GRAPH BASED LOCALISATION REFINEMENT BY ORBITAL IMAGES

Evangelos Boukas[1], **Ioannis Kostavelis**[1], **Lazaros Nalpantidis**[2], **and Antonios Gasteratos**[1]

[1]*Laboratory of Robotics and Automation*
Department of Production and Management Engineering
Democritus University of Thrace, Xanthi, Greece, 67100
{*gkostave,evanbouk,agaster*}*@pme.duth.gr*

[2]*Computer Vision and Active Perception Lab.,*
Centre for Autonomous Systems,
Royal Institute of Technology - KTH,
SE-100 44 Stockholm, Sweden
lanalpa@kth.se

ABSTRACT

The paper in hand proposes a localization algorithm, where refinements in the robot's trajectory take place by exploiting the orbital images that cover the same area as a surface exploratory robot. It makes use of elementary graph theory terms in order to compare the 3D reconstructed area with the respective satellite image by examining the spatial distribution of the salient landmarks between the two different views. The utilized dissimilarity metric is the Graph Edit Distance (GED), which compares the two views and defines wether improvements in global orientation and position of the robot should be done. Once there is an indication for improvement the Iterative Closest Point (ICP) algorithm is used to refine the position of the robot backwards to the last executed orbital refinement. The proposed method is evaluated in unstructured non-urban scenes, where canonical formations are not available, as it is the case of space environments.

Key words: orbital imaging, visual SLAM, visual odometry, MST Sub-graph, ICP.

1. INTRODUCTION

To effectively navigate in their operation environments and accurately reach their target location, planetary robots require reliable self-localisation abilities. The use of vision sensors is a common choice for space exploration [KBN+11], where a lot of research has been conducted [Doc]. The problem of a mobile robot learning a map has been intensively studied in the past and is usually cited as the Simultaneous Localization and Mapping (SLAM) problem. However, the existing solutions to the SLAM problem typically rely on either sparse bundle adjustment, or loop-closure refinement steps as described in [SKLP06] and [NH05] respectively, in order to obtain global consistency and do not exploit any relevant information, even if it is available. In recent years an abundant of research methods have been proposed in the field of mobile robot navigation, taking advantage of prior knowledge of the robot's environment, which is then fussed with the results of SLAM to obtain more accurate and robust localization results. This extra piece of information might derive from aerial or satellite images. Compared to the standard SLAM approaches, the use of a global external information enables these techniques to provide more accurate solutions by delimiting the error when visiting new regions for the first time. Contrary to the traditional localisation techniques, those that

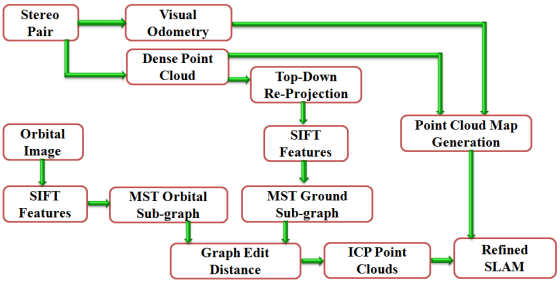


Figure 1. Block diagram of the proposed method.

include orbital imagery can substitute auxiliary sensors, e.g. GPS, in environments where such sensors are not operable, such as on a Lunar or on a Martian surface.

More analytically, the authors in [KSD+11] match the frames captured by 3D laser scanners with aerial images that have been acquired from a viewpoint significantly different from the one of the robot. The 3D-scans are top-down re-projected in order to be consistent with the visible territory in the reference map. An edge detection preprocessing step is utilized to extract the feature points from the aerial image, while the matching is accomplished by detecting structures in the 3D scan, which potentially correspond to intensity variations in the preprocessed aerial image. Then the correspondences are added as constraints in a graph based formulation of the SLAM problem, providing highly accurate solutions. Additionally, the method described in [KR04] utilizes image processing techniques to detect roads on aerial images. In this work, Gabor filters were chosen, which are used for texture analysis, due to the fact that the employed aerial images are of low contrast and, thus, characterized by excessive clutter in urban regions that renders the simple image processing techniques impractical. In this method the extracted edges are combined via particle filtering with the noisy GPS way-points in order to produce accurate location estimations for the robot. On the other hand, some techniques have adopted already existing tools in order to obtain the additional information from the aerial images. In [NTD+11] the Google Static Maps API has been utilized in order to create a custom map style, the so called ”road map”, by removing all non-road features and texture from the street maps. Moreover a simple image segmentation is performed by selecting white as the foreground color and black as the road one. The satellite image regions that correspond to the roads are thresholded using their intensity values and the result is a binary image that contains only on road visual features. The ground camera and the satellite image matching, is performed by mixing particle filters with Bayesian tracking. The particle filter holds the samples from the probability distribution of the possible robot states and performs a global refinement in the trajectory of the robot. In a similar attempt the authors in [LCH08] presented a particle filter system performing localization on aerial photographs by matching images acquired from the ground by a monocular vision system. Correspondences between aerial and ground images have been detected by matching line features. These have been generated from aerial images by a Canny edge detector and Progressive Probabilistic Hough Transform (PPHT). Moreover in the methods described in [PMB09] and [Pin08], basic image processing techniques for the features detection combined with machine learning have been adopted for the matching of the aerial and ground camera images, to produce refinements for the initially estimated Visual Odometry (VO). These methods rely on two different assumptions; the first one is that the computational demanding pre-processing steps on the aerial images and the training procedure of a classifier, are routines that can take place off-line, while the second one is that only a very small training set is required to classify large map areas. However, the majority of the existing relevant methods focus on the localisation of robots operating in structured environments, where both the ground and the orbital images are characterised by prominent and well-defined formations such as buildings and roads. Yet, space scenes lack such canonical formations and as a result a different, less texture-dependent method should be adopted instead.

The method proposed in this work processes orbital images that cover the same area as a surface exploratory robot. The result of the discussed method is a refined orientation and position estimation of the robot, as compared to the estimations obtained by pure VO. The robot observes its environment, extracts SIFT features out of it, and utilizes 3D stereo vision techniques to reproject them onto the surface plane. A similar feature extraction procedure is applied to the or-

Figure 2. Satellite image and the respective region of interest.

bital images. The two sets of features are projected on the surface plane and, consequently two different graphs are formed by calculating all the inner-distances (e.g. Euclidean) among the features. We aim at extracting patterns within these graphs in order to assess the quality of the resulting 3D mapping. Therefore, we adopt the *minimum spanning tree* (MST) subgraph detection methodology [GH85], which reveals possible patterns on the topology of the detected features. Due to the nature of the explored surfaces the two different subgraphs are examined for common spatial information by means of a dissimilarity measure such as the Graph Edit Distance (GED). This measure is used to determine whether the orbital image is able to provide any improvement to the robot location estimations, or the algorithm should rely only on the VO output. If GED indicates that orbital images can improve the results, the Iterative Closest Point (ICP) algorithm is used to refine the positioning of the robot backwards since the last executed orbital refinement.

2. ORBITAL IMAGING

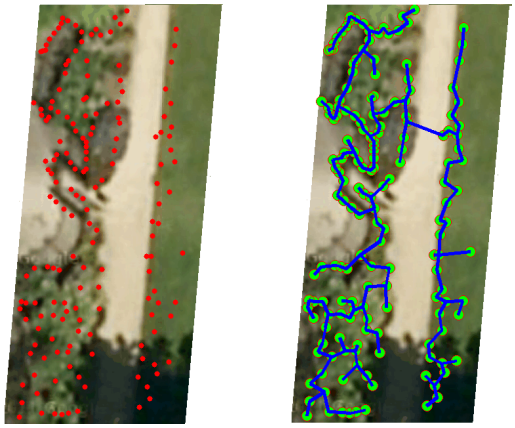
2.1. Satellite Image Acquisition

The orbital images, that cover the same area as the surface exploratory robot, have been obtained by the Google Static Maps API [API]. It's resolution of the source images is approximately 5cm/pixel in cities, however in non-urban areas the resolution can be even worse such as 30cm/pixel. The users can acquire respective images with simple HTTP requests, by specifying the location, the zoom factor, the map type, the image size and the format in the query string. For the needs of the proposed method, the utilized map style was the terrain one. At this point it should be mentioned that an orbital image covers a very large surface, contrary to the surface covered by the field of view of the ground camera; therefore, an orbital image acquired from the Google API can produce refinements for large 3D reconstructed areas. In addition, the feature detection procedure from the ground camera should only occur in real-time, whilst the entire pre-processing of the aerial image can be done off-line. Therefore, the initial orbital image has been divided into regions of interest each one having size of $W \times H$, where W indicates the field of view of the ground camera expressed in pixels over the orbital image and H is the height equal to a specific traveled distance that corresponds to a constant number of frames, assuming a normal average speed for the robot. Fig. 2.1 presents a detected region of interest from a higher resolution satellite image.

The next step of the proposed method comprises the detection of the most salient features in the orbital image. There is a great variety of feature detector algorithms that can be utilized, however, due to the fact that the used orbital images have low resolution and comparisons among views with different scales will take place (as it will be described in Sec. 4), a very robust algorithm should be employed here, namely the Scale Invariant Feature Transform (SIFT)[Low99]. This algorithm comprises a scale and rotation invariant detector and descriptor and the main reason for chosing it lies in its potential to achieve high repeatability, distinctiveness and robustness. The output of the SIFT features on a specific region of interest on an orbital image is described in Fig. 3(a).

2.2. Orbital MST Sub-graph

Assuming that in Fig. 3(a) the data matrix $X^{2\times N}$ corresponds to N detected features and $\mathbf{x}_i \in \mathbb{R}^2$, $i = 1, 2, \ldots, N$, then $G = (X, \mathbb{G})$ is a fully connected, undirected graph defined on X. Also, let



(a) (b)

Figure 3. SIFT features on the satellite image and b) the MST graph between the features superimposed on the image.

$\mathbb{E} = \{e_{ij}\}$ be a set of all edges $e_{ij} = (\mathbf{x}_i, \mathbf{x}_j)$. For each edge e_{ij} we can additionally assign a weight w_{ij} related by its Euclidean distance, i.e. $w_{ij} \equiv \|e_{ij}\| = \|\mathbf{x}_i - \mathbf{x}_j\|$. An acyclic subgraph $\mathfrak{G}$ can be defined, which connects all the vertices such that the total weight, or the total length of the edges, to be minimum. Since the weights are symmetrical, it is sufficient to consider only the edges e_{ij} for which $i > j$ (or alternatively the edges e_{ij} for which $i < j$). This can equivalently be expressed as a problem of finding (n-1) edges forming a tree that minimizes the total weight:

$$\min \sum_{e \in \mathfrak{G}} \|e_{ij}\| \qquad (1)$$

with the condition that $\|\mathfrak{G}\| = n - 1$, $\mathfrak{G} \subset \mathbb{E}$ and $\exists!\ path(\mathbf{x}_i, \mathbf{x}_j), \forall \mathbf{x}_i, \mathbf{x}_j \in X, i > j$. Here $\exists!$ denotes a unique existence and the minimum $path(\mathbf{x}_i, \mathbf{x}_j) \equiv path(\mathbf{x}_i, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_j)$. This optimization is equivalent to computing the MST for a given set of observations. The aforementioned subgraph detection method is performed on the detected features of Fig 3(a) and the resulted MST is presented in Fig. 3(b). Note that the chosen edges outline a specific pattern and describe uniquely the spatial distribution of the detected salient features in the orbital image.

3. SIMULTANEOUSLY LOCALIZATION AND 3D MAPPING

3.1. Visual Odometry

The proposed VO method employs a SIFT feature detection and matching methodology, which detects and matches the salient points within two consecutive frames. The output of this procedure is a set of N matched features between two successive images. In the next step, the previously extracted 2D points, are transformed into 3D ones, i.e. a convertion from image to world coordinates occurs. This procedure comprises two distinct steps: the first one being the disparity estimation of the scene utilizing a stereo correspondence algorithm and, consequently, every salient point obtains a depth value; the second one is the triangulation of the 3D points, taking advantage of the previously estimated disparity values. The proposed system assumes a stereo rig and, therefore, the resulting algorithm is a specially designed stereo correspondence one, which is described analytically in [NSG11]. Once the disparity value has been computed for all the N features, they are transformed into the 3D world coordinate system by triangulation, given the intrinsic parameters of the stereo camera setup. The result of this procedure is a sparse 3D point cloud of N matched features among the consecutive frames. Given the previously matched 3D point clouds, we now seek for the rigid body motion of the stereo camera between all the consecutive pair time instances. The assumption required in the analysis hereafter is that the observed environment is static concerning the two adjacent image pairs and that the sole non-static object is the camera. In this case, the local coordinates of the features position vectors $\mathbf{p}1^{'}, \ldots, \mathbf{p}_N^{'}$ in the reference image of the second pair are related to the position vectors $\mathbf{p}_1, \ldots, \mathbf{p}_N$ in the reference image of the first pair by the equation:

$$\mathbf{p}_i = T + R \cdot \mathbf{p}_i^{'} \qquad \text{for i = 1, 2, \ldots, N,} \qquad (2)$$

where T and R are the translation and the rotation matrices, respectively, describing the camera's movement between the two reference sparse point clouds. In the ideal case, six perfectly matched features are sufficient to compute the

matrices T and R. However, in realistic, error-suffering situations, a larger set of redundant points is needed. The sought T and R should conform with a sum of quadratic differences minimization criterion. The application of a Procrustes transformation [SSV09] to the resulting two point clouds, reveals the relative translation $T(x_0, y_0, z_0)$ and rotation R of the rover. This way, a linear transformation is determined between the point cloud at time t and $t + 1$ that correspond to consecutive samplings.

3.2. 3D Map Building

Given the depth information from the disparity image and the aforementioned triangulation procedure, a dense 3D point is constructed for every consequent frame. Additionally, since the depth information has been extracted from the same camera, a *texture mapping* procedure is straight forward, by re-projecting the calculated disparity image into a z-buffer so that the left reference image $I_{rgb}(x)$ will refer to the color and depth of the same world point. This procedure is repeated in every consequent frame and by utilizing the incremental motion estimation the different 3D point clouds can be merged into a global 3D map of the explored environment.

The output of the previously described map building procedure is depicted in Fig. 4(b), which is the 3D reconstruction of the scene with reference image to the left of the stereo pair Fig. 4(a). The 3D map presented in Fig. 4(b) is a result of merging distinct 3D point clouds following the output of 20 consecutive robot motion estimations. In the ideal case where the motion estimation procedure does not introduce an error in the system, the resulting merged 3D point clouds whould have great consistency. However, due to drifts in VO calculation the map building procedure suffers from accumulative errors causing faulty registrations of the 3D point clouds. The solution to this problem is presented analytically in Sec. 4. The resulted 3D map is then reprojected top-down in order to achieve the same viewpoint as one of the orbital images and, once again, the SIFT feature detection methodology is employed (Fig.4(c)). Then, the set of the detected features is processed further to calculate the MST subgraph revealing the pattern of the topology of the detected features Fig. 4(d).

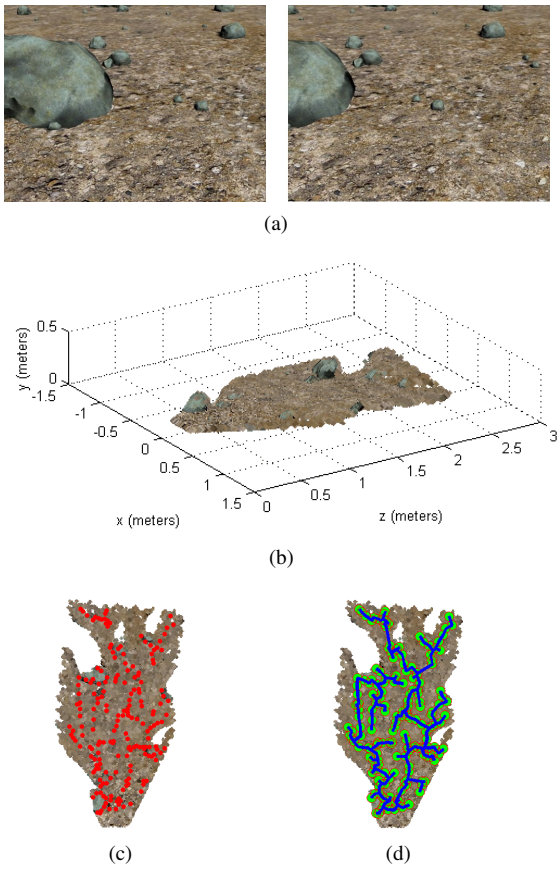


Figure 4. a) An stereo pair sample, b) the respective 3D reconstructed surface, c) top-down re-projection and SIFT features detection and d) the MST graph between the detected features, overlaid on the top-down re-projected scene.

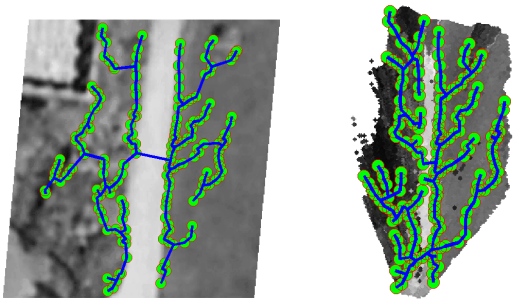
4. MAP MATCHING AND ICP REFINEMENT

In cases where drifts do not occur in the VO calculation, the 3D mapping is free from erroneous registrations. However, this rarely happens and the matching of different 3D point clouds demands further optimization. There are methods such as the *loop-closure* one [NH05], which increases the efficacy of a VO algorithm, however the computational burden of the algorithm is prohibited in space exploratory robots where sparing technologies should be adopted. Therefore, this paper introduces the use of orbital images

as prior knowledge of the unexplored territory which is compared to the 3D reconstructed surface in order to supervise the accuracy of the VO algorithm.

More analytically, once the orbital MST subgraph and its ground plane counterpart have been calculated, a metric that defines the common spatial information of the two different views should be examined. The utilized metric is the GED [GXTL10] deriving from the graph theory. According to the GED, the two examined graphs might describe the same topological pattern, when the number of the common edges to the total number of edges is greater than a predefined threshold. In our case the threshold value was set tolerable enough due to the great unevenness in the resolution between the orbital image and the 3D reconstructed one, i.e. the top-down re-projected image produces more SIFT features than the orbital one. In addition this metric indicates that the resulted MST subgraphs should have the same number of the edges, yet rare to happen. This can be counterbalanced by omitting SIFT features with great spatial proximity to neighbor ones, leading in a subset of features equal to those that produced on the orbital image.

Once the GED between the calculated MST subgraphs is greater than the predefined threshold, we can safely assume that the output of the VO algorithm is accurate enough and further adjustment would not be helpful. On the other hand, if the GED is below the threshold, then it is indicated that the 3D map has been erroneously estimated and, therefore, a further optimization is sought. In these cases the Iterative Closest Point (ICP) algorithm is employed to refine the positioning of the robot backwards since the last check executed with orbital images. In the ICP [BM92], points in a source cloud obtained at time t are matched with their nearest neighboring points in a target cloud acquired at time $t+1$ and a rigid transformation is found by minimizing the sum of the squared spatial error distances between the associated points. ICP has been shown to be effective when the two clouds are already nearly aligned, therefore, an initial motion estimation is firstly performed on the 3D point cloud $(t+1)$ and then the output of the ICP is combined with the rigid transformation. Otherwise, the unknown data association between the point clouds at time t and $t+1$ may lead to convergence at an incorrect local minimum.



(a) (b)

Figure 5. a) Orbital image with the MST subgraph overlaid, b) top-down re-projection of the 3D reconstructed area with the MST subgraph overlaid.

5. ALGORITHM ASSESSMENT

The performance of the proposed localization algorithm has been evaluated with real outdoor data. The dataset used is the New College Dataset [SBC$^+$09]. However, due to the fact that the dataset mainly induces urban areas, the algorithm has been evaluated only in those parts that lack canonical formations. The total traveled distance that has been examined is 1000 frames that corresponds to approximately 57m route. The check with the orbital imaging along this route was performed every 20m with overlapping regions. Indicatively, we present an example were the orbital MST subgraphs and the ground plane MST graph share great coherence and no further ICP optimization was needed (Fig. 5).

More analytically Fig. 5(a) depicts the satellite image with the MST subgraph calculated over the detected features, while Fig. 5(b) depicts the respective 3D reconstructed area with the MST subgraph overlaid on it. Note that in this case the initial detected SIFT features have been downsampled ensuring that the two subgraphs will share the same number of edges. The estimated GED is 0.8820 indicating that the two subgraphs share a great amount of common spatial information something that reveals that the VO and, consequently, the map building procedure was accurate enough and no further refinements should take place. In Fig. 5 we exhibit an occasion where the GED returns value equal to 0.2186, something that indicates revision to the VO cal-

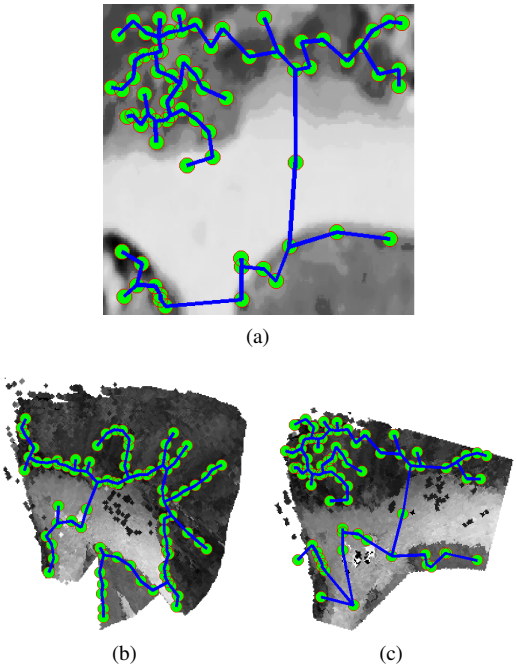


Figure 6. a) Orbital image with the MST subgraph overlaid, b) top-down re-projection of the false 3D reconstructed area with the MST subgraph overlaid, c) top-down re-projection of the 3D reconstructed area refined with ICP, and the MST subgraph overlaid on it.

culation. Moreover, it is obvious that the resulted MST subgraphs do not share any common topological patterns. Therefore, the ICP refinement performed on the on 3D point clouds, since the last executed orbital check. The new GED was calculated to be 0.7805, revealing that the accuracy of the VO and the 3D map building algorithm has been significantly improved. In addition we demonstrate the refined global 3D reconstructed area, that corresponds to the entire traveled route. During this route we performed five different checks with overlapping regions and the GED indicated that in two of them a refinement should be performed with the ICP algorithm. The traveled route is depicted in Fig. 7(a) as it derives from the Google API, while the output of the SLAM algorithm, which is a 3D map is presented in Fig. 7(b).

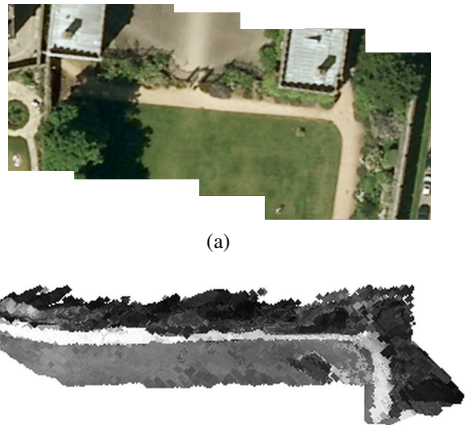


Figure 7. a) Orbital image of the traveled route, b) the 3D map of the proposed methodology including the ICP refinements. Note that none of the buildingsappeared in (a) participates in the 3D map, as they do not appear in any of the stereo pairs.

6. CONCLUSIONS

In this paper a VSLAM localization algorithm that utilizes orbital images for the refinement of the resulted 3D map has been presented. The algorithm takes advantage of a set of efficient tools based on graph theory for the detection and the matching of graph patterns in the topology of detected landmarks. More specifically it utilizes the MST subgraph detection and a simple metric i.e. GED to match the graphs. The whole procedure makes use of the ICP algorithm only when there is a major mismatch between the orbital and the reconstructed images, thus avoiding redundant computational burden. Moreover, the proposed method is specifically targeted to the characteristics of unstructured natural environments, as the ones found on Moon or Mars. Last, we have performed preliminary experiments in unstructured terrains and we found that the proposed method can provide reliable solutions to fine-tune localization and 3D mapping problems.

REFERENCES

[API] Google Static Maps API. http://code.google.com/apis/maps/documentation/staticmaps/.

[BM92] P.J. Besl and N.D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on pattern analysis and machine intelligence*, 14(2):239–256, 1992.

[Doc] SPARTAN System: A Detailed Design Document. http://utopia.duth.gr/ gkostave/spartan/ddd.pdf.

[GH85] R.L. Graham and P. Hell. On the history of the minimum spanning tree problem. *Annals of the History of Computing*, 7(1):43–57, 1985.

[GXTL10] X. Gao, B. Xiao, D. Tao, and X. Li. A survey of graph edit distance. *Pattern Analysis & Applications*, 13(1):113–129, 2010.

[KBN+11] I. Kostavelis, E. Boukas, L. Nalpantidis, A. Gasteratos, and M.A. Rodrigalvarez. Spartan system: Towards a low-cost and high-performance vision architecture for space exploratory rovers. In *IEEE ICCV Workshops*, pages 1994–2001. IEEE, 2011.

[KR04] T. Korah and C. Rasmussen. Probabilistic contour extraction with model-switching for vehicle localization. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 710–715. IEEE, 2004.

[KSD+11] R. Kümmerle, B. Steder, C. Dornhege, A. Kleiner, G. Grisetti, and W. Burgard. Large scale graph-based slam using aerial images as prior information. *Autonomous Robots*, 30(1):25–39, 2011.

[LCH08] K.Y.K. Leung, C.M. Clark, and J.P. Huissoon. Localization in urban environments by matching ground level video images with an aerial image. In *IEEE International Conference on Robotics and Automation*, pages 551–556. IEEE, 2008.

[Low99] D.G. Lowe. Object recognition from local scale-invariant features. In *IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157. Ieee, 1999.

[NH05] P. Newman and K. Ho. Slam-loop closing with visually salient features. In *IEEE International Conference on Robotics and Automation, 2005.*, pages 635–642. IEEE, 2005.

[NSG11] L. Nalpantidis, G.C. Sirakoulis, and A. Gasteratos. Non-probabilistic cellular automata-enhanced stereo vision simultaneous localization and mapping. *Measurement Science and Technology*, 22:114027, 2011.

[NTD+11] M. Noda, T. Takahashi, D. Deguchi, I. Ide, H. Murase, Y. Kojima, and T. Naito. Vehicle ego-localization by matching in-vehicle camera images to an aerial image. In *Computer Vision–ACCV 2010 Workshops*, pages 163–173. Springer, 2011.

[Pin08] O. Pink. Visual map matching and localization using a global feature map. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–7. IEEE, 2008.

[PMB09] O. Pink, F. Moosmann, and A. Bachmann. Visual features for vehicle localization and ego-motion estimation. In *Intelligent Vehicles Symposium, 2009 IEEE*, pages 254–260. Ieee, 2009.

[SBC+09] M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman. The new college vision and laser data set. *The International Journal of Robotics Research*, 28(5):595–599, 2009.

[SKLP06] N. Sünderhauf, K. Konolige, S. Lacroix, and P. Protzel. Visual odometry using sparse bundle adjustment on an autonomous outdoor vehicle. *Autonome Mobile Systeme 2005*, pages 157–163, 2006.

[SSV09] B. Siciliano, L. Sciavicco, and L. Villani. *Robotics: modelling, planning and control*. Springer Verlag, 2009.