Large scale vision-based navigation without an accurate global reconstruction*

Siniša Šegvić, Anthony Remazeilles, Albert Diosi and François Chaumette
IRISA/INRIA, Campus de Beaulieu, F-35042 Rennes Cedex, France
sinisa.segvic@tugraz.at albert.diosi@irisa.fr
anthony.remazeilles@cea.fr francois.chaumette@irisa.fr

Abstract

Autonomous cars will likely play an important role in the future. A vision system designed to support outdoor navigation for such vehicles has to deal with large dynamic environments, changing imaging conditions, and temporary occlusions by other moving objects. This paper presents a novel appearance-based navigation framework relying on a single perspective vision sensor, which is aimed towards resolving of the above issues. The solution is based on a hierarchical environment representation created during a teaching stage, when the robot is controlled by a human operator. At the top level, the representation contains a graph of key-images with extracted 2D features enabling a robust navigation by visual servoing. The information stored at the bottom level enables to efficiently predict the locations of the features which are currently not visible, and eventually (re-)start their tracking. The outstanding property of the proposed framework is that it enables robust and scalable navigation without requiring a globally consistent map, even in interconnected environments. This result has been confirmed by realistic off-line experiments and successful real-time navigation trials in public urban areas.

1. Introduction

The design of an autonomous mobile robot requires establishing a close relation between the perceived environment and the commands sent to the low-level controller. This necessitates complex spatial reasoning relying on some kind of internal environment representation [5]. In the mainstream *model-based* approach, a monolithic environment-centred representation is used to store the landmarks and the descriptions of the corresponding image features. The considered features are usually geometric primitives, while their positions are expressed in coordinates of the common environment-wide frame [2, 16]. During the navigation, the detected features are associated with the elements of the model, in order to localize the robot and to effectively search for new model elements. However, the quality of the obtained results depends directly on the precision of the underlying model. This poses a strong assumption which impairs the scalability and, depending on the input, may not be attainable at all.

The alternative *appearance-based* approach employs a sensor-centred representation of the environment, which is usually a multidimensional array of sensor readings. In the context of computer vision, the representation usually contains a set of *key-images* which are acquired during a learning stage and organized within a graph [6]. Nodes of the graph correspond to key-images, while the arcs link the images containing a distinctive set of common landmarks. This is illustrated in Figure 1. The navigation between two neighbouring nodes is performed using well developed techniques from the field of mobile robot control [17]. Different types of landmark representations have been considered in the literature, from the integral contents of a considered image [11] and global image descriptors [6], to more conventional point features such as Harris corners [2, 3]. We consider the latter feature-oriented approach, in which the next intermediate key-image is reached by tracking the feature correspondences from the previous key-image. Recognition of new landmarks is a critical is-

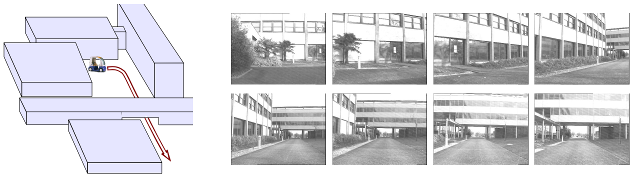


Figure 1. Appearance-based navigation: the sketch of a navigation task (left), and the set of first eight images from the environment representation forming a linear graph (right). Note that the graph has been constructed automatically, as described in 2.1.

*This work has been supported by the French national project Predit Mobivip, by the project Robea Bodega, and by the European MC IIF project AViCMaL.

sue in this approach, since it implies a risk of introducing an association error. Predicting approximate locations of currently invisible features (*feature prediction*) is therefore an essential capability in feature-oriented appearance-based navigation.

In this paper, a novel vision framework for scalable mapping and localization is presented, enabling robust appearance-based navigation in large outdoor environments. We consider separate mapping and navigation procedures as an interesting and not completely solved problem, despite the ongoing work on a unified solution [4]. The proposed framework employs a hybrid hierarchical environment representation [6, 1], with a graph of key-images at the top, and local 3D reconstructions at the bottom. The global topological representation ensures an outstanding scalability, limits the propagation of association errors, simplifies consistency management in interconnected environments, and enables appearance-based navigation. On the other hand, the bottom-level geometric models enable feature prediction by multi-view geometry techniques. The viability of the approach has been confirmed by successful experiments in real-time robot control. The results clearly demonstrate that a globally consistent 3D reconstruction is not required for large-scale navigation to be successful: we consider that as the most important contribution of this work.

An appearance-based navigation approach with feature prediction has been described in [8]. Simplifying assumptions with respect to the motion of the robot have been used, while the prediction was implemented using intersection of the two epipolar lines, which has important limitations [9]. The need for feature prediction has been alleviated in [3], where the points from the next key-image are introduced using wide-baseline matching [12]. A similar approach has been proposed in the context of omnidirectional vision [7]. In this closely related work, feature prediction based on point transfer [9] has been employed to recover from tracking failures, but not to introduce previously unseen features as well. However, introduction of new features by wide-baseline matching [7, 3] implies a great potential for introducing association errors caused by ambiguous landmarks. Our experiments have shown that considerably better results are obtained by optimizing the new feature location starting from a prediction obtained by point transfer.

In comparison with model-based navigation approaches such as the one described in [16], our approach does not require a global consistency. By posing weaker requirements, we increase the robustness of the mapping phase, likely obtain better local consistencies, can close loops regardless of the extent of the accumulated drift and have better chances to survive correspondence errors. Notable advances in prediction of feature positions have been achieved in model-based SLAM [4]. Nevertheless, current implementations have limitations with respect to the number of mapped points, so that a prior learning step still seems a necessity in realistic navigation tasks. Our approach has no scaling problems: experiments with 15000 landmarks have been performed without any performance degradation.

The paper is organized as follows. The details of the proposed framework for mapping and localization are described in Section 2. Section 3 provides the experimental results, while the conclusion is given in Section 4.

2. Scalable mapping and localization

This section briefly describes the two high-level components of the proposed vision framework for appearance-based navigation. The mapping component extracts point features from the learning sequence acquired along a desired possibly circular physical path. During the navigation, the localization component tracks the mapped features and employs them to locate new features. Both components rely on a multi-scale differential tracker with warp correction and checking towards the reference appearance [19]. The employed warp includes isotropic scaling and affine contrast compensation [18]. The output of the framework is a set of 2D vectors connecting the current features with their corresponding locations in the next key-image. These vectors are finally used to support appearance-based navigation based on visual servoing.

2.1. The mapping component

The mapping component constructs the environment graph and annotates its nodes and arcs with geometric information. Here we consider linear and circular graphs, while the work on complex topologies [15] will be integrated in the future. The nodes of the graph are formed by choosing the corresponding key-images I_i. The same indexing is used for arcs as well, by defining that arc i connects nodes $i-1$ and i. If the graph is circular, arc 0 connects the last node $n-1$ with the node 0. Each node is assigned the set X_i of features from I_i, denoted by distinctive identifiers. Each arc is assigned an array of identifiers M_i denoting landmarks located in the two incident key-images. As shown in Figure 2, arcs are finally annotated with two-view geometries W_i, recovered from M_i by random sampling with the five-point algorithm [13] as the hypothesis generator.

The elements of W_i include motion parameters $\mathbf{R_i}$ and $\mathbf{t_i}$ ($|\mathbf{t_i}| = 1$), as well as the metric landmark reconstructions Q_i. The two-view geometries W_i are deliberately *not* put into an environment-wide frame, since contradicting scale sequences may be obtained along the graph cycles. The scale ratio s_i between the incident geometries W_i and W_{i+1} is therefore stored in the common node i. Note that each neighbouring pair of geometries W_{i+1} and W_{i+2} needs to have some features in common, $M_{i+1} \cap M_{i+2} \neq \varnothing$, in order to enable the transfer of features from the next two key-images (I_{i+1}, I_{i+2}) on the path (see 2.2.2 for details).

Many maps can be constructed for the same motion of the robot in the learning phase, depending on the selected set of key-images and on the technique for extracting correspondences. Quantitatively, a particular arc of the map can be evaluated by an estimate of the reprojection error [9] $\sigma(W_i)$, and the number of correspondences $|M_i|$. These parameters are respectively related with accuracy of the point transfer and robustness to interferences (occlusions, illumination variations). There is a trade-off in interpreting the criterion $|M_i|$, since more points usually means better robustness but lower execution speed. Different maps of the same environment can be evaluated by the total count of arcs in the graph $|\{M_i\}|$, and by the parameters of the individual arcs $\sigma(W_i)$ and $|M_i|$. It is usually favourable to have less arcs, since that ensures a smaller difference in lines of sight between the relevant key-images and the images acquired during navigation. This is important since the ability to deviate from the reference path enables the robot to tolerate control errors and to avoid detected obstacles.

The devised mapping solution uses the tracker to find the stablest point features in a given subrange of the learning sequence. The tracker is initiated with all Harris points in the initial frame of the subrange. The features are tracked until the reconstruction error between the first and the current frame of the subrange rises above a predefined threshold σ. At this moment the current frame is discarded, while the previous frame is registered as the new node of the graph, and the whole procedure is repeated from there. The above is similar to visual odometry [14], except that we employ larger feature windows and more involved tracking in order to achieve more distinctive features and longer feature lifetimes. To ensure a minimum number of features within an arc of the graph, a new node is forced when the absolute number of tracked points falls below n. Bad tracks are identified by a threshold R on RMS residual between the current feature and the reference appearance [19, 18]. Typically, the following values were used: $\sigma = 4$, $n = 50$, $R = 6$.

The above basic mapping scheme provides substantially better results than the approach [7] based on wide-baseline matching with state-of-the-art algorithms [12]. This should be regarded as no surprise, since more information is used to achieve the same goal. However, exceptions to the above occur when there are discontinuities in the learning sequence caused by a large moving object, or a "frame gap" due to preemption of the acquisition process. In the presented scheme, such events are reflected by a general tracking failure in the *second* frame of a new subrange. A recovery is consequently attempted by matching the last key-image with the current image. This is especially convenient when the mapping is performed online, from a manually controlled robotic car.

Wide-baseline matching is also useful for connecting a cycle in the environment graph, which occurs if the learning sequence is acquired along a closed physical path. After the learning sequence acquisition is over, the first and the last key-image are subjected to matching: a circular graph is created on success, and a simple linear graph otherwise. Note that in case of a monolithic geometric model, the above loop closing process would need to be followed by a sophisticated map correction procedure, in order to try to correct the accumulated error. Due to topological representation at the top-level, this operation proceeds reliably and smoothly, regardless of the extent of the drift.

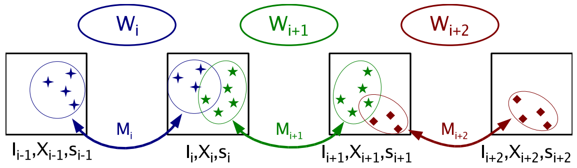


Figure 2. The linear environment graph. Nodes contain images I_i, extracted features X_i and scale factors s_i. Arcs contain match arrays M_i and the two-view geometries W_i.

2.2. The localization component

In the feature-oriented appearance-based navigation, two distinct kinds of localization are required: (i) explicit topological localization, and (ii) implicit fine-level localization through the locations of the tracked landmarks. Topological location corresponds to the arc of the environment graph incident to the two key-images having most content in common with the current image. This is usually well defined in practice since the motion of a robotic car is constrained by the traffic infrastructure. Maintaining an accurate topological location is extremely important since that defines the landmarks which are currently considered for tracking. In the proposed framework, the tracked features belong either to the *actual* arc (topological location), or the two neighbouring arcs as illustrated in Figure 3.

In this paper, we focus on the on-line facets of the localization problem: (i) robust fine-level localization relying on feature prediction, and (ii) maintenance of the topological location as the navigation proceeds. However, for completeness, we first present a minimalistic initialization procedure used in the experiments.

2.2.1 The initialization procedure

The navigation program is started with the following parameters: (i) map of the environment (ii) initial topological location of the robot (index of the actual arc) (iii) calibration parameters of the attached camera. The execution starts with wide-baseline matching of the current image with the two key-images incident to the actual arc. From the obtained correspondences, the pose is recovered in the actual geometric frame, allowing to project the mapped features and to bootstrap the processing loop. Note that automatic initialization using content based image retrieval is feasible.

2.2.2 Feature prediction and tracking resumption

The point features which are tracked in the current image I_t are employed to estimate the current two-view geometries $W_{t:i}(I_i, I_t)$ and $W_{t:i+1}(I_{i+1}, I_t)$ towards the two incident key-images, using the same procedure as in 2.1. An accurate and efficient recovery of the three-view geometry is devised by a decomposed approach [10] in the calibrated context. The approach relies on recovering the relative scale between the two independently recovered metric frames, by enforcing the consistency of the common structure. The main advantages with respect to the "golden standard" method [9] are the utilization of pairwise correspondences (which is of particular interest for forward motion), and real-time performance. Thus, the three-view geometry (I_t, I_i, I_{i+1}) is recovered by adjusting the precomputed two-view geometry W_{i+1} towards the more accurate (in terms of reprojection error) of $W_{t:i}$ and $W_{t:i+1}$ (see Figure 3). The geometry (I_t, I_{i+1}, I_{i+2}) is recovered from W_{i+2} and $W_{t:i+1}$, while (I_t, I_{i-1}, I_i) is recovered from W_i and $W_{t:i}$. Current image locations of landmarks mapped in the actual arc $i+1$ are predicted by the geometry (I_t, I_i, I_{i+1}). Landmarks from the previous arc i and the next arc $i+2$ are transferred by geometries (I_t, I_{i-1}, I_i) and (I_t, I_{i+1}, I_{i+2}), respectively.

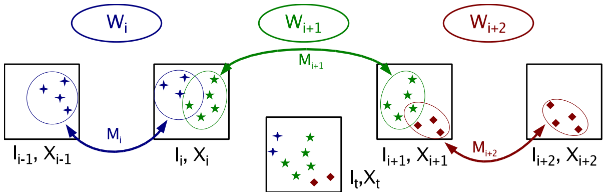


Figure 3. The current image I_t and the three groups of features considered for tracking when the topological location is $i+1$. The notation is explained in Figure 2. See text for more details.

In any case, the prediction by point transfer is performed only if the estimated reprojection error of the employed current geometry is within the safety limits. The obtained predictions are refined (or rejected) by minimizing the residual between the warped current feature and the reference appearance. As in tracking, the result is accepted if the procedure converges near the predicted location, with an acceptable residual. The above procedure is also employed to check the consistency of the tracked features, which occasionally "jump" to the occluding foreground. Thus, following the sanity check on the employed two-view geometry, the tracking of a feature is discontinued if the tracked position becomes too distant from the prediction.

2.2.3 Maintaining the topological location

Maintaining a correct topological location is critical since both feature prediction and robot control depend on its accuracy. This is especially the case in sharp turns where the tracked features die quickly due to the contact with the image border. An incorrect topological location implies a suboptimal introduction of new features and may be followed by a failure due to insufficient features for calculating $W_{t:i}$ and $W_{t:i+1}$, and performing the prediction.

Best results have been obtained using a straightforward geometric criterion: a forward transition is taken when the camera pose in the actual geometric frame W_{i+1} is in front of the farther camera I_{i+1}. This can be expressed as:

$$\langle -\mathbf{R_{i+1}}^\top \cdot \mathbf{t_{i+1}}, \mathbf{t_{t:i+1}} \rangle < 0 \, . \qquad (1)$$

The decision is based on the current geometry related to the next key-image $W_{t:i+1}$, which is geometrically closer to the hypothesized transition, as shown in Figure 4. As before, the above is cancelled if the estimated reprojection error of the employed current geometry is not within the safety limits. Note that backwards transitions can be analogously defined in order to support reverse motion of the robot.

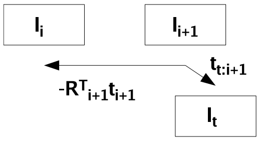


Figure 4. Condition for changing the topological location.

After each change of the topological location, the reference appearances (*references*) are redefined for all relevant features in order to achieve better tracking. For a forward transition, references for the features from the actual geometry W_{i+1} are taken in I_{i+1}, while the references for the features from W_{i+2} are taken in I_{i+2} (see Figure 3). Previously tracked points from geometries W_{i+1} and W_{i+2} are instantly resumed using their previous positions and new references while the features from W_i are discontinued.

3. Experimental results

The experiments have been carried out on sequences taken from the robotic car and in real-time, during navigation. The experiments are organized in three groups, involving mapping, off-line localization, and navigation (real-time localization with robot control).

3.1. Mapping experiments

We first present quantitative mapping results obtained on the learning sequence `ifsic5`, corresponding to the reverse of the path shown in Figure 1. The selected set of key-images is presented in Figure 5.

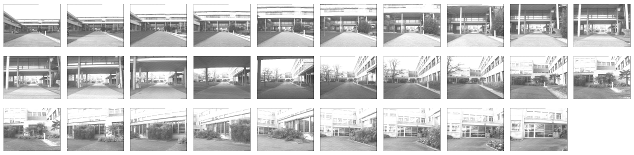
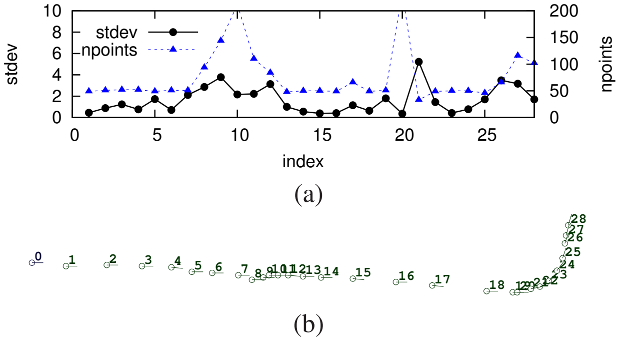


Figure 5. Key-images from the map of the sequence `ifsic5`. The sequence contains 1900 images, acquired along a 150 m path. The images can be enlarged within the pdf document of the article.

The analysis was performed in terms of the parameters of individual geometric models, which were introduced in 2.1. These parameters are (i) the number of point features (more is better), (ii) the reprojection error (less is better), and (iii) the inter-node distance (more is better). Figure 6(a) shows the variation of the first two parameters along the arcs of the created environment graph. A qualitative illustration of the third parameter (inter-node distance) is presented in Figure 6 as the sequence of recovered camera poses corresponding to the nodes of the environment graph.

In order to achieve a uniform representation, all geometric models were put into the common metric frame of the first geometry W_1. The figure suggests that the mapping component adapts the density of key-images to the inherent difficulty of the scene. The dense nodes 7-14 correspond to the first difficult moment of the learning path: approaching the traverse building and passing underneath it. Nodes 20 to 25 correspond to the sharp left turn, while passing very close to a building. The hard conditions persisted after the turn due to large featureless bushes and a reflecting glass surface (see Figure 5, bottom row), which is reflected in dense nodes 26-28. The number of features in arc 20 is exceptionally high, while the incident nodes 19 and 20 are very close. The anomaly is due to a large frame gap causing most feature tracks to terminate instantly. Wide-baseline matching succeeded to relate the key-image 19 and its immediate successor which consequently became key-image 20. The error peak in arc 21 is caused by an another gap which has been successfully bridged by the tracker alone.

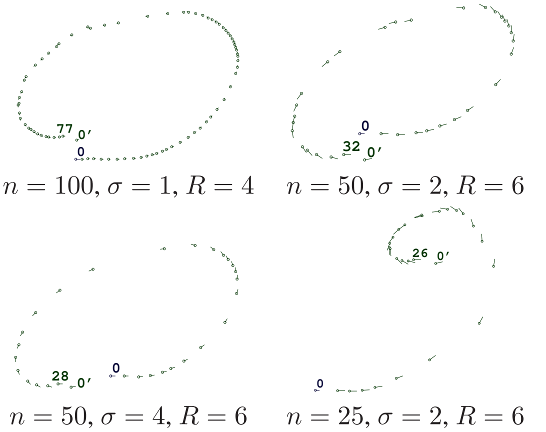


(a)

(b)

Figure 6. Counts of mapped point features and reprojection errors (a), and sequence of camera poses corresponding to 28 arcs of the environment graph obtained from the sequence `ifsic5` (b).

In the second group of experiments, we consider the learning sequence `loop-clouds`, taken along a *circular* path of approximately 50 m. Circular sequences are especially suitable for testing the mapping alternatives since they provide an intuitive notion about the achieved overall accuracy. We investigate the sensitivity of the mapping algorithm with respect to the three main parameters described in 2.1: (i) minimum count of features n, (ii) maximum allowed reprojection error σ, and (iii) the RMS residual threshold R. The resulting poses have been plotted in Figure 7 for 4 different parameter triples.



$n = 100, \sigma = 1, R = 4$ $n = 50, \sigma = 2, R = 6$

$n = 50, \sigma = 4, R = 6$ $n = 25, \sigma = 2, R = 6$

Figure 7. Poses from the maps obtained on input sequence `loop-clouds`, by employing different mapping parameters.

Reasonable and usable representations have been obtained in all cases, despite the smooth planar surfaces and vegetation which are visible in Figure 8. The presence of node 0' indicates that the cycle at the topological level has been successfully closed by wide-baseline matching. Ideally, nodes 0' and 0 should be very close; the extent of the distance indicates the magnitude of the error due to the accumulated drift. The relations between the two nodes in the first three results in Figure 7 suggest that the distance between the corresponding locations is around 1.5 m. The last map in Figure 7 (bottom-right) was deliberately constructed using suboptimal parameters, to show that our navigation approach essentially works even when the global consistency is difficult to enforce. The navigation can smoothly proceed despite a discontinuity in the global geometric re-

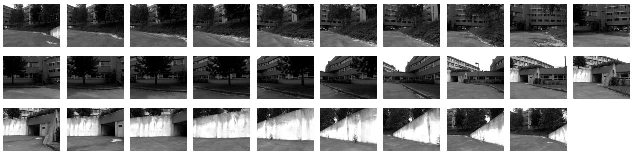


Figure 8. Key-images from the map obtained on the sequence `loop-clouds`, with $n = 50$, $\sigma = 4$, $R = 6$. The images can be enlarged within the pdf document of the article.

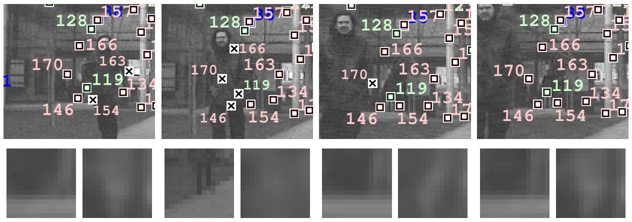
construction, since the local geometries are "elastically" glued together by the continuous topological representation.

The experiments show that there is a direct coupling between the number of arcs $|\{M_i\}|$, and the number of features in each arc $|M_i|$. Thus, it is beneficial to seek the smallest $|\{M_i\}|$ ensuring acceptable values for $\sigma(W_i)$ and $|M_i|$. The requirement that neighbouring triples of images need to contain common features did not cause problems in practice: the accuracy of the two-view geometries $\sigma(W_i)$ was the main limiting factor for the mapping success.

In some cases, a more precise overall geometric picture might have been obtained by applying a global optimization post-processing step. This has been omitted since, in the context of appearance-based navigation, global consistency brings no immediate benefits and poses scalability problems. Enforcing the global consistency is especially fragile for forward motion which occurs predominantly in the case of non-holonomic robotic cars. In this context, more than half of the correspondences are *not* shared between neighbouring geometries, and the ones that are shared are more likely to contain association errors due to a larger change in appearance.

3.2. Localization experiments

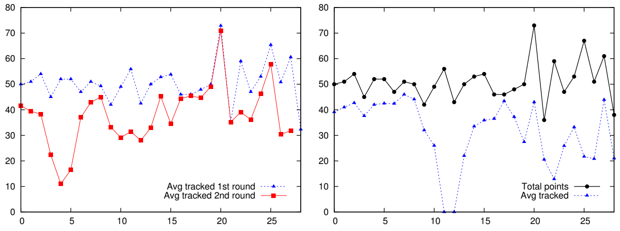
We first illustrate the capability of the localization component to resume temporary occluded and previously unseen features. Figure 9 shows the results of feature tracking within the localization component. The employed map has been illustrated in Figure 6 and discussed in the accompanying text. The figure shows a situation in which six features have been wiped out by a moving pedestrian, and subsequently resumed without errors. In the figure, the rejected predictions are designated with crosses: notice that they are near to where the corresponding landmarks would have been projected had they not been occluded. In the case of feature 146 in frame 743, the tracker "zoomed out" so that the legs of the occluding person are aligned with the edge of the tracked corner. Feature 170 has been found in the same frame by "zooming in" onto a detail on the jacket. Both findings were rejected due to a large residual towards the reference appearance. The danger of introducing an association error while searching for an occluded feature can not be completely avoided, but is largely suppressed within a conservatively configured tracker with warp correction and checking [18] (large features, low RMS threshold R).



frame 723 frame 743 frame 758 frame 762

Figure 9. Re-introducing disoccluded landmarks: tracked features and rejected projections are designated with squares and crosses, respectively. The bottom row shows the references and optimized warps for the features 146 (left) and 170 (right).

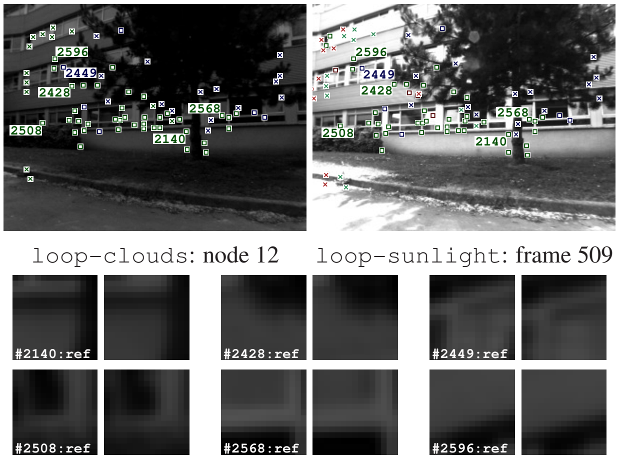
The capability of the localization component to traverse a topological cycle created by the mapper was tested on a sequence obtained during two rounds roughly along the same circular physical path. This is a quite difficult scenario since it requires continuous and fast introduction of new features due to persistent changes of viewing direction. The first round was used for mapping (this is the sequence `loop-clouds`, discussed in Figures 7 and 8), while the localization is performed along the combined sequence, involving two complete rounds. During the acquisition, the robot was manually driven so that the two trajectories were more than 1 m apart at several occasions during the experiment. Nevertheless, the localization was successful in both rounds, as summarised in Figure 10(a). All features have been successfully located during the first round, while the outcome in the second round depends on the extent of the distance between the two trajectories.



(a) (b)

Figure 10. Average counts of tracked features on the map shown in Figure 8, while processing the sequences (a) `loop-clouds` (two rounds), and (b) `loop-sunlight` (one round).

The map built from the sequence `loop-clouds` has also been tested on the sequence `loop-sunlight`, acquired along a similar circular path in bright sunlight. The imaging conditions during the acquisition of the two sequences were considerably different, which can be seen in Figure 11. Nevertheless, the localization component successfully tracked enough mapped features, except in arcs 10, 11 and 12 as shown in Figure 10(b). The recovered geometries in arc 10 were too uncertain so that the switching towards arc 11 did not occur at all, resulting in zero points tracked in arcs 11 and 12. The two factors amplifying the effects of feature decimation due to different illumination were a tree covering most of the field of view, and a considerable curvature of the learning path (see Figures 8 and 11). The localization component was re-initialized by wide-baseline matching using the key-images incident to the arc 13, where the buildings behind the tree begin to be visible. Figure 11 shows the processing results immediately after the reinitialization, within arc 13. The figure shows that there is



`loop-clouds`: node 12 `loop-sunlight`: frame 509

the 6 designated features

Figure 11. Results at 509th frame of `loop-sunlight` using the map obtained on `loop-clouds`. The same notation is used as in Figure 9. The bottom part shows references and warped current appearances for the six features designated in the upper part.

a big potential for association errors since many prominent landmarks are ambiguous due to structural regularity typical for man-made environments. The framework deals successfully with such ambiguities, since good predictions of invisible feature positions are provided by point transfer.

3.3. The navigation experiments

The proposed framework performed well in navigation experiments featuring real-time control of the robotic car. A simple visual servoing scheme was employed, in which the steering angle ψ is determined from average x components of the current feature locations $(x_t, y_t) \in X_t$, and their correspondences in the next key-image $(x^*, y^*) \in X_{i+1}$.

$$\psi = -\lambda\,(\overline{x}_t - \overline{x}^*)\,, \quad \text{where} \quad \lambda \in \mathcal{R}^+\,. \tag{2}$$

We present an experiment carried out along an 1.1 km reference path, offering a variety of driving conditions including narrow sections, slopes and driving under a building. In order to accomodate the control frequency of 1 Hz, the navigation speed was set to 30 cm/s in turns, and otherwise 80 cm/s. The map was built by the procedure described in 2.1, on a learning sequence acquired under manual control. The compound appearance-navigation system performed in a way that only five human interventions were required, at locations shown in Figure 12. Between the points A and B the robot smoothly drove over 740 m despite a passing car occluding the majority of the features, as shown in Figure 13. Several similar encounters with pedestrians have been dealt with in a graceful manner too. The system succeeded to map features (and subsequently find them) in seemingly featureless areas where the road and the grass occupied most of the field of view. The reasons for the five interventions were (i) failures within the localization component due to unsuccessful maintenance of the topological location in turns (A, B and D), and (ii) prevention of a curb contact due to an extremely narrow section of the road (E) and a tendency of the control law (2) to "cut the corners" (C).

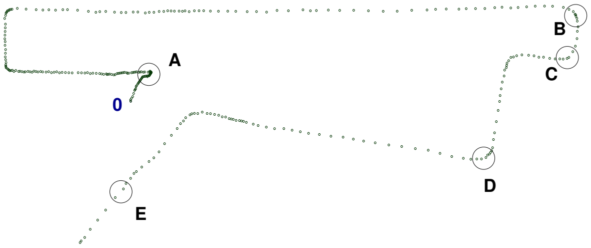


Figure 12. The graph of 320 nodes mapping an 1.1 km reference path. Large circles mark places where a human intervention was necessary. The distance between A and B is approximately 740 m.

The environment representation shown in Figure 12 is quite inaccurate from the global point of view. The beginning and the final node of the graph correspond to the same physical location, but this is not the case in the figure due to evident deviations in shape and scale. Nevertheless, the experimental system succeeds to perform large autonomous displacements, while also being robust to other moving objects. We consider this as a strong indication of the potential

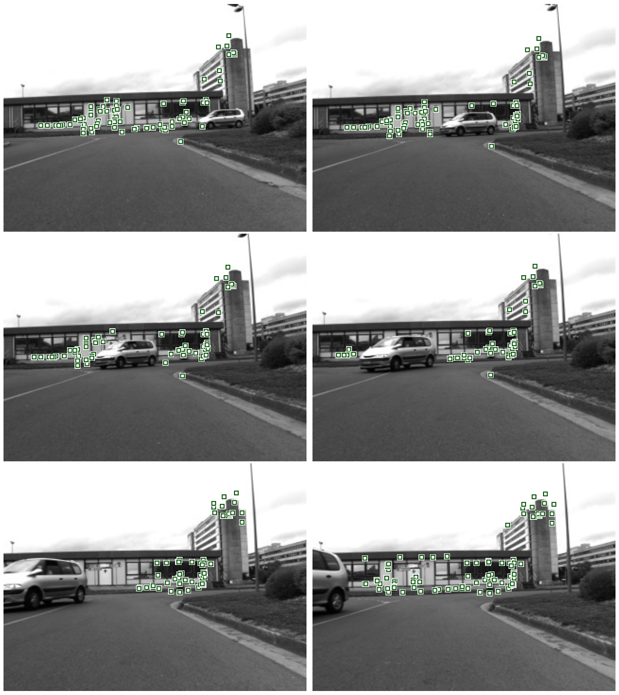


Figure 13. Sequence of images obtained during the execution of a navigation experiment. The points used for navigation re-appear after being occluded and disoccluded by a moving car.

of the proposed framework towards real applications of autonomous vehicles in the near future.

4. Conclusion

We described a novel framework for large-scale mapping and localization, based on point features mapped during a learning session. The purpose of the framework is to provide 2D image measurements for appearance-based navigation. The tracking of temporarily occluded and previously unseen features can be (re-)started on-the-fly due to feature prediction based on point transfer. 2D navigation and 3D prediction smoothly interact through a hybrid hierarchical environment representation. The navigation is concerned with the upper topological level, while the prediction is performed within the lower, geometrical level.

In comparison with the mainstream approach involving a monolithic geometric representation, the proposed framework enables robust large-scale navigation without requiring a geometrically consistent global view of the environment. This point has been demonstrated in the experiment with a circular path, in which the navigation bridges the first and the last node of the topology regardless of the extent of the accumulated error in the global 3D reconstruction. Thus, the proposed framework is applicable even in interconnected environments, where a global consistency may be difficult to enforce.

The localization component requires imaging and navigation conditions such that enough of the mapped landmarks have recognizable appearances in the acquired current images. The performed experiments suggest that this can be achieved even with very small images, for moderate-to-large changes in imaging conditions. The difficult situations include featureless areas (smooth buildings, vegetation, pavement), photometric variations (strong shadows and reflections), and the deviations from the reference path used to perform the mapping, due to control errors or obstacle avoidance. In the spirit of active vision, the last problem will be addressed within the control domain.

In our recent implementation, the mapping and localization throughput on 320×240 gray–level images is 5 Hz and 7 Hz, respectively, using a notebook computer with a CPU performance roughly equivalent to a Pentium 4 at 2GHz. Most of the processing time is spent within the point feature tracker, which uses a three-level image pyramid in order to be able to deal with large feature motion in turns. The computational complexity is an important issue: with more processing power we could deal with larger images and map more features, which would result in even greater robustness. Nevertheless, encouraging results in real-time autonomous robot control have been obtained even on very small images. In the light of future increase in processing performance, this suggests that the time of vision-based autonomous transportation systems is getting close.

References

[1] M. Bosse, P. Newman, J. Leonard, M. Soika, W. Feiten, and S. Teller. An ATLAS framework for scalable mapping. In *Proc. of ICRA*, pages 1899–1906, Taiwan, Sept. 2003. 2

[2] D. Burschka and G. D. Hager. Vision-based control of mobile robots. In *Proc. of ICRA*, pages 1707–1713, Seoul, South Korea, May 2001. 1

[3] Z. Chen and S. T. Birchfield. Qualitative vision-based mobile robot navigation. In *Proc. of ICRA*, pages 2686–2692, Orlando, Florida, May 2006. 1, 2

[4] A. Davison. Real-time simultaneous localisation and mapping with a single camera. In *Proc. of ICCV*, pages 1403–1410, Nice, France, Oct. 2003. 2

[5] G. N. DeSouza and A. C. Kak. Vision for mobile robot navigation: a survey. *IEEE Trans. PAMI*, 24(2), Feb. 2002. 1

[6] J. Gaspar and J. Santos-Victor. Vision-based navigation and environmental representations with an omni-directionnal camera. *IEEE Trans. RA*, 16(6):890–898, 2000. 1, 2

[7] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. V. Gool. Omnidirectional vision based topological navigation. *Int. J. Comput. Vis.*, 2007. to appear. 2, 3

[8] G. D. Hager, D. J. Kriegman, A. S. Georghiades, and O. Ben-Shalar. Toward domain-independent navigation: dynamic vision and control. In *Proc. of ICDC*, pages 1040–1046, Tampa, Florida, Dec. 1998. 2

[9] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, UK, 2004. 2, 3, 4

[10] M. Lourakis and A. Argyros. Fast trifocal tensor estimation using virtual parallax. In *Proc. of ICIP*, pages 169–172, Genoa, Italy, June 2005. 4

[11] Y. Matsumoto, M. Inaba, and H. Inoue. Exploration and navigation in corridor environment based on omni-view sequence. In *Proc. of IROS*, pages 1505–1510, Takamatsu, Japan, Oct. 2000. 1

[12] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *Int. J. Comput. Vis.*, 60(1):63–86, 2004. 2, 3

[13] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Trans. PAMI*, 26(6):756–770, 2004. 2

[14] D. Nistér, O. Naroditsky, and J. Bergen. Visual odometry. In *Proc. of CVPR*, pages 652–659, Washington, DC, 2004. 3

[15] A. Remazeilles, F. Chaumette, and P. Gros. 3D navigation based on a visual memory. In *Proc. of ICRA*, pages 2719–2725, Orlando, Florida, May 2006. 2

[16] E. Royer, M. Lhuillier, M. Dhome, and T. Chateau. Localization in urban environments: Monocular vision compared to a differential GPS sensor. In *Proc. of CVPR*, volume 2, pages 114–121, Washington, DC, 2005. 1, 2

[17] C. Samson. Control of chained systems: application to path following and time-varying point stabilization. *IEEE Trans. AC*, 40(1):64–77, 1995. 1

[18] S. Šegvić, A. Remazeilles, and F. Chaumette. Enhancing the point feature tracker by adaptive modelling of the feature support. In *Proc. of ECCV*, pp. 112–124, Graz, 2006. 2, 3, 6

[19] J. Shi and C. Tomasi. Good features to track. In *Proc. of CVPR*, pages 593–600, Seattle, Washington, June 1994. 2, 3