A Survey on Data Association Methods in VSLAM

S.SriVidhya[#1], Dr.C.B. Akki[*2], Dr.Prakash S[#3]
[1] Assistant Professor, Dept of ISE, BNM Institute of Technology, Bangalore-560071, India.
[2] Professor, Dept of ISE, SJB Institute of Technology, Bangalore-560060, India.
[3] Professor, Dept of CSE, Dayananda Sagar University, Bangalore, India.

Abstract

In robotics the Simultaneous Localization and Mapping (SLAM) is the problem in which an autonomous robots acquires a map of the surrounding environment while at the same time localizes itself inside this map. One of the most challenging fields of research in SLAM is the so called Visual- SLAM problem, in which various types of cameras are used as sensor for the navigation. Cameras are inexpensive sensors and can provide rich information about the surrounding environment, on the other hand the complexity of the computer vision tasks and the strong dependence on the characteristics of the environment in current approaches makes the Visual-SLAM far to be considered a closed problem.

Visual SLAM (simultaneous localization and mapping) refers to the problem of using images, as the only source of external information, in order to establish the position of a robot, a vehicle, or a moving camera in an environment, and at the same time, construct a representation of the explored zone. Nowadays, the problem of SLAM is considered solved when range sensors such as lasers or sonar are used to build 2D maps of small static environments. However SLAM for dynamic, complex and large scale environments, using vision as the sole external sensor, is an active area of research. The computer vision techniques employed in visual SLAM, such as detection, description and matching of salient features, image recognition and retrieval, among others, are still susceptible of improvement. The objective of this article is to provide new researchers in the field of visual SLAM a brief and comprehensible review of data association categories in VSLAM.

Keywords *Visual SLAM - Detectors-Descriptors-Data association*

1. Introduction

The SLAM is a problem of spatial exploration. The Simultaneous Localisation and Mapping (SLAM) problem asks if it is possible for a mobile robot to be placed at an unknown location in an unknown environment and for the robot to incrementally build a consistent map of this environment while simultaneously determining its location within this map. A solution to the SLAM problem has been seen as a `holy grail' for the mobile robotics community as it would provide the means to make a robot truly autonomous.

The `solution' of the SLAM problem has been one of the notable successes of the robotics community over the past decade. SLAM has been formulated and solved as a theoretical problem in a number of different forms. SLAM has also been implemented in a number of domains from indoor robots, to outdoor, underwater and airborne systems. At a theoretical and conceptual level, SLAM can now be considered a solved problem. However, substantial issues remain in practically realizing more general SLAM solutions and notably in building and using perceptually rich maps as part of a SLAM algorithm.

One of the most fundamental features of an autonomous mobile robot is the capability to localize itself inside the environments where it moves. Without the knowledge of its own position, a robot can't perform complex tasks as rescue, surveillance, or fetch and carry. In order to provide a robot with localization capabilities, the programmer must give it a representation of the environment (map) where it will move. For many reasons, this representation is not always available, for example because of the working area is not known a priori (as in the case of a rescue robot). Generating incrementally consistent maps of the environment, while locating itself within this map is therefore another fundamental task of mobile robots, more general than the localization, and obviously more challenging. In robotics this capability is commonly referred as the Simultaneous Localization and Mapping (SLAM) problem [1], and in the last years it has received much attention within the research community. SLAM has been formulated and solved as a theoretical problem in a number of different ways and many researchers presented several implementations using different robotic platforms and sensors. However, SLAM still remains an open problem, due to the strong dependency of almost all current implementations on the specific environment and the specific sensors used.

Moreover, the capability of autonomously navigate in an unknown environment becomes critically important in indoor application, where no global positioning systems as GPS are available.

The overview of the SLAM process consists of following phases:

Fig. 1: Overview of the SLAM

In Fig. 1 the SLAM involves a moving agent (for example a robot), which embarks at least one sensor able to gather information about its surroundings (a camera, a laser scanner, sonar: these are called exteroceptive sensors). Optionally, the moving agent can in-corporate other sensors to measure its own movement (wheel encoders, accelerometers, and gyrometers: these are known as proprioceptive sensors). The minimal SLAM system consists of one moving exteroceptive sensor (for example, a camera in your hand) connected to a computer.

SLAM consists of three basic operations:

1. **Motion model (Odometry update):**
   The robot moves, reaching a new point of view of the scene. Due to unavoidable noise and errors, this motion increases the uncertainty on the robot's localization.
2. **Inverse observation model (Feature Extraction):**
   It is a mathematical model to determine the position of the landmarks in the scene from the data obtained by the sensors. The robot discovers interesting features in the environment, which need to be incorporated to the map. It is called as features landmarks. Because of errors in the exteroceptive sensors, the location of these landmarks will be uncertain. Moreover, as the robot location is already uncertain, these two uncertainties need to be properly composed.
3. **Direct observation model (Data Association):**
   It is a mathematical model to predict the values of the measurement from the predicted landmark location and the robot localization. The robot observes landmarks that had been previously mapped, and uses them to correct both its self-localization and the localization of all landmarks in space. Therefore, both localization and landmarks uncertainties decrease.

With these three models plus an estimator engine it is able to build an automated solution to SLAM. The estimator is responsible for the proper propagation of uncertainties each time one of the three situations above occurs. An extended Kalman filter (EKF) is used as an estimator. A solution to SLAM needs to chain all these operations together and to keep all data healthy and organized, making the appropriate decisions at every step.

2. Vision in SLAM

Visual SLAM is the process of building maps of the surrounding environment and in the same time estimates the robot motion using mainly visual information. Conventional SLAM approaches commonly use information provided by range finder sensors as lasers or sonar rings. Range finder sensors provide easily interpreted outputs that can be directly used in the SLAM state estimation problem. On the other hand, vision-based sensors provide the robot with a large amount of information that should be properly interpreted before the estimation process. The process of understanding of the sensory information coming from vision is called visual perception. Generally visual perception is a complex task and it involves various scientific subjects as signal processing, geometry and pattern recognition. Often useful information, for example visual landmark positions, are difficult to extract from images due to the sensor noise and the illumination changes, additionally 3D positions are not observable given only a single frame.
A lot of computer vision techniques are involved in Visual SLAM systems, such as visual features detection and extraction (feature selection), features matching (data association), image transformations and structure reconstruction. The current visual-SLAM systems use various types of cameras (perspective, stereo, panoramic). Due to size and balance constraints, small robots are usually equipped with a single, often low-cost, perspective camera.

2.1 Feature Selection

A salient feature is a region of the image described by its 2D position (on the image) and an appearance. The term salient feature is used as a generalization that can include points, regions, or even edge segments which are extracted from images. Feature extracting methods are designed to extract salient areas from an image. There are different features, which can be extracted. Edges, corners and blobs are the most often used features.
The feature should fulfil certain criteria:

1. Invariance - The detection of a specific feature should be invariant with respect to geometric and radiometric distortions, for instance relative rotations or intensity changes.
2. Stability - The detection of a feature should be robust against noise in the observation.
3. Distinctness - The feature should be distinguishable from neighbouring features in terms of local image information.

4. Infrequency - For the task of loop closing detection the surrounding local image information of a feature should be unique.
5. Interpretability - In case of object recognition tasks it is necessary that the feature or a feature group can be assigned to semantic objects.

The salient feature extraction process is composed of two phases: detection and description. The detection consists in processing the image to obtain a number of salient features. The description consists in building a feature vector based on visual appearance in the image. The invariance of the descriptor to changes in position and orientation will permit to improve the image matching and data association processes. The features will be extracted from the IR Laser field which are said to be the landmarks. These features will act as an input to the data association phase. The various descriptors and detectors that have been used in SLAM are discussed below:

2.1.1 Phase I: Detectors

The majority of visual SLAM systems use corners as landmarks due to their invariant features and their wide study in the computer vision context. However, in [7] the edge segments called edge lets in a real-time MonoSLAM system, allowing the construction of maps with high levels of geometrical information are used. The edges are good features for tracking and SLAM, due to their invariance to lighting, orientation and scale changes. The use of edges as features looks promising, since edges are little affected by blurring caused by the sudden movements of the camera [8]. However, the edges have the limitation of not being easy to extract and match. On the other hand, in [9] and [10] the fusion of features (i.e. points, lines and planar structures) in a single map, with the purpose of increasing the precision of SLAM systems and creating a better representation of the environment was investigated.

2.1.2 Phase II: Visual Descriptors

The set of different descriptors that have been evaluated in this study are:

SIFT: The Scale-Invariant Feature Transform (SIFT) detects distinctive key points in images and computes a descriptor for them. The algorithm, developed by Lowe, was initially used for object recognition tasks [2]. SIFT features are located at maxima and minima of a difference of Gaussian functions applied in scale space. Next, the descriptors are computed based on orientation histograms at a 4x4 sub region around the interest point, resulting in a 128 dimensional vector.

SURF: Speeded Up Robust Features (SURF) is a scale and rotation invariant descriptor presented in [3]. The detection process is based on the Hessian matrix. SURF descriptors are based on sums of 2D Haar wavelet responses, calculated in a 4x4 sub region around each interest point. The standard SURF descriptor has a dimension of 64 and the extended version (e-SURF) of 128. The u-SURF version is not invariant to rotation and has a dimension of 64.

Gray level patch: This method describes each landmark using the gray level values at a sub region around the interest point. This method has been used in [4] as descriptor of Harris points in a visual SLAM framework.

Orientation Histograms: The orientation histograms are computed from the gradient image, which represents the gray value variations in the x and y direction. In [5] orientation histograms are applied for navigation tasks.

Zernike Moments: The moment formulation of the Zernike polynomials [6] appears to be one of the most popular in terms of noise resilience, information redundancy and reconstruction capability. They are constructed using a set of complex polynomials which form a complete orthogonal basis set.

2.2 Data Association in VSLAM:

In data association phase the newly extracted features will be mapped with the existing features and uses them to correct both the localization of robot and the landmarks. When the odometry changes as the robot moves to new position it is updated in the kalman filter through odometry update phase, which is a repetitive process. The Kalman filter is the heart of the SLAM process. It is responsible for updating where the robot thinks is based on these features. In visual SLAM data association is performed by means of visual place recognition techniques. It is categorized into following cases:

- Cooperative SLAM
- Loop closure detection
- Kidnapped SLAM

2.2.1 Cooperative SLAM

CoSLAM is a vision-based simultaneous localization and mapping (SLAM) in dynamic environments with multiple cameras. These cameras move independently and can be mounted on different platforms. All cameras work together to build a global map, including 3D positions of static background points and trajectories of moving foreground points. The inter-camera pose estimation and inter-camera mapping to deal with dynamic objects in the localization and mapping process has been used. To enhance the system robustness, the position uncertainty of each map point has to be maintained. To facilitate intercamera operations, cameras are clustered into groups according to their view overlap, and manage the split and merge of camera groups in real-time.

In dynamic environments, it is often important to reconstruct the 3D trajectories of the moving objects for tasks such as collision detection and path planning [11], [12]. This 3D reconstruction of dynamic points can hardly be achieved by a single camera. To address these problems, a collaborative visual SLAM system using multiple cameras was used. The relative positions and orientations between cameras are allowed to change over time. This setting is different from existing SLAM systems with a stereo camera [13], [14] or a multi-camera rig [15] where all cameras are fixed on a single platform. The camera configuration makes the system applicable to the following interesting cases:

1) Wearable augmented reality [16], where multiple cameras are mounted on different parts of the body

2) Robot teams [17], [18], [19], where multiple robots work in the same environment and each carries a single camera because of limited weight and energy capacity, e.g. micro air vehicles (MAVs)

The collaborative SLAM system treats each camera as a sensor input, and incorporates all inputs to build a global map, and simultaneously computes the poses of all cameras over time. The system detects and tracks feature points at every frame, and feed them to the four SLAM components. The Kanade-Lucas-Tomasi (KLT) [20] tracker for both feature detection and tracking was used because of its good balance between efficiency and robustness. However, there is no restriction to use other feature detectors and trackers such as the 'active matching' [21]. The four SLAM components are 'camera pose estimation', 'map building', 'point classification', and 'camera grouping'. The main pipeline of the system follows conventional sequential structure-from motion (SFM) methods. It is assumed that all cameras look at the same initial scene to initialize the system. After that, the 'camera pose estimation' component computes camera poses at every frame by registering the 3D map points to 2D image features. From time to time, new map points are generated by the 'map building' component. At every frame, points are classified into different types by the 'point classification' component. The system maintains the view overlap information among cameras throughout time. The 'camera grouping' component separates cameras into different groups, where cameras with view overlap are in the same group. These groups could merge and split when cameras meet or separate. Several issues in pose estimation, mapping and camera group management were addressed, so that the system can work robustly in challenging dynamic scenes and the whole system runs in real-time.

The cooperative mapping consists in align two or more partial maps of the environment collected by a robot in different periods of operation or by several robots at the same time (visual cooperative SLAM) [22][23][24]. In the past, the problem of associating measurements with landmarks on the map was solved through algorithms such as Nearest Neighbour, Sequential Compatibility Nearest Neighbour and Joint Compatibility Branch and Bound [25]. However, these techniques are similar because they work only if a good initial guess of the robot in the map is available [26].

2.2.2 Loop-Closure Detection

A graph is constructed where nodes represent locations in which a complete 360-degrees panoramic reference image is acquired and links represent consecutive reached reference positions. Loops in the graph represent previously visited places. As described in [27], while the robot moves it checks for a loop closing for every incoming (perspective) image. If the loop-closure is not detected, a new reference panoramic image is acquired and hence it is associated to a new node added to the graph. The process for the loop closing detection is the following:

1. A new perspective image is acquired.
2. If the similarity between the current perspective image and the last panoramic image added to the graph is over a threshold then, return to point (1), otherwise proceed to point (3).
3. A loop-closure between the current image and all the reference panoramic images (except the last visited) is attempted.
4. If the loop-closure is detected, a link between the last visited node in the graph and the node associated with the matched reference image is added.
5. If the loop-closure fails, a new reference panoramic image is acquired and hence it is associated to a new node added to the graph.
6. The process restarts from the point (1).

Loop closure detection consists in recognizing a place that has already been visited in a cyclical excursion of arbitrary length [22][28][29]. This problem has been one of the greatest obstacles to perform large scale SLAM and recover from critical errors. From this problem arises another one called perceptual aliasing [26][30] where two different places from the surrounding are recognized as the same. This represents a problem even when using cameras as sensors due to the repetitive features of the environment, e.g. hallways, similar architectural elements or zones with a large quantity of bushes. A good loop closure detection method must not return any false positive and must obtain a minimum of false negatives.

According to Williams [31] detection methods for loop closures in visual SLAM can be divided into three categories: (1) map to map; (2) image to image; and (3) image to map. Categories differ mainly about where the association data are taken from (metric map space or image space). However the ideal would be to build a system that combines the advantages of all three categories. Loop closure detection is an important problem for any SLAM system, and taking into account that cameras have become a very common sensor for robotic applications, many researchers focus on vision methods to solve it. A similarity matrix to code the relationships of resemblance between all the possible pairs in captured images has been proposed in [32]. They demonstrate by means of single value decomposition that it is possible to detect loop closures, despite of the presence of repetitive and visually ambiguous images. A unified method to recover from tracking failures and detect loop closures in the problem of monocular visual SLAM in real time has been proposed in [33]. They also propose a system called GraphSLAM where each node stores landmarks and maintains estimations of the transformations relating nodes. In order to detect failures or loop closures, they model appearance as a Bag of Visual Words (BoVW) to find the nodes that have a similar appearance in the current video image. A method to detect loop closures under a scheme of Bayesian filtering and a method of incremental BoVW, where the probability to belong to a visited scene is computed for each acquired image and it has been proposed in [30]. A probabilistic framework to recognize places, which uses only image appearance data, has been proposed in [28].

Through the learning of a generative model of appearance, they demonstrate that not only it is possible to compute the resemblance of two observations, but also the probability that they belong to the same place and thus they calculate a probability distribution function (pdf) of the observed position. Finally, in [29] a new topometric representation of the world, based on co-visibility, which allows simplifying data association and improving the performance of recognition based on appearance has been proposed.

All the loop closure works described above, aim to achieve a precision of 100%.This is due to the fact that a single false positive can cause permanent failures during the creation of the map. In the context of SLAM, false positives are serious than false negatives [34]. False negatives reduce recall percentage but have no impact on precision percentage. Thus, in order to determine the efficiency of a loop closure detector, the recall rate should be as high as possible, with a precision of 100%.

2.2.3 Kidnapped robot

The “Kidnapped robot problem” is closely related to multi-session mapping. In the Kidnapped robot problem, the goal is to estimate the robot’s position with respect to a prior map given no *a* priori information about the robot’s position. In multi-session SLAM, in conjunction with this global localisation problem the robot should begin mapping immediately and upon localisation the map from the current session should be incorporated into the global map from previous sessions.

If the robot is put back into an already mapped zone, without the knowledge of its displacement while it is being transported to that place, or when robot performs blind movements due to occlusions, temporary sensor malfunction, or fast camera movements [34][35][36]. A system capable of tolerating the uncertainty about camera pose and recover from minor tracking failures generated by continuous erratic movement or by occlusions has been proposed in [35]. The work consists in generating a descriptor (based on SIFT) at multiple resolutions to provide robustness in the data association task. In addition, it uses an index based on low-order coefficients of the Haar wavelet. A re-localization module that monitors the SLAM system, detects tracking failures, determines the camera pose in the map landmarks framework and resumes tracking as soon as conditions have improved has been proposed in [36]. Re-localization is performed by a landmark recognition algorithm using the randomized trees classifier technique proposed in [37] and trained online through a feature harvesting technique. In this way a high recovery rate and a rapid recognition time are obtained. To find the camera pose, candidate poses are generated from correspondences between the current frame and landmarks on the map. There is a selection of sets of three potential matches, then, all the consistent poses with these sets are calculated by a three-point algorithm. These poses are evaluated seeking consensus among the other correspondences in the image found by RANSAC. If a pose with a large consensus is found, that pose is assumed to be correct.

3. Conclusion

VSLAM is an extension of SLAM employed in robotic application. The data association is still an open research area in the fields of robotic vision. In this paper, authors have surveyed for the existing challenges, solutions to these challenges related to VSLAM. The authors have also tried to cover the basics of VSLAM available in the literature. The focus in this survey work is more towards various data association techniques in VSLAM. Our future aim is to come out with a new sophisticated VSLAM that can outperform the existing techniques.

Acknowledgement

This work is part of Aeronautics Research and Development Board, Government of India. Authors wholeheartedly express their gratitude for having shared the work.

References

[1] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part I. Robotics and Automation Magazine, IEEE, 13:99{110, June 2006.
[2] D.G. Lowe. Object recognition from local scale-invariant features. In Int. Conf. On Computer Vision, 1999.
[3] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. In European Conference on Computer Vision, 2006.
[4] Andrew J. Davison and David W. Murray. Simultaneous localisation and map-building using active vision. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002.
[5] J. Kosecka, L. Zhou, P. Barber, and Z. Duric. Qualitative image based localization in indoor environments. In Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, 2003.
[6] F. Zernike. Diffraction theory of the cut procedure and its improved form, the phase contrast method. Physica, 1:689–704, 1934.
[7] Eade E, Drummond T (2006a) Edge landmarks in monocular SLAM. In: Proceedings of the British machine vision conference
[8] Klein G, Murray D (2008) Improving the agility of key frame-based SLAM. In: Proceedings of the European conference on computer vision, pp 802–815
[9] Gee A, Chekhlov D, Calway A, Mayol W (2008) Discovering higher level structure in visual SLAM. IEEE Trans Robot 24(5):980–990
[10] Martinez J, Calway (2010) A unifying planar and point mapping in monocular SLAM. In: Proceedings of the British machine vision conference, pp 1–11
[11] D. Wolf and G. Sukhatme. Mobile robot simultaneous localization and mapping in dynamic environments. Autonomous Robots, 19(1):53–65, 2005.
[12] C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant Whyte. Simultaneous localization, mapping and moving object tracking. Int'l of Robotics Research, 26(9):889, 2007.
[13] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In IEEE Proc. of CVPR, volume 1, 2004.
[14] L. Paz, P. Pini´es, J. Tard´ os, and J. Neira. Large-scale 6-dof slam with stereo-in-hand. IEEE Trans. on Robotics, 24(5):946–957, 2008.
[15] M. Kaess and F. Dellaert. Visual slam with a multi-camera rig. Georgia Institute of Technology, Tech. Rep. GIT-GVU-06-06, 2006.
[16] R. O. Castle, G. Klein, and D. W. Murray. Video-rate localization in multiple maps for wearable augmented reality. In Proc 12th IEEE Int Symp on Wearable Computers, pages 15–22, 2008.
[17] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In IEEE Proc. of Robotic and Automation, volume 1, pages 476–481, 2002.
[18] S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In IEEE Proc. of Robotics and Automation, volume 1, pages 321–328. IEEE, 2002.
[19] J. Allred, A. Hasan, S. Panichsakul, W. Pisano, P. Gray, J. Huang, R. Han, D. Lawrence, and K. Mohseni. Sensorflock an airborne wireless sensor network of micro-air vehicles. In Proc. of Int'l Conf. on Embedded networked sensor systems, pages 117–129. ACM, 2007.
[20] J. Shi and C. Tomasi. Good features to track. In IEEE Proc. Of CVPR, pages 593–600, 1994.
[21] M. Chli and A. J. Davison. Active matching for visual tracking. Robot. Auton. Syst., 57(12):1173–1187, 2009.
[22] Ho K, Newman P (2007) Detecting loop closure with scene sequences. Int J Comput Vis 74(3):261–286.
[23] Gil A, Reinoso O, Ballesta M, Juliá M (2010) Multi-robot visual SLAM using a rao-blackwellized particle filter. Robot Autonom Syst 58(1):68–80
[24] Vidal T, Berger C, Sola J, Lacroix S (2011) Large scales multiple robot visual mapping with heterogeneous landmarks in semi-structured terrain. Robot Autonom Syst, pp 654–674
[25] Neira J, Tardós JD (2001) Data association in stochastic mapping using the joint compatibility test. In: Proceedings of the IEEE international conference on robotics and automation 17(6): 890–897
[26] Cummins M, Newman P (2008) FAB-MAP: probabilistic localization and mapping in the space of appearance. Int J Robot Res 27(6):647–665
[27] A. Angeli, S. Doncieux, J.A. Meyer, and D. Filliat. Incremental vision based topological slam. In In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1031{1036, 22-26 Sept.2008.
[28] Clemente L, Davison A, Reid I, et al (2007) Mapping large loops with a single hand-held camera. In: Proceedings of robotics: science and systems conference
[29] Mei C, Sibley G, Cummins M et al (2010) RSLAM: a system for large-scale mapping in constant-time using stereo. Int J Comput Vision 94(2):1–17
[30] Angeli A, Doncieux S, Filliat D (2008) Real time visual loop closure detection. In: Proceedings of the IEEE international conference on robotics and automation
[31] Williams B, Cummins M, Neira J, Newman P, Reid I, Tardós JD (2009) A comparison of loop closing techniques in monocular SLAM. Robot Autonom Syst 57(12):1188–1197.
[32] Ho K, Newman P (2007) Detecting loop closure with scene sequences. Int J Comput Vis 74(3):261–286
[33] Magnusson M, Andreasson H, et al (2009) Automatic appearance-based loop detection from 3D laser data using the normal distribution transform. J Field Robot. Three Dimensional Mapping Parts 2, 26(12):892–914
[34] Chekhlov D, Mayol W, Calway A (2008) Appearance based indexing for relocalisation in real-time visual SLAM. In: Proceedings of the British machine vision conference, pp 363–372
[35] Eade E, Drummond T (2008) Unified loop closing and recovery for real time monocular SLAM. In Proceedings of the British Machine vision conference
[36] Williams B, Klein G, Reid I (2007) Real-time SLAM relocalisation. In: Proceedings of the IEEE international conference on computer vision
[37] Lepetit V, Fua P (2006) Key point recognition using randomized trees. IEEE Trans Pattern Anal Mach Intell 28(9): 1465–1479 IEEE Std. 802.11, 1997.