

# Design and Analysis of Visualization Techniques for Mobile Robotics Development

Alex Kozlov  
The University of Auckland, New Zealand  
ako002@aucklanduni.ac.nz

Bruce A. MacDonald  
The University of Auckland,  
New Zealand  
b.macdonald@auckland.ac.nz

Burkhard C. Wünsche  
The University of Auckland,  
New Zealand  
burkhard@cs.auckland.ac.nz

## ABSTRACT

Simultaneous localization and mapping (SLAM) algorithms are of vital importance in mobile robotics. This paper presents novel Augmented Reality (AR) visualization techniques for SLAM algorithms, with the purpose of assisting algorithm development. We identify important algorithm invariants and parameters and combine research in uncertainty visualization and AR, to develop novel AR visualizations, which offer an effective perceptual and cognitive overlap for the observation of SLAM systems. A usability evaluation compares the new techniques with the state-of-the-art inferred from the SLAM literature. Results indicate that the novel correlation and color-mapping visualization techniques are preferred by users and more effective for algorithm observation. Furthermore the AR view is preferred over the non-AR view, while being at least similarly effective. Since the visualizations are based on general algorithm properties, the results can be transferred to other applications using the same class of algorithms, such as Particle Filters.

## Keywords

Algorithm visualisation, augmented reality, robot programming, human-robot interfaces

## 1 INTRODUCTION

*Simultaneous Localization and Mapping (SLAM)* [SSC90, LDW91] is a popular and important class of estimation algorithms, addressing the challenge of autonomous map-building for mobile robots. A robot must have a model, or a map, of the physical environment in order to carry out useful navigation tasks. The robot must additionally localize itself within the environment. In SLAM the robot autonomously explores and maps its environment with its sensors while localizing itself at the same time. Despite considerable research, open challenges in SLAM include implementations in unstructured, difficult, and large scale environments [BFMG07], multi-robot SLAM [NTC03] as well as SLAM consistency and convergence [MCDFC07].

SLAM development is made more difficult by its probabilistic nature. In SLAM, neither the robot location nor the environment map are known in advance. However, in order to map the environment the robot location needs to be known with accuracy, and in order to localize the robot the environment map needs to be known

with accuracy. Visualizations aid the development and testing of SLAM algorithms by revealing relationships between robot and algorithm states and environmental parameters. Existing SLAM visualization techniques are purely virtual and limited to basic state information, thus lacking *perceptual* and *cognitive* overlap between the robot and the human developer [BEFS01].

*Augmented Reality (AR)* involves spatially registering virtual objects in real time within a view of a real scene [BR05, Azu97]. AR has been used in robotics to enhance the human-robot interface, but has never been applied to SLAM visualization. This paper presents and evaluates novel AR visualization techniques for SLAM with the purpose of assisting algorithm development and debugging. The introduced concepts and lessons learned are applicable to other estimation algorithms in robotics and related fields.

Section 2 outlines the SLAM algorithms for which the visualizations have been developed. Section 3 presents a review of fields we draw on. Section 4 summarizes the visualization requirements. Section 5 explains the visualization techniques. Section 6 presents the evaluation of the visualizations, and section 7 concludes our paper.

## 2 SLAM BACKGROUND

The two most popular SLAM algorithm archetypes, the Extended Kalman Filter (EKF) SLAM [GL06, DWB06] and FastSLAM [MTKW03, Mon03], are both based on recursive Bayesian estimation. These

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

algorithms were targeted for visualization because they are the two most important and prevalent SLAM solution methods [DWB06]. The visualizations presented in this paper would also be relevant for any modified Kalman Filter or Particle Filter based algorithm.

## 2.1 EKF SLAM

In feature-based EKF SLAM the state is represented by a multivariate Gaussian distribution with mean  $x$  and covariance  $P$ :

$$x = \begin{bmatrix} x_r \\ x_{f_1} \\ \vdots \\ x_{f_n} \end{bmatrix} \quad (1)$$

$$P = \begin{bmatrix} P_r & P_{r,f_1} & \cdots & P_{r,f_n} \\ P_{f_1,r} & P_{f_1} & \cdots & P_{f_1,f_n} \\ \vdots & \vdots & \ddots & \vdots \\ P_{f_n,r} & P_{f_n,f_1} & \cdots & P_{f_n} \end{bmatrix} \quad (2)$$

$x_r$  is the estimated robot pose and  $x_{f_i}, i = 1, \dots, n$  is the estimated position of an environment feature  $f_i$ . The main diagonal elements  $P_r, P_{f_1}, \dots, P_{f_n}$  are error covariance matrices of the robot pose and the landmark locations. The off-diagonal elements are cross-covariance matrices between the robot and feature positions. The recursive estimation steps of the algorithm are *motion prediction*, *data association*, *observation update* and *feature initialization*.

## 2.2 FastSLAM

In FastSLAM, which is based on Rao-Blackwellized Particle Filters, the state is represented by a set of  $N$  particles:

$$\{w^{(i)}, X_r^{(i)}, \mu_{f_1}^{(i)}, \dots, \mu_{f_n}^{(i)}, \Sigma_{f_1}^{(i)}, \dots, \Sigma_{f_n}^{(i)}\}_i^N \quad (3)$$

where for particle  $i$ ,  $w^{(i)}$  is the particle weight,  $X_r^{(i)}$  is the sampled robot path, and each map feature  $f_j$  is represented independently by a Gaussian distribution with mean  $\mu_{f_j}^{(i)}$  and covariance  $\Sigma_{f_j}^{(i)}$ . The recursive estimation steps of the algorithm are *motion prediction*, *weight update*, *resampling* and *map update*.

## 3 LITERATURE REVIEW

A number of *Robotics Development Environments (RDEs)* are available for use in robotics programming, but none offers visualizations for SLAM. Examples include Player [GVS<sup>+</sup>01], CARMEN (Carnegie Mellon robot navigation toolkit) [MRT] and Pyro (Python Robotics) [BKMY03]. These offer purely virtual sensor data visualizations. Possibly the most

flexible support for visualizations is offered by ROS (Robot Operating System) [QGC<sup>+</sup>09], which includes a variety of data-types such as point clouds, geometric primitives, robot poses and trajectories.

No formal studies have been done for visualization techniques in SLAM. The SLAM visualization state of the art is inferred from the visual representations of SLAM systems and data in the literature. The current “conventional” method of visualizing EKF-style SLAM is by showing the mean estimates for the robot and features, along with the covariance ellipsoids showing the individual uncertainties (e.g. [BNG<sup>+</sup>07, NCMCT07]). For Particle Filter type SLAM, all current robot poses and mean feature locations are shown for all particles (e.g. [MTKW03, Mon03]). Perhaps the most interesting example of an existing SLAM visualization is the 3D graphical representation of the robot in relation to the mapped obstacles, with the uncertainties shown by dotted lines around the objects [NLTN02]. Martinez-Cantin et al. visualized a constructed 3D map from the robot’s point of view [MW03].

None of the basic SLAM visualizations suggested so far employs an AR environment. However, AR systems have been developed and used in robotics. An example is *ARDev* [CM06, CM09]. It provides perceptual overlap between the human and the robot by visualizing sensor data within the robot’s real world environment. Nunez et al. use AR for supervision of semi-autonomous robot navigation tasks [NBPLS06]. A topological map is generated online and visualized with AR. Daily et al. use AR to supervise a swarm of 20 robots for search and rescue scenarios [DCMP03]. Virtual arrows above every swarm member in view convey the intention and direction of travel. AR has also seen considerable application in mobile robot tele-operation [BOGH<sup>+</sup>03] and manipulator robotics [NCE<sup>+</sup>07].

## 4 VISUALIZATION REQUIREMENTS

The underlying requirement for all of the visualizations is to embed information within the context of the real world environment the mobile robot operates in. This provides a qualitative comparison between the estimates and the ground truth, and shows sources of potential errors within the real environment.

### 4.1 EKF SLAM Requirements

The fundamental EKF SLAM requirement is to visualize the state and the individual uncertainty covariances. The state consists of 2D robot and feature locations, and the robot orientation in the ground plane. The 2 by 2 covariance matrices for the robot and each feature indicate the positional uncertainty, together with the robot orientation variance.

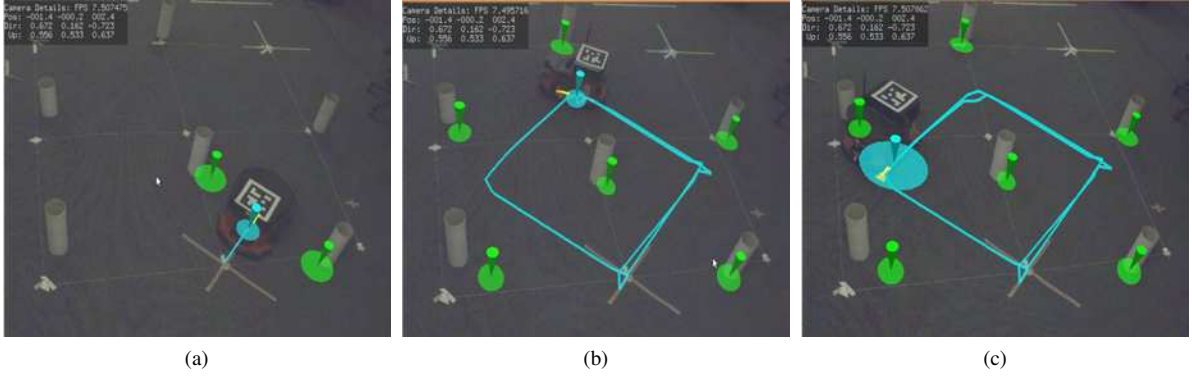


Figure 1: **Conventional AR EKF Visualization:** EKF-SLAM state and covariance visualization in AR, showing progression over time. The AR view provides a direct comparison of the estimates against the ground truth in the real world environment; this shows the discrepancies between the truth and the estimates.

*Correlations* between features are well known to be important for correct SLAM operation [DWRN96]. In [DNC<sup>+</sup>01] feature correlations are defined as follows. Let  $d_{ij}$  be the relative position between any two feature estimates  $f_i$  and  $f_j$ . Let  $P_{d_{ij}}$  be the covariance of  $d_{ij}$  as follows:

$$d_{ij} = x_{f_i} - x_{f_j} \quad (4)$$

$$P_{d_{ij}} = P_{f_i} + P_{f_j} - P_{f_i, f_j} - P_{f_i, f_j}^T \quad (5)$$

$P_{d_{ij}}$  is a measure of relative error between the two features, and is therefore also a measure of their correlation. The expected convergence property [HD07, DNC<sup>+</sup>01] is that correlations strengthen as observations are made over time, i.e. the volume of uncertainty in  $P_{d_{ij}}$  is non-increasing. Visualization of the correlation behaviour is essential. Violations of this behaviour (i.e. weakening correlations) indicate problems in the SLAM system, and therefore must be detected by the user. Specifically, the 2 by 2 covariance matrix  $P_{d_{ij}}$  must be visualized for all feature pairs, together with its non-increasing volume of uncertainty trend. Violations must be exemplified.

## 4.2 FastSLAM Requirements

The fundamental FastSLAM requirement is to visualize the state represented by the set of particles. This means visualizing 2D points for the robot and Gaussian mean feature locations, for all particles. Additionally robot orientations for all particles must be visualized.

Due to the Rao-Blackwellized structure of FastSLAM, sampling is only done on the robot path. The error in the map estimation for a given particle is dependent on the accuracy of the robot path. For this reason, it is important to visualize the relationship between the robot path and map estimates within particles, or intra-particle associations. Specifically, this refers to visually linking estimates from the same particle, and distinguishing these from other particles. Visualization of the individual weight for each particle is also important

in order to gain insight into the resampling phase of the algorithm.

Lastly, a more qualitative and more intuitive representation of the SLAM solution produced by the filter would be useful. This needs to show a better overall picture of the solution, possibly at the cost of lower information content or being less exact.

## 5 AR VISUALIZATION TECHNIQUES

### 5.1 EKF SLAM Visualizations

#### 5.1.1 Conventional EKF SLAM Visualization

Fig. 1 presents the state-of-the-art conventional EKF visualization implemented in AR. The underlying real world images present an overhead view of the robot and its environment. The robot is a PIONEER 3-DX [Rob08]. The map the robot is building consists of two dimensional points in the ground plane represented by white cardboard cylinders. The cylinders are the only physical objects being mapped and are extracted from raw laser rangefinder data. The robot drives around and performs SLAM in a small 1 by 1 meter loop. The graphical icons augmenting the video frames represent SLAM information:

- **Cyan Marker** - The cyan downward pointed cone represents the estimated robot position. The cyan ellipsoid underneath is the robot position covariance. The cyan line is the robot path.
- **Green Marker** - The green downward pointed cone represents the estimated feature position. The green ellipsoid underneath is the feature position covariance.
- **Yellow Marker** - The yellow triangular pointer represents the robot orientation estimate. A semitransparent yellow circular wedge represents the orientation variance.

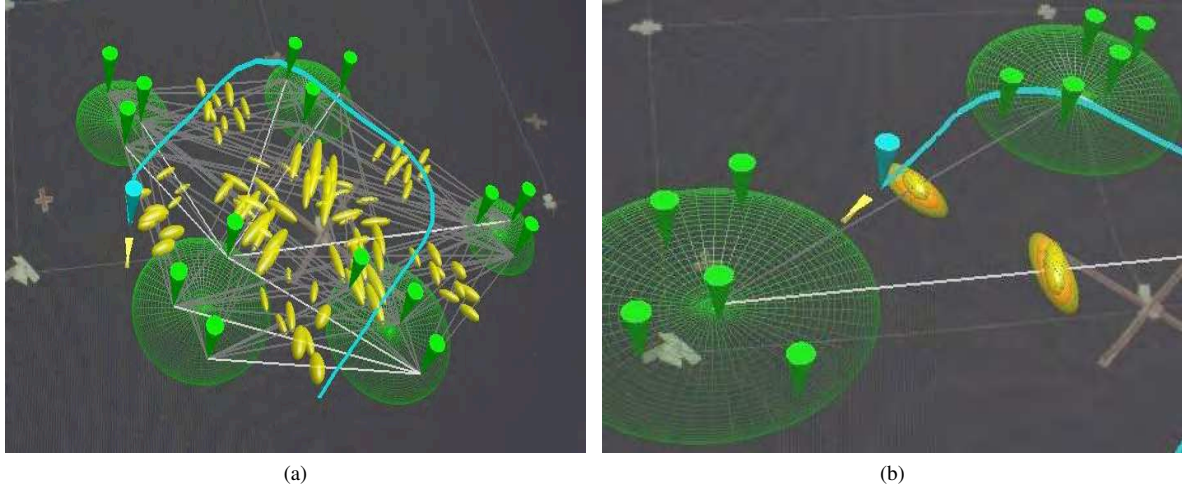


Figure 2: **EKF SLAM Correlations Clustering.** (a) shows all inter-cluster correlations, (b) shows only the maximum, mean, and minimum inter-cluster correlations. Green wireframe circles represent spatial clusters.

The markers represent the SLAM estimates for qualitative real-time visual comparison against the ground truth presented in the real image, i.e. the green markers correspond to the white cardboard cylinders and the blue marker to the physical robot. For the orientation an “arrow” type marker was chosen, as commonly used in SLAM and robotics visualizations. For the covariance, the common tensor ellipsoid glyph was used, which is superior to line or arrow type ellipsoid glyphs. Colour was used to define specifically what the estimate refers to, i.e. robot or features. The design follows the “Natural Scene Paradigm”, which is based on humans’ ability to immediately perceive complex information in a natural scene [WL01].

### 5.1.2 Correlations Visualization

In previous work [KMW09] we presented the novel feature correlation visualization shown in Fig. 2a. It satisfies the requirements discussed earlier. For every pair of features  $\{f_i, f_j\}$  the visualization contains:

- A line linking the feature estimates  $f_i$  and  $f_j$
- A yellow tensor “correlation ellipsoid” for  $P_{d_{ij}}$  rendered at the half-way point on the line

As  $P_{d_{ij}}$  is a two-dimensional covariance, it produces a 2D tensor ellipsoid. However, the problem of visual cluttering becomes evident when many such ellipsoids grow in size and overlap. It becomes difficult to discern any individual ellipsoids. To mitigate this issue the ellipsoids were inflated to a shaded 3D shape. The second eigenvalue is used for the length of the axis into the third dimension. Giving a 3D shaded volume to the correlation ellipsoids provides better visual distinction to overlapping ellipsoids, however a limitation of

this method is that it occludes more of the background world image.

Strengthening correlations show up as decreasing volumes of the correlation ellipsoids. If the volume of the correlation ellipsoid *increases* (i.e. the correlation weakens), this is considered a violation of the expected behaviour. This occurrence is exemplified in the visualization by changing the colour of the ellipsoid to red. Thus, the visualization allows the observation of the expected correlations trend, and the detection of its violations.

The problem of visual cluttering is resolved by *spatial clustering* using the standard single-linkage hierarchical clustering method [LL98]. Features are divided into spatial clusters, and only the correlations between features in different clusters are shown. The green wireframe circles exemplify the clusters computed by the algorithm. This image demonstrates a pure virtual simulation of the SLAM algorithm, and hence no physical robot and environment is shown.

In order to further reduce the number of correlations in view the user can select to only see the minimum (yellow), mean (orange), and maximum (yellow) inter-cluster correlations (Fig 2b). The expected correlation convergence can be observed through the non-increasing size of the minimum correlation ellipsoid [Koz11].

## 5.2 FastSLAM Visualizations

### 5.2.1 Conventional FastSLAM Visualization

Fig. 3 presents the conventional state-of-the-art FastSLAM visualization implemented for the first time in AR. The underlying real world images present an overhead view of the robot and the environment the robot is working in. The graphical icons augmenting the video frames represent SLAM information:

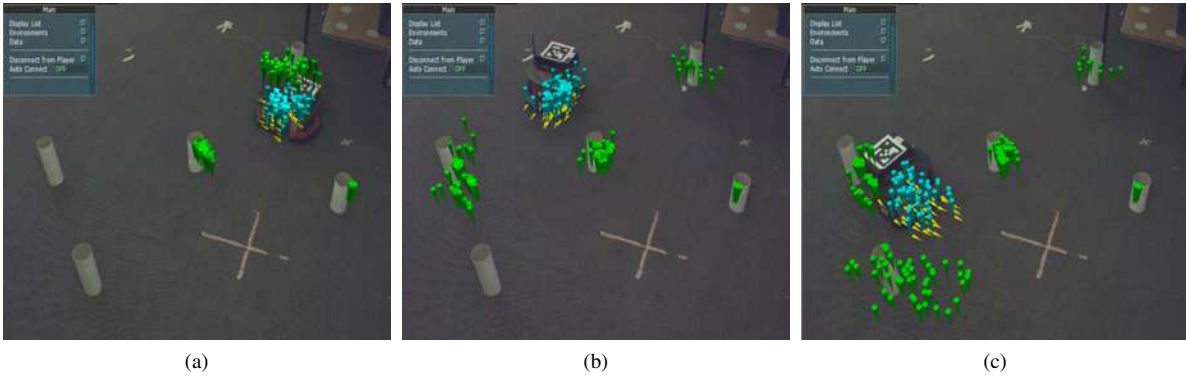


Figure 3: **Conventional FastSLAM AR Visualization:** particle representation for the robot pose and features, showing the joint SLAM posterior distribution computed by the particle filter and its progression over time.

- **Cyan Marker** - The cyan downward pointed cone represents the sampled robot location for a given particle.
- **Yellow Marker** - The yellow arrow-type marker represents the sampled robot orientation for a given particle.
- **Green Marker** - The green downward pointed cone represents a Gaussian mean feature location for a given particle.

The visualization shows the joint SLAM posterior probability distribution of the robot pose and the feature map. As for the EKF, the markers represent the SLAM state for qualitative visual comparison against the ground truth presented in the real image, i.e. the green markers correspond to the white cardboard cylinders and the blue marker to the physical robot.

### 5.2.2 Colour Mapping Visualizations

Fig. 4 presents a colour-mapping visualization technique addressing the requirements of intra-particle associations and particle weights, as discussed earlier. First the centroid and maximum distance from the centroid are computed for the current robot positions in the particles. This creates a local polar coordinate frame for the robot positions (and thus the particles they belong to), originating at the centroid. Then each particle's polar coordinates are mapped directly onto the Hue and Saturation parameters in the HSV colour model. Thus, each particle which has a unique robot position is assigned a unique colour. This colour is used to encode members of the same particle (intra-particle associations), e.g. a red feature and a red robot pose belong to the same particle. This shows the important relationship between the map and robot path estimations. In the final step, the particle weight is encoded into the Value (brightness) parameter of the HSV model. Lighter coloured markers indicate higher weights, and darker colours indicate lower weights. Fig. 5 shows the

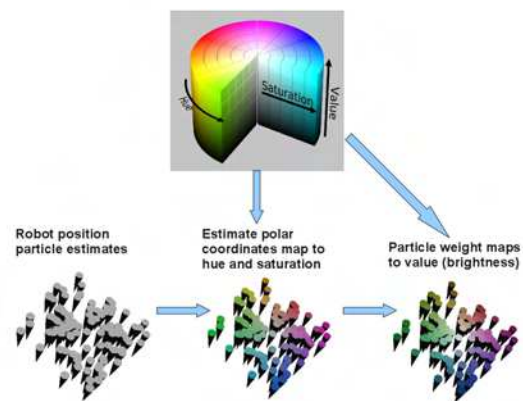


Figure 4: **The colour-mapping technique.**

colour-mapping visualization applied in SLAM. The colour-coded relationship between the robot position and feature errors is clearly visible.

### 5.2.3 Surface Density Visualization

Fig. 6 presents a novel surface density visualization technique developed for FastSLAM. The purpose of this visualization is to present a better overall qualitative picture of the SLAM solution produced by the filter. Here the joint SLAM posterior probability of the robot and the features is represented by a smooth, shaded 3D surface. The mapping area is divided into an uniform (customizable) grid, where the density of a given cell is given by the number of particle members (robot pose or features) within that cell. Then the surface is interpolated over the grid using a *Radial Basis Function (RBF)* [Buh03], with the density giving the surface height. If colour-mapping is enabled, the colour for each cell is the average of the particle colours within it. Otherwise, the cyan surface represents the robot pose and the green surfaces the features. In addition, a single arrow is drawn above each cell of the robot pose surface. This is the average robot orientation within the cell.



Figure 5: **Intra-particle Associations Visualization:** colour-mapping used to show members belonging to the same given particle, showing progression over time. Brightness indicates particle weights.

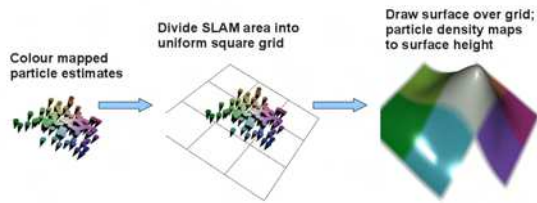


Figure 6: **The surface colour-mapping technique.**

Fig. 7 shows the surface density visualization without the colour mapping. Intuitively the height of the surface indicates the SLAM posterior probability. The shape of the surface provides a good qualitative picture of the uncertainty spread of the distribution, as compared to rendering each individual marker. Fig. 8 shows the colour-mapped surface density visualization. This offers the benefits of both the surface and the colour-mapping techniques. The visualization shows both the shape of the uncertainty spread and the colour-mapped intra-particle associations.

## 6 EVALUATION

### 6.1 Experimental Setup

The visualization system was implemented with a ceiling mounted Prosilica EC 1350C Firewire camera for image capture. Registration was done using ARToolKitPlus and a fiducial marker mounted on the robot. The robot's initial position is registered in the AR coordinate frame as the origin of the SLAM map. This allows the registration of the SLAM data in the AR coordinate frame. Videos were taken of the robots SLAM performance using different visualization techniques for correctly implemented SLAM algorithms and versions with typical errors we inferred from the literature and a previous user survey [Koz11].

### 6.2 Methodology

We performed five experiments summarised in table 1 in order to investigate the effectiveness of the visual-

Fault Detection Experiments		
Experiment	Vis 1	Vis 2
EKF Exp 1	Conventional AR	Conventional non-AR
EKF Exp 2	Correlations AR	Conventional AR
FastSLAM Exp 1	Conventional AR	Conventional non-AR
FastSLAM Exp 2	Colour-mapping AR	Conventional AR
FastSLAM Exp 3	Surface density AR	Conventional AR

Table 1: Fault detection experiments summary. Each experiment compared Vis 1 with Vis 2.

izations for assisting SLAM development. In particular we evaluated AR-based visualisation techniques versus non-AR visualisation techniques and novel AR visualisation versus AR-implementations of techniques considered current state-of-the-art. The purpose of the study was to compare the effectiveness of the visualization techniques for SLAM algorithm development, i.e. fault detection and fault correction.

The experiments were performed as a web-based survey questionnaire. Participants were invited over email, through the major international robotics mailing lists. These included Robotics Worldwide, Australian Robotics and Automation Association (ARAA) and European Robotics Research Network (EURON). Ideally the desired population of the participants would be SLAM developers; but in practice to obtain sufficient participants the population scope was widened to *robotics* developers. The experiments involved participants watching online videos of the visualizations and answering questions about the visualizations.

Within the questionnaire document, the concepts of SLAM and AR were first explained, along with in-

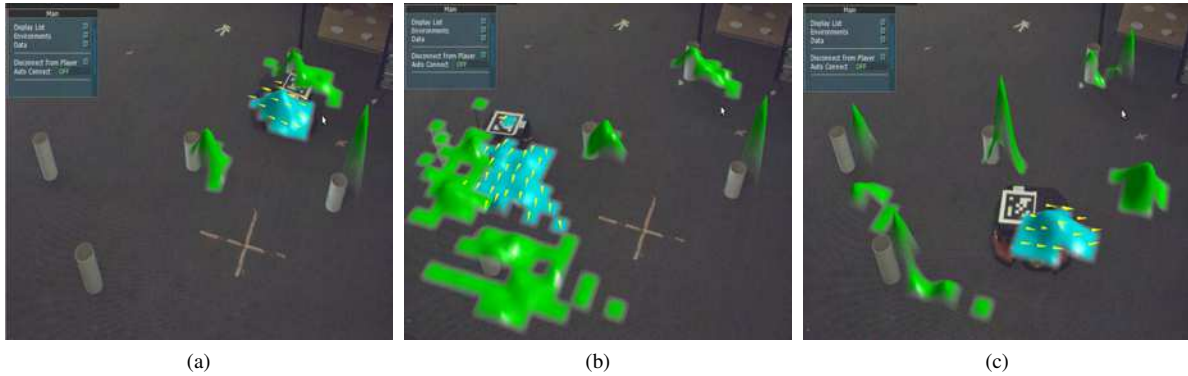


Figure 7: **Surface Density Visualization:** the surface density visualization without the colour-mapping, showing progression over time. The shape and height of the surface conveys the joint SLAM posterior distribution computed by the particle filter.

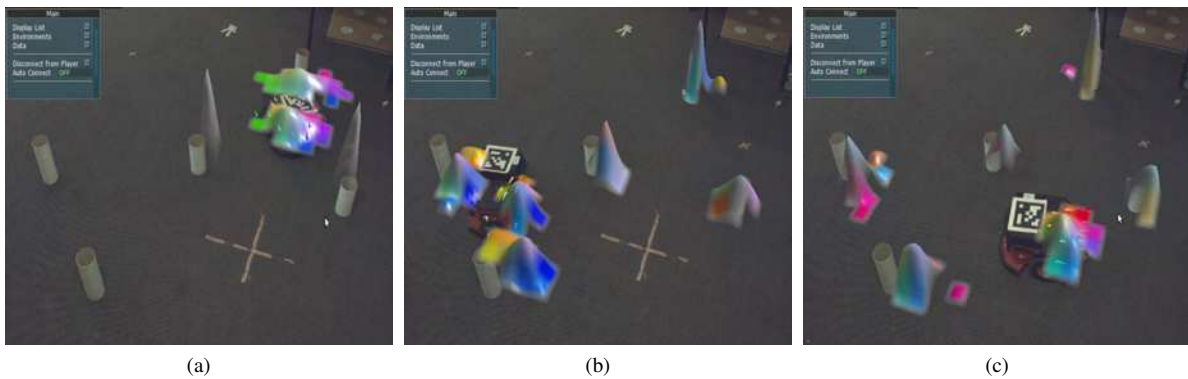


Figure 8: **Colour Mapped Surface Visualization:** the surface density visualization with the colour-mapping, showing progression over time. The visualization shows both the shape of the joint SLAM posterior distribution and the associations within particles.

troductory videos and explanations about the visualizations. Each AR visualization was presented with a video of it being used for SLAM with a real robot and cylindrical point features, along with a written explanation. To present the non-AR visualization, two videos were used. One was the virtual SLAM visualization, and the other was the video of the physical robot performing SLAM corresponding to the SLAM visualization.

After showing correct operation, artificial faults were introduced into the SLAM systems. Within each experiment the same fault was used to compare the visualizations, however the visualizations showed the fault in different ways. For each visualization, the participants were asked as a multi choice question what SLAM fault is present in the visualization (if any). For each pair of visualizations compared, the participants were also asked in a short answer question which visualization they felt was more effective (Vis 1, Vis 2, or neither) and why. Details of the study are found in [Koz11].

### 6.3 Results

There were 24 participants in the EKF evaluation, and 14 participants in the FastSLAM evaluation.

In EKF Exp 1 users detected 75% of errors with the AR visualization and 70% of errors with the non-AR visualization. Users liked that the AR visualization combined a view of the real environment with the SLAM information. Reasons for preferring non-AR were perception difficulties in AR due to the 3D camera perspective, deformation, depth and projection, and the real-world camera image. In FastSLAM Exp 1 all of the participants preferred the AR visualization. In terms of effectiveness both visualisations resulted in 57% of errors being detected.

In EKF Exp 2 our new correlation visualization allowed users to detect 79% of errors, whereas the traditional visualization only allowed detection of 50% of errors. Users liked in the correlation visualization the explicit representation of correlation faults enabling a faster detection. Reasons for preferring the conventional AR visualization were clearer, more intuitive representation of robot pose/landmarks and faults therein, the correlation ellipsoids being hard to understand and occluding the landmark/robot markers, and robot/landmark covariances being more representative of the estimation process.

In FastSLAM Exp 2 users were able to detect 64% of errors using colour mapped particles and 35% of errors using the conventional visualization. Users liked about colour-mapping the clear representation of particle weighting and the resampling process, and that colour mapping offers more information in a compact representation allowing for better fault detection.

In FastSLAM Exp 3 users identified 42% of errors using the surface density visualization and 71% of errors using the conventional visualization. Users liked the compact and effective representation of the particle set distribution in the surface density AR visualization, and that the peak of the surface indicates the most likely estimate position whereas the spread shows the amount of divergence in the estimates. However, users complained that the surface representation is too opaque and obscures the true landmarks, and that the surface view does not show the robot orientation clearly. Users stated that the conventional AR visualization is easier to analyze in order to detect errors.

## 7 CONCLUSIONS

This paper presented novel AR visualization techniques for SLAM algorithms. The visualization requirements were challenging because SLAM algorithms are detailed, complex, and must address real world uncertainties. To address the requirements, visualizations were developed in AR to target the most important aspects of the SLAM algorithms, including feature correlations, particle weights and relationships.

Our Evaluation shows that AR visualizations are preferred over non-AR visualizations, and that the novel techniques for feature correlations is more effective than the existing state of the art for SLAM fault detection. The visualizations are effective because they target specific aspects of the algorithm and because AR enables visualization of the real world and associated uncertainties. The correlation visualization can be adapted to any application requiring representation of correlations between physical entities. Care must be taken that visualization icons do not obscure relevant real-world information in the camera view and that visual complexity does not put undue stress on the user. Hence small point based icons are preferable over more complex and information rich surface representations. The presented visualizations perform differently well for different types of errors. Ideally the user should be able to swap interactively between all of the presented techniques.

## 8 REFERENCES

[Azu97] R.T. Azuma. A survey of augmented reality. *Presence: Teleoperators and Virtual Environments*, 6(4):355–385, 1997.

- [BEFS01] C. Breazeal, A. Edsinger, P. Fitzpatrick, and B. Scassellati. Active vision for sociable robots. *IEEE Trans. Syst., Man, Cybern.*, 31(5):443–453, 2001.
- [BFMG07] J.L. Blanco, J.A. Fernandez-Madrigal, and J. Gonzalez. A New Approach for Large-Scale Localization and Mapping: Hybrid Metric-Topological SLAM. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2061–2067, 2007.
- [BKMY03] D. Blank, D. Kumar, L. Meeden, and H. Yanco. Pyro: A python-based versatile programming environment for teaching robotics. *Journal on Educational Resources in Computing (JERIC)*, 3(4):1–15, 2003.
- [BNG<sup>+</sup>07] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot. Consistency of the EKF-SLAM algorithm. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 3562–3568. IEEE, 2007.
- [BOGH<sup>+</sup>03] V. Brujic-Okretic, J.Y. Guillemaut, L.J. Hitchin, M. Michielen, and G.A. Parker. Remote vehicle manoeuvring using augmented reality. In *International Conference on Visual Information Engineering*, pages 186–189, 2003.
- [BR05] O. Bimber and R. Raskar. *Spatial Augmented Reality : Merging Real and Virtual Worlds*. A K Peters, Limited, 2005.
- [Buh03] M. Buhmann. *Radial basis functions theory and implementations*. Cambridge University Press, 2003.
- [CM06] T.H.J. Collett and B.A. MacDonald. Developer oriented visualisation of a robot program. In *ACM SIGCHI/SIGART Human-Robot Interaction*, pages 49–56, 2006.
- [CM09] T. H. J. Collett and B. A. MacDonald. An augmented reality debugging system for mobile robot software engineers. *Journal of Software Engineering for Robotics*, 1(1):1–15, 2009.
- [DCMP03] M. Daily, Y. Cho, K. Martin, and D. Payton. World embedded interfaces for human-robot interaction. In *Annual Hawaii International Conference on System Sciences*, pages 6–12, 2003.
- [DNC<sup>+</sup>01] M. W. M. Gamini Dissanayake, Paul Newman, Steven Clark, Hugh F. Durrant-Whyte, and M. Csorba. A solution to the simultaneous localization



- and map building (slam) problem. *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION*, 17(3):229–241, 2001.
- [DWB06] H. Durrant-Whyte and T. Bailey. Simultaneous localisation and mapping: Part 1. *IEEE Robotics and Automation Magazine*, 13(2):99–108, 2006.
- [DWRN96] H. Durrant-Whyte, D. Rye, and E. Nebot. Localisation of automatic guided vehicles. In *Robotics Research: The 7th International Symposium (ISRR'95)*, pages 613—625, 1996.
- [GL06] S. Ge and F. Lewis. *Autonomous mobile robots : sensing, control, decision-making, and applications*. Boca Raton, FL CRC/Taylor and Francis, 2006.
- [GVS<sup>+</sup>01] B.P. Gerkey, R.T. Vaughan, K. Stoy, A. Howard, G.S. Sukhatme, and M.J. Mataric. Most valuable player: a robot device server for distributed control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1226–1231, 2001.
- [HD07] Shoudong Huang and Gamini Dissanayake. Convergence and consistency analysis for extended kalman filter based slam. *Robotics, IEEE Transactions on*, 23(5):1036–1049, 2007.
- [KMW09] Alex Kozlov, Bruce MacDonald, and Burkhard C. Wünsche. Covariance Visualisations for Simultaneous Localisation and Mapping. In *Proceedings of the Australasian Conference on Robotics and Automation (ACRA 2009)*, pages 1 – 10, December 2009. [http://www.cs.auckland.ac.nz/~burkhard/Publications/ACRA2009\\_KozlovMacDonaldWuensche.pdf](http://www.cs.auckland.ac.nz/~burkhard/Publications/ACRA2009_KozlovMacDonaldWuensche.pdf).
- [Koz11] Alex Kozlov. *Augmented Reality Technologies for the Visualisation of SLAM Systems*. PhD thesis, The University of Auckland, 2011.
- [LDW91] J.J. Leonard and H.F. Durrant-Whyte. Simultaneous map building and localisation for an autonomous mobile robot. In *Proc. IEEE Int. Workshop Intell. Robots Syst. (IROS)*, pages 1442–1447, 1991.
- [LL98] Pierre Legendre and Louis Legendre. *Numerical ecology*. New York : Elsevier, 1998.
- [MCDFC07] R. Martinez-Cantin, N. De Freitas, and J.A. Castellanos. Analysis of particle methods for simultaneous robot localization and mapping and a new algorithm: Marginal-slam. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2415–2420, 2007.
- [Mon03] M. Montemerlo. *FastSLAM: A factored solution to the simultaneous localization and mapping problem with unknown data association*. PhD thesis, Carnegie Mellon University, 2003.
- [MRT] M. Montemerlo, N. Roy, and S. Thrun. *Carmen, Robot Navigation Toolkit*. Retrieved June 2007 from <http://carmen.sourceforge.net/>.
- [MTKW03] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. Fast-slam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1151–1156, 2003.
- [MW03] I. Mahon and S. Williams. Three-dimensional robotic mapping. In *Australasian Conference on Robotics and Automation (ACRA)*, 2003.
- [NBPLS06] R. Nunez, J.R. Bandera, J.M. Perez-Lorenzo, and F. Sandoval. A human-robot interaction system for navigation supervision based on augmented reality. In *IEEE Mediterranean Electrotechnical Conference*, pages 441–444, 2006.
- [NCE<sup>+</sup>07] A. Nawab, K. Chintamani, D. Ellis, G. Auner, and A. Pandya. Joystick mapped Augmented Reality Cues for End-Effector controlled Tele-operated Robots. In *IEEE Virtual Reality Conference*, pages 263–266, 2007.
- [NCMCT07] J. Neira, J.A. Castellanos, R. Martinez-Cantin, and J.D. Tardos. Robocentric map joining: Improving the consistency of EKF-SLAM. *Robotics and Autonomous Systems*, 55(1):21–29, 2007.
- [NLTN02] P. Newman, J. Leonard, J.D. Tardos, and J. Neira. Explore and return: experimental validation of real-time concurrent mapping and localization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1802–1809, 2002.
- [NTC03] J. Neira, J.D. Tardos, and J.A. Castellanos. Linear time vehicle relocation in SLAM. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 427–433, 2003.

- [QGC<sup>+</sup>09] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng. ROS: an open-source Robot Operating System. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2009.
- [Rob08] ActivMedia Robotics. *PIONEER P3-DX*. Retrieved January 2008 from <http://www.activrobots.com/ROBOTS/p2dx.html>, 2008.
- [SSC90] R. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. *Autonomous Robot Vehicles*, pages 167–193, 1990.
- [WL01] Burkhard C. Wünsche and Richard Lobb. A scientific visualization schema incorporating perceptual concepts. In *Proceedings of IVCNZ '01*, pages 31–36, 2001. [http://www.cs.auckland.ac.nz/~burkhard/Publications/IVCNZ01\\_WuenscheLobb.pdf](http://www.cs.auckland.ac.nz/~burkhard/Publications/IVCNZ01_WuenscheLobb.pdf).