classify this map into relevant and irrelevant traversals to finally produce a compressed map that incorporates a minimum number of traversals with diverse environmental conditions.

Our approach is employed on an experience-based mapping system [2], [1] that is based on key-frames and local features and is fundamentally similar to several open-source frameworks like ORB-SLAM [3] and Maplab [4]. In our experiments, we use Harris corner detector [5] for extracting key-points which are matched with ZNCC — Zero-mean Normalized Cross-Correlation — computed on 11×11 pixel windows around each key-point. However, our method can still be applied in the same way using other descriptors. We evaluated our approach on a new dataset recorded on our vehicle that we make available to the community and which

[1]Université Clermont Auvergne, CNRS, SIGMA Clermont, Institut Pascal, F-63000 CLERMONT-FERRAND, FRANCE

[2]Institut VEDECOM, 23 bis allée des Marronniers, 78000 Versailles, France

is called IPLT[1](Institut Pascal Long-Term) dataset. We also compared the results obtained with our approach with the results obtained by another state of the art approach.

II. Related work

Considerable efforts have been made in VBL in static environments or with few minor changes, but it is only recently that localization in dynamic environments with changing environmental conditions has been addressed. Achieving reliable lifelong navigation under such environments is one of the biggest challenges for VBL.

Many works [6], [7] have addressed this challenge by using image based matching techniques since traditional feature-based comparison methods have shown weaknesses in changing conditions. Murillo and Kosecka [6] proposed to improve localization in such environments by describing places with global descriptors computed on the entire images. However, this requires an exhaustive search in the map to find matches and recognize places. This means that in large-scale environments, such an operation will be very costly and will require high-end CPUs to achieve real-time performance. In order to improve localization with global images descriptors, Milford and Wyeth [7] enhanced global image descriptors performance by introducing a sequential images matching technique called SeqSLAM. This technique consists of matching a sequence of current images with sequences of images in the database. SeqSLAM has significantly improved the performance of global images descriptors under harsh conditions, but it exhibits major sensitivity to changes in viewpoint.

Deep learning was employed in some approaches like [8], [9], [10] to ameliorate localization performance in dynamic environments. However, these methods require high-end GPUs to achieve real-time performance and cannot provide a 6DoF pose estimation. Recently, several luminance invariant descriptors such as LIFT [11], LUIFT [12], SOSNet [13] and REST [14] were proposed to improve accuracy of matching features in outdoor environments where brightness is subject to important changes. Such descriptors can achieve better performance in dynamic environment than SIFT [15] and SURF [16] and they can be employed with other map management approaches to further improve VBL robustness under challenging conditions.

In their recent works, Bürki *et al.* [17] addressed the landmarks retrieval issue in changing conditions. They have employed information retrieval techniques from document retrieval community to fetch the landmarks that are related to the current environmental conditions. Similarly, MacTavish *et al.* [18] inspired a landmark retrieval approach from recommender systems. They have proposed a collaborative filtering approach that recommends experiences according to the current environmental conditions.

In order to control the map's growth and to ensure a lifelong navigation, Muhlfellner *et al.* [19] proposed a map management approach where they are scoring landmarks according to the number of distinct localization sessions in which they appear. Afterwards, the landmarks with the lowest scores are removed in the offline map maintenance operation. Bürki *et al.* [20] have proposed an approach in which they have combined a landmark retrieval technique with a map management operation. In this approach, the retrieved landmarks will be updated if they lead to a successful localization, otherwise, they will be replaced by new landmarks.

In this work, we propose a map management strategy with the aim of reducing the size of the map. This strategy depends only on solar information related to each localization session. The results show that localization on a map produced by our approach was more reliable than localization on a map produced by Muhlfellner *et al.'s* approach [19].

III. Methodology

This section addresses our proposed map update strategy. In this approach we aim to maintain a reliable map with a fixed size throughout the frequent traversals. The core idea of our approach is to produce a map that incorporates a minimum number of traversals ($\hat{N}$) and able to operate in different environmental conditions. $\hat{N}$ is the number of traversals to be maintained in the map after carrying out the map management. Different values of $\hat{N}$ were chosen in the experiments section to evaluate the efficiency of our approach in different cases.

The methodology of our approach is described in the Figure 1. After each localization session, we test if the number of traversals in the map (N) is greater than the predefined number $\hat{N}$. If it is the case, our map management algorithm will be executed offline to reduce the size of the map. $N > \hat{N}$ means that the map contains an additional

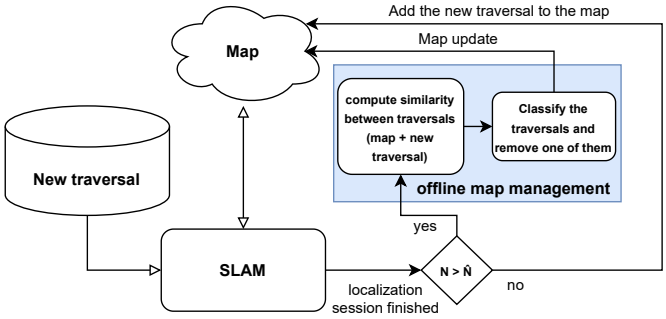


Fig. 1: A diagram illustrating the map management process. A new traversal is localized and added to the existing map. After finishing localization and mapping with SLAM, the map management algorithm is executed offline to check whether the map needs to be compressed or not (if $N > \hat{N}$). If the answer is yes, the algorithm has to act in two parts. Firstly, it has to compute the similarities between the N traversals in the map ($N = \hat{N} + 1$). Secondly, it uses these similarity information to classify all the traversals and pick out the one that needs to be removed.

traversal ($N = \hat{N} + 1$) which also means that it occupies additional memory space. In this case, our approach has to act offline to bound the size of the map, i.e. it has to

[1]We invite you to visit this link `http://ipltuser:iplt_ro@iplt.ip.uca.fr/datasets/` to download our dataset.

decide which traversal has more resemblance to the others to eventually remove it.

As explained in the Figure 1, this mechanism incorporates two main parts, the similarity computation part in which we exploit information related to the sun position at each acquisition time and associate the traversals with their corresponding environmental conditions to finally compute the similarity between them with respect to the associated environmental conditions, and the classification part in which we use these similarity information to classify the traversals according to their importance and to remove the less important ones.

A. Similarity computation

In this part, we take advantage of the sun position to compare resemblance between the traversals in the map in order to determine which ones to keep in the map and which ones to remove. For each traversal in the map, we use the Astronomical Almanac's algorithm [21] to compute the corresponding sun spherical coordinates: the Solar Elevation Angle (el) and the Solar Azimuth Angle (az) (see Figure 2). This algorithm takes as inputs the acquisition date/time and GPS latitude/longitude coordinates of each corresponding traversal. These inputs (date/time + GPS coordinates) are captured at the first frame of each traversal. This can be valid only when using short sequences like the case in this paper where we used $\sim$ 200 m length sequences, each one of them was recorded over $\sim$ 2 minutes. This means that there will be no important gap in the sun position between the start and the end of the traversal.

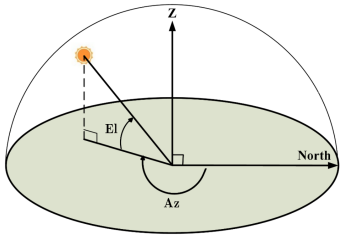


Fig. 2: Solar Elevation Angle and Solar Azimuth Angle [22].

After computing all the sun positions associated to the N traversals in the map, we generate a similarity matrix D with the shape of $[N \times N]$. D is a 2D symmetric distance matrix containing the distances, taken pairwise, between the computed sun positions of the N traversals.

In this paper, we tested two variants of the similarity matrix D. The first variant, D_{el}, is built by computing the distances between the elevation angles of the traversals:

$$D_{el}(i,j) = dist(el_i, el_j), \quad \forall i,j \in [1,N] \tag{1}$$

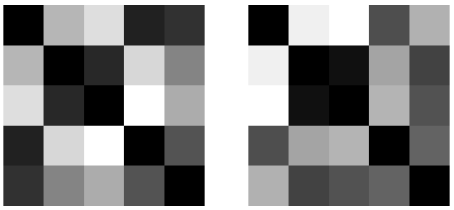
Where the function $dist$ computes the angular distance between two solar elevation angles. The second variant, D_{az_el}, is built using both of elevation and azimuth angles. For each traversal i, we compute the Cartesian vector from the spherical coordinates:

$$\vec{u_i} = \begin{pmatrix} \cos az_i \cos el_i \\ \sin az_i \cos el_i \\ \sin el_i \end{pmatrix}, \quad \forall i \in [1,N] \tag{2}$$

After that, we compute the distance matrix D_{az_el} by calculating the angle between each pair of the N Cartesian vectors:

$$D_{az\_el}(i,j) = |\arccos(\vec{u_i} \cdot \vec{u_j})|, \quad \forall i,j \in [1,N] \tag{3}$$

Figure 3 illustrates an example of the similarity matrices D_{el} and D_{az_el}. Both matrices were built using the same map that contains 5 traversals.

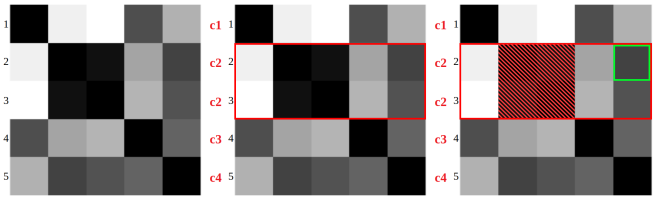


(a) D_{el} (b) D_{az_el}

Fig. 3: Example of matrices D_{el} and D_{az_el} built with a map containing $N = 5$ traversals. The values of the matrices were normalized between 0 and 1, black color refers to 0 and white color refers to 1.

B. Classification and traversal removal

In this part, we exploit the matrix D (D_{el} or D_{az_el}) built in the previous step to find out which traversal has to be removed. A hierarchical clustering [23] algorithm is used to classify the matrix D in order to select the traversal to remove. Figure 4 shows the different steps of the classification and the traversal removal:



(a) D (b) Clustering D (c) Searching for min

Fig. 4: Steps for classification and the selection of the traversal to remove.

(a) D is the distance matrix obtained from the previous section, we apply this method the same way on D_{el} and on D_{az_el}. In this example, D refers to matrix D_{az_el} obtained with a map containing $N = 5$ traversals.

(b) We classify D into $\hat{N}$ (4 in this example: c1, c2, c3 and c4) classes using the hierarchical clustering algorithm. This will result in classifying the two traversals i and j having the highest similarity in the same class (class c2, i.e., $i = 2$ and $j = 3$ in this example).

(c) In this step, we want to remove a traversal from the map while maintaining it as diverse as possible. To do so, we have to remove either the traversal i or the traversal j. Therefore, we search which one of them has more similarity to the other traversals by looking for for the minimum value in the i^{th} and j^{th} rows of the matrix while ignoring the diagonal and the elements with coordinates (i, j) and (j, i) (the hatched area in the figure). After finding the minimum, we remove its corresponding traversal (traversal 2 in the example).

Lighting changes produced by day-night transitions can result in enormous visual gap between images recorded in the same place but at different times of the day [14]. This can create serious difficulties in local features matching especially using traditional features descriptors like SIFT and SURF. Our previous works [1] demonstrated that the hour of the day has more impact than long-term changes of the environment on the localization performance. This was demonstrated after performing localization daily over 3 months on a map recorded once at the beginning of the experiment. The results show that variations in the sun elevation and in the direction of sunlight are resulting in a more reduced number of correctly matched features than other factors like a moderate rain or a gap of 3 months.

Localization under night condition can be considered as a particular case since there are no lights provided by the sun after the astronomical night. Therefore, street lights are almost the only source of lighting which means that all sequences recorded in the night-time are visually similar. This means that keeping more than one night sequence in the map is pointless, and on the other side, removing all night sequences can lead to a serious problem and extremely impact the localization performance on night sequences. For this reason, we impose a new constraint to our algorithm: the traversal with the lowest solar elevation angle will not be nominated as the traversal to be removed (the night traversals have a negative solar elevation angle), any other night traversal will be removed. This will guarantee that the produced map will incorporate only one night traversal.

IV. Experiments and Results

To evaluate the performance of our approach, we use our own collected dataset (IPLT dataset). This dataset contains currently more than 100 sequences and each one of them is about 200 m length. Our dataset incorporates multiple environmental conditions (day, night, dusk, rain, overcast...) and in all the sequences the vehicle has followed the same path in a parking lot as shown in the Figure 5.

We were also interested in evaluating our approach on other widely used datasets like Oxford RobotCar dataset [24] or NCLT dataset [25], but unfortunately those datasets do not provide a great number of sequences traversing the same path which is primordial to test the efficiency of our work. Therefore, we made our dataset open to the community hoping that it will help researchers in this field to overcome this lack. Our dataset was created from recorded images of two gray-scale 100° FOV cameras mounted on our experimental

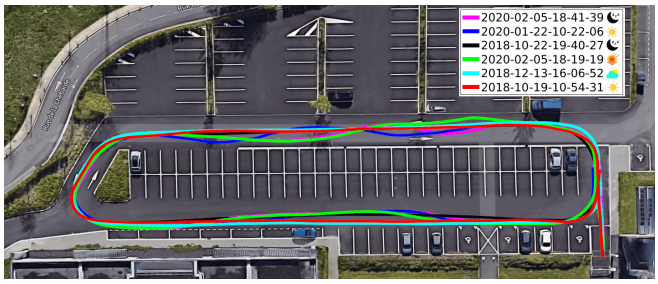


Fig. 5: Example of some sequences from IPLT dataset which were recorded in a parking lot.

vehicle (one front and one rear camera) and wheel-odometry. We divided our dataset into 10 mapping sequences and 93 test sequences. The mapping sequences are used to build our reference map and the test sequences are used to evaluate the localization performance on the produced map with our map management approach. The 10 mapping sequences are constituted from 3 sequences recorded in a sunny condition, 3 in overcast, 2 in rain, 1 in dusk and 1 in night. In Figure 6, we present an overview of images from the 10 mapping sequences.

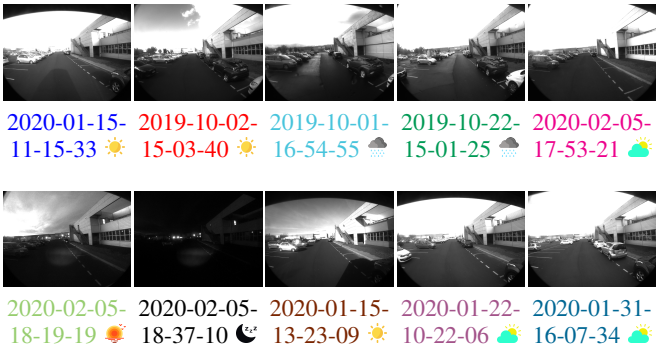


Fig. 6: An overview of images from the mapping sequences taken with the front camera. For each sequence we are indicating the acquisition date by a different color and symbolizing the environmental condition by a small icon (the colors are used to make easier reading Table I).

As mentioned in Section III, our approach has to limit the number of traversals used to build the reference map to a predefined number $\hat{N}$. To do so, we keep performing SLAM on the mapping sequences, one by one, and at the end of each session we employ our map management approach to shrink the size of the map as described in Section III.

Considering that the order in which the sequences are added to the map can affect the resulting map, we tested our approach with $100,000$ different orders of the N traversals. This will result in reproducing several different $\hat{N}$-session maps (We call an $\hat{N}$-session map a map composed of $\hat{N}$ traversals) and the most reproduced $\hat{N}$-session map, denoted by M^*, will be used in our tests. We have obtained 4 different $\hat{N}$-session maps using different configurations of our approach: M^*_{el}, $M^*_{az\_el}$, $\widetilde{M}^*_{el}$ and $\widetilde{M}^*_{az\_el}$. The maps M^*_{el} and $M^*_{az\_el}$ are obtained by performing our approach on

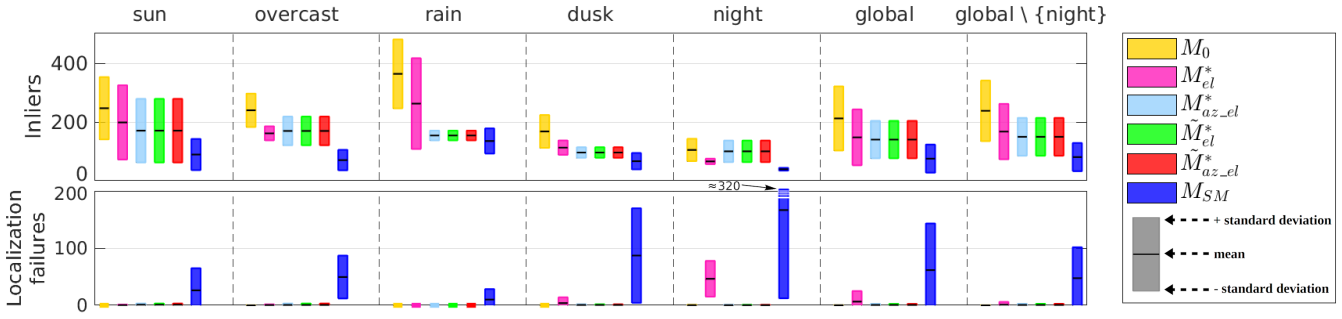
the two distance matrices D_{el} and $D^*_{az\_el}$ respectively. We denote $\tilde{M}^*$ the map that has been obtained by performing our approach while imposing the constraint that ensures keeping a night sequence on the map as described in Section III-B.

In the results, we represent the average number of inliers observed in the test sequences along with the number of localization failures as criteria to evaluate the performance of the localization. Practically, we found that the localization can be considered as reliable when there are at least 30 points matched between the current image and the database, below this threshold, we count a localization failure. This is a conservative threshold to ensure the security of our autonomous vehicle [1].

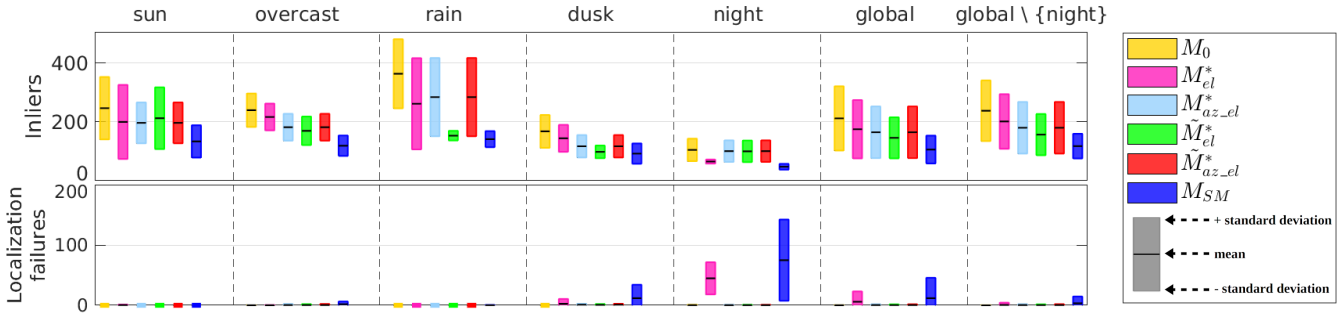
In Figure 7, we present a comparison of the localization performance between the different $\hat{N}$-session maps obtained by our approach (M^*_{el}, $M^*_{az\_el}$, $\tilde{M}^*_{el}$ and $\tilde{M}^*_{az\_el}$) and the map M_0 that incorporates all the 10 traversals of the mapping sequences. We also compared our approach with the map M_{SM} generated by the Summary Maps approach proposed by Muhlfellner *et al.* [19].

In their approach, Muhlfellner *et al.* are scoring the landmarks by the number of distinct sessions in which they appear and the ones with lowest scores are removed.

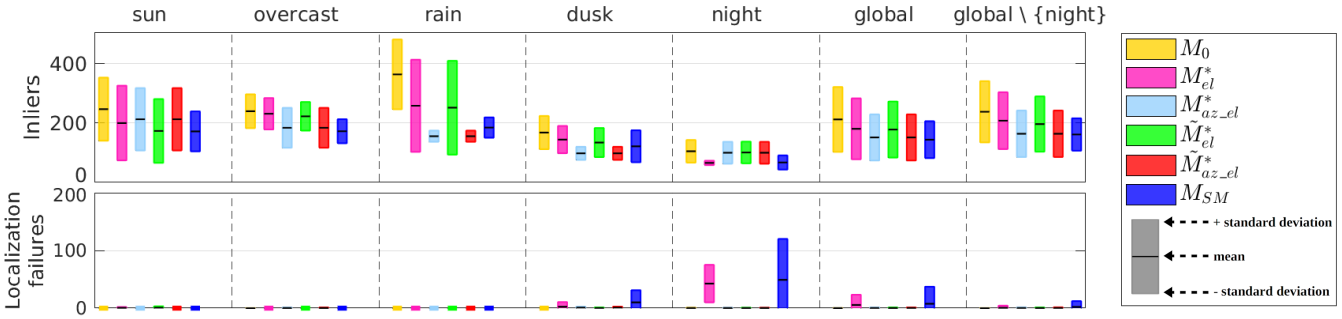
For each choice of $\hat{N}$, the size of the maps generated by our approach are very close since they incorporate the same number of traversals. To ensure a fair comparison with Muhlfellner *et al.*'s approach, we remove from the map M_{SM} the lowest scored landmarks until we get a map that has a similar size with the other maps. Figure 8 presents an overview of memory occupation and landmarks count of our



(a) $\hat{N} = 3$



(b) $\hat{N} = 4$



(c) $\hat{N} = 5$

Fig. 7: Localization performance comparison on M_0, M^*_{el}, $M^*_{az\_el}$, $\tilde{M}^*_{el}$, $\tilde{M}^*_{az\_el}$ and the map M_{SM} generated with the Summary Maps approach [19]. Each color refers to a map as indicated in the legend, and the boxes represent the mean +/- the standard deviation of inliers or localization failures on all the sequences of the corresponding class. Sub-figures (a), (b) and (c) represent the localization performance when choosing 3, 4 and 5 respectively as values for $\hat{N}$.

maps for each choice of $\hat{N}$.

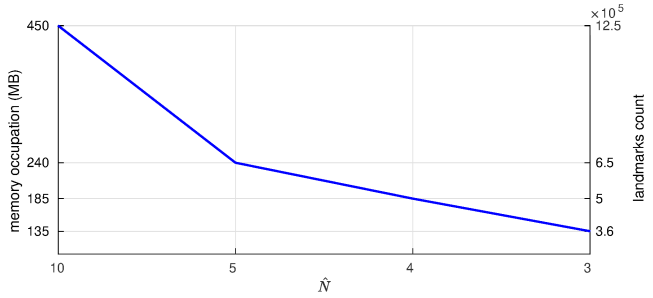


Fig. 8: A curve in which we inspect the size of M^*_{el}, $M^*_{az\_el}$, $\tilde{M}^*_{el}$, $\tilde{M}^*_{az\_el}$ and M_{SM}. For each value of $\hat{N}$, the curve is presenting the corresponding memory occupation (Megabytes) and the number of landmarks in each one of these maps (all the maps have approximately the same size). For $\hat{N} = 10$, we provide the size of the map M_0.

We evaluated and compared the localization performance in the 93 sequences when using the different maps produced (M_0, M^*_{el}, $M^*_{az\_el}$, $\tilde{M}^*_{el}$, $\tilde{M}^*_{az\_el}$ and M_{SM}) with $\hat{N} = \{3, 4, 5\}$. These sequences were manually classified into 5 different classes according to their corresponding environmental conditions: 13 sequences in "sun" class, 39 in "overcast", 12 in "rain", 17 in "dusk" and 11 in "night". The "global" class contains all 93 sequences while the "global \ {night}" class contains all the sequences excluding the night sequences (82 sequences).

For a better understanding of the results, we present in Table I the traversals included in each of the maps produced by our approach (M^*_{el}, $M^*_{az\_el}$, $\tilde{M}^*_{el}$ and $\tilde{M}^*_{az\_el}$). According to Figure 7, the localization performance has globally improved when we increased the value of $\hat{N}$. For all the choices of $\hat{N}$, the localization performance on the map M^*_{el} with night sequences was very poor since this map does not include any night traversals. This is not the case for $M^*_{az\_el}$ which incorporates a night traversal. Both $\hat{N}$-session maps $\tilde{M}^*_{el}$ and $\tilde{M}^*_{az\_el}$ also include a night traversal due to the imposed constraint explained in Section III-B. According to the figure, localization under overcast and rainy conditions can be reliable even if the map does not include traversals of the same class (e.g. for $\hat{N} = 3$, M^*_{el} contains a traversal from "rain" class while it is not the case for the other maps which is distinguishable by the inliers average. However, all of them have no localization failures except for M_{SM}).

By increasing the value of $\hat{N}$, the localization performance was improved in different classes for example on rainy sequences for $M^*_{az\_el}$ and $\tilde{M}^*_{az\_el}$ and on overcast sequences for M^*_{el}. Both maps $\tilde{M}^*_{el}$ and $\tilde{M}^*_{az\_el}$ have shown similar performances on different classes, and they have both managed to ensure a reliable localization without any localization failure.

The Summary Maps approach (M_{SM}) has shown a major weakness as we significantly decrease the size of the map. This weakness was more relevant with night sequences. This can be explained by the fact that the 10 mapping sequences include only one night sequence. Thus, landmarks of the night traversal will have a low score since they were seen only in one session. Consequently, these landmarks will be removed with the Summary Maps approach. In the opposite side, our approach has proved its capability of balancing experiences in the map. Therefore, it has managed to greatly decrease the size of the map while guaranteeing a reliable localization under different environmental conditions. According to Figure 7 and Figure 8, our approach was able to drastically reduce the size of the map (e.g. from 450MB to 185MB for $\hat{N} = 4$) while ensuring a reliable localization under different conditions.

V. Conclusion

We have introduced a map management approach which computes and uses solar coordinates information to classify and reduce the number of traversals in the map in order to ensure a lifelong navigation. In the experiments part, we have compared some different variants of our method in multiple environmental conditions. The experiments have demonstrated that with our approach we were able to achieve

TABLE I: Traversals included in each of the maps produced by our approach. The traversal names are colored to match the colors in the Figure 6.

map	M^*_{el}	$M^*_{az\_el}$	$\tilde{M}^*_{el}$	$\tilde{M}^*_{az\_el}$
$\hat{N} = 3$	2020-01-15-11-15-33 2019-10-02-15-03-40 2019-10-22-15-01-25	2019-10-02-15-03-40 2020-02-05-18-37-10 2020-01-22-10-22-06	2019-10-02-15-03-40 2020-02-05-18-37-10 2020-01-22-10-22-06	2019-10-02-15-03-40 2020-02-05-18-37-10 2020-01-22-10-22-06
$\hat{N} = 4$	2020-01-15-11-15-33 2019-10-02-15-03-40 2019-10-22-15-01-25 2020-02-05-17-53-21	2019-10-01-16-54-55 2020-02-05-18-37-10 2020-01-22-10-22-06 2020-01-22-10-22-06	2019-10-02-15-03-40 2020-02-05-18-37-10 2020-01-22-10-22-06 2020-01-22-10-22-06	2019-10-01-16-54-55 2020-02-05-18-37-10 2020-01-22-10-22-06 2020-01-22-10-22-06
$\hat{N} = 5$	2020-01-15-11-15-33 2019-10-02-15-03-40 2019-10-22-15-01-25 2020-02-05-17-53-21 2020-01-31-16-07-34	2019-10-02-15-03-40 2020-02-05-18-37-10 2020-01-22-10-22-06 2020-01-22-10-22-06 2020-01-31-16-07-34	2019-10-02-15-03-40 2019-10-22-15-01-25 2020-02-05-17-53-21 2020-02-05-18-37-10 2020-01-22-10-22-06	2019-10-02-15-03-40 2020-02-05-18-37-10 2020-01-22-10-22-06 2020-01-22-10-22-06 2020-01-31-16-07-34

long-term performance with compressed maps outperforming performance of localization of a state of the art approach.

In future works, we will concentrate on combining other weather information with solar coordinates for further improvements in our approach.

ACKNOWLEDGMENT

This work has been sponsored by the French government research program "Investissements d'Avenir" through the IMobS3 Laboratory of Excellence (ANR-10-LABX-16-01) and the RobotEx Equipment of Excellence (ANR-10-EQPX-44), by the European Union through the Regional Competitiveness and Employment program 2014-2020 (ERDF - AURA region) and by the AURA region.

REFERENCES

[1] E. Royer, F. Marmoiton, S. Alizon, D. Ramadasan, M. Slade, A. Nizard, M. Dhome, B. Thuilot, and F. Bonjean, "Lessons learned after more than 1000 km in an autonomous shuttle guided by vision," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 2248–2253.

[2] P. Lébraly, E. Royer, O. Ait-Aider, C. Deymier, and M. Dhome, "Fast calibration of embedded non-overlapping cameras," in *2011 IEEE international conference on robotics and automation*. IEEE, 2011, pp. 221–227.

[3] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[4] T. Schneider, M. Dymczyk, M. Fehr, K. Egger, S. Lynen, I. Gilitschenski, and R. Siegwart, "maplab: An open framework for research in visual-inertial mapping and localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1418–1425, 2018.

[5] C. G. Harris, M. Stephens *et al.*, "A combined corner and edge detector." in *Alvey vision conference*, vol. 15, no. 50. Citeseer, 1988, pp. 10–5244.

[6] A. C. Murillo and J. Kosecka, "Experiments in place recognition using gist panoramas," in *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*. IEEE, 2009, pp. 2196–2203.

[7] M. J. Milford and G. F. Wyeth, "Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 1643–1649.

[8] T. Naseer, G. L. Oliveira, T. Brox, and W. Burgard, "Semantics-aware visual localization under challenging perceptual conditions," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 2614–2620.

[9] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine: Robust hierarchical localization at large scale," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 716–12 725.

[10] N. Piasco, D. Sidibé, V. Gouet-Brunet, and C. Demonceaux, "Learning scene geometry for visual localization in challenging conditions," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 9094–9100.

[11] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "Lift: Learned invariant feature transform," in *European Conference on Computer Vision*. Springer, 2016, pp. 467–483.

[12] J. Diaz-Escobar, V. Kober, and J. A. Gonzalez-Fraga, "Luift: Luminance invariant feature transform," *Mathematical Problems in Engineering*, vol. 2018, pp. 1–17, 2018.

[13] Y. Tian, X. Yu, B. Fan, F. Wu, H. Heijnen, and V. Balntas, "Sosnet: Second order similarity regularization for local descriptor learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 016–11 025.

[14] S. Shoman, T. Mashita, A. Plopski, P. Ratsamee, and Y. Uranishi, "Real-to-synthetic feature transform for illumination invariant camera localization," *IEEE Computer Graphics and Applications*, 2020.

[15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[16] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*. Springer, 2006, pp. 404–417.

[17] M. Bürki, C. Cadena, I. Gilitschenski, R. Siegwart, and J. Nieto, "Appearance-based landmark selection for visual localization," *Journal of Field Robotics*, vol. 36, no. 6, pp. 1041–1073, 2019.

[18] K. MacTavish, M. Paton, and T. D. Barfoot, "Selective memory: Recalling relevant experience for long-term visual localization," *Journal of Field Robotics*, vol. 35, no. 8, pp. 1265–1292, 2018.

[19] P. Mühlfellner, M. Bürki, M. Bosse, W. Derendarz, R. Philippsen, and P. Furgale, "Summary maps for lifelong visual localization," *Journal of Field Robotics*, vol. 33, no. 5, pp. 561–590, 2016.

[20] M. Bürki, M. Dymczyk, I. Gilitschenski, C. Cadena, R. Siegwart, and J. Nieto, "Map management for efficient long-term visual localization in outdoor environments," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 682–688.

[21] J. J. Michalsky, "The astronomical almanac's algorithm for approximate solar position (1950–2050)," *Solar energy*, vol. 40, no. 3, pp. 227–235, 1988.

[22] H.-Y. Cheng, C.-C. Yu, K.-C. Hsu, C.-C. Chan, M.-H. Tseng, and C.-L. Lin, "Estimating solar irradiance on tilted surface with arbitrary orientations and tilt angles," *Energies*, vol. 12, no. 8, p. 1427, 2019.

[23] G. J. Szekely, M. L. Rizzo *et al.*, "Hierarchical clustering via joint between-within distances: Extending ward's minimum variance method," *Journal of classification*, vol. 22, no. 2, pp. 151–184, 2005.

[24] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000km: The oxford robotcar dataset," *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 1, pp. 3–15, 2017. [Online]. Available: http://dx.doi.org/10.1177/0278364916679498

[25] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of Michigan North Campus long-term vision and lidar dataset," *International Journal of Robotics Research*, vol. 35, no. 9, pp. 1023–1035, 2015.