Vision-based Localization for a Swarm of Ground Mini-Robots with Relative and Absolute Measurements

Pascal J. Gohl[1], Javier Alonso Mora[1,2], Roland Siegwart[2] and Paul Beardsley[1]

Abstract—

This paper describes an on-board localization system for a swarm of ground robots. The motivation is in entertainment robotics - each robot conceptually provides one mobile illuminated 'pixel', and the whole swarm is used to create visual effects. The requirements for this application are small robot size, precise localization, and the ability to localize in a dense deployment of robots.

An external sensor would offer a viable approach for localization, but is not suitable because it constrains straightforward system deployment for a variety of situations. Instead we describe an on-board system based on computer vision, fitting each robot with a camera and processor. Localization is achieved by utilizing (a) absolute measurements from known fixed landmarks around the swarm arena, and (b) relative measurements between robots. Robots that are at the center of a dense swarm may be surrounded by neighbors and blocked from viewing landmarks. In this situation, the outer robots detect landmarks, and location information is effectively propagated throughout the swarm via relative measurements between robots. The method builds on previous work in simulation with a central computer and a single Extended Kalman Filter to estimate relative positioning in a swarm. We extend the previous work both by incorporating absolute measurements, and by demonstrating a real implementation with physical robots.

It would require significant on-board compute power to realize this system if the fixed landmarks and neighboring robots were detected using natural features. We bypass this issue by using fiducial markers both as landmarks and as identity markers on robots. This enables the focus of the work to stay on the localization algorithm as the main contribution of the paper, while the computer vision is a self-contained component that could be enhanced later. Experimental results are provided for a swarm of five differentially driven robots.

I. Introduction

The miniaturization of components and reductions in costs has led to a rise in multi-robot applications. A multi-robot entertainment application, in which tens of small mobile robots create a novel kind of display, was presented in [1]. Localization was performed using an overhead camera. However this approach reduces the ability to do ad-hoc deployment in a range of settings. In contrast, this paper provides an on-board, vision-based method for localization. It is based on absolute measurements relative to fixed landmarks plus relative measurements between robots, and is well suited to scenarios where many robots interact and therefore occlusions are frequent in the on-board camera images.

The method builds on the work of Martinelli et al [2] to localize a team of robots using relative measurements. Martinelli's algorithm is based on a single centralized Extended Kalman Filter (EKF), with the states of all robots in a single state vector, and showed good performance in simulation. We extend the work with two contributions - firstly we show the use of absolute measurements relative to fixed landmarks as well as the use of relative measurements between robots; secondly we show real results for a set of physical robots.

[1]Disney Research Zurich
[2]Autonomous System Lab, ETH Zurich

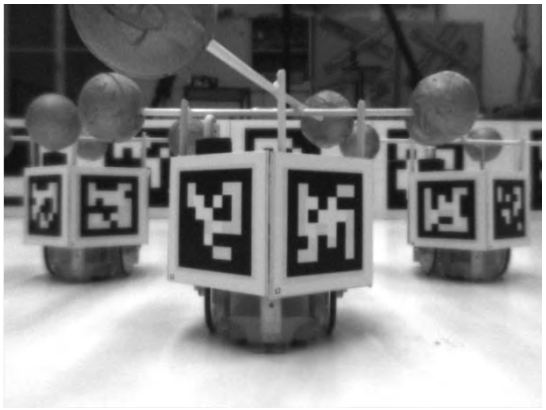

Fig. 1. On-board view of a robots in a crowded situation with occluded environment. The retro-reflective balls on top of the robots are for the ground truth measurement.

This paper is organized as follows. Section II describes currently available localization technology. Section III introduces the hardware system and Section IV shows how the system is modeled. Section V describes our localization algorithm, and Section VI presents the experiments and results. Section VII concludes and gives an outlook on future work.

II. Related Work

This section reviews existing localization methods.

A. Global Positioning System (GPS)

The accuracy of standard GPS is about 2m in many environments. It can be enhanced by a new technique called Real Time Kinematic (RTK). This allows the measurement of position relative to a reference ground station with an accuracy as low as 10cm. This is an outdoor technology.

B. Ultra Wide Band Localization

The Ultra Wide Band (UWB) radio standard and the current development in fast signal analyzing hardware created a completely new method for localization which reaches an accuracy of down to 1cm [3] and is robust against multipath [4]. However the smallest hardware available today is a similar size to our robots and therefore not applicable.

C. Ultrasonic Localization

A distance measurement with an accuracy below 1cm can be achieved with ultrasonic sensors and common electronic components [5]. A relative multi-robot localization system based on ultrasound was developed and tested in [6]. Such a system can easily be extended to fixed beacons for absolute position measurements as needed in this project. But the slow speed of sound also introduces the problem of echoes and the time needed to let them decline. The approach is impractical for a large number of robots as we envisage.

D. Vision

A more modern approach for localization is the use of computer vision. The field of research is facilitated by the current rapid increase in computer power, making it possible to run complex image processing algorithms in real time on high resolution images.

1) Fiducial Marker: This paper uses the ARToolKit [7] library developed at the University of Washington. Many successors have followed [8] [9] with the same approach but increased accuracy and error detection. The use of a special marker make the processing demands small enough to run on board the robot. A second benefit of this approach is the ability to provide a unique identity for each robot.

E. Types of Environment

1) Known Environment: If the environment map is static, localization can be done by using a recursive estimator like a extended Kalman filter. This has the benefit of small and constant computational complexity which allows it to run on-board.

2) Unknown Environment - Simultaneous Localization and Mapping (SLAM): If the map is unknown the SLAM technique can be used to build the map at the same time the robot is being localized. This brings flexibility. The down side is a requirement for high compute power.

III. System Overview

For testing the localization system small diverentially driven E-Puck robots [10] are used.

The on-board computer extension contains a Gumstix board with an ARM Cortex A8 CPU and has enough power to run the ArtoolkitPlus on-board or to compress the image and stream it to an external computer.

To avoid image distortion during fast robot rotations s 1.2Mpx global shutter camera is used. It is connected by USB to the on-board computer and is set to stream images with 7.5Hz.

For the optaining relative distance and orientation measurements, five AR markers are placed on the robots in exactly defined positions. The complete arrangement is shown in Fig. 2.

A central computer runnning the algorithm is used with a two way communication to all the robots. Wifi network is used which allows to stream the images in real time to a desktop computer to run more complex image processing algorithm if needed.

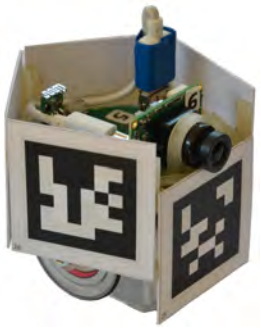

Fig. 2. E-puck equiped with camera and AR-markers

As fiducial marker system the ARToolKitPlus software [8] is used with BCH markers. This id-encoded AR marker system allows up to 4096 uniquely identifyable markers including a CRC error correction algorithm to restore partially covered markers.

IV. Measurement and Motion Model

A. Motion Model

The states x and y define the position and θ the orientation of a robot. Given the previous state and the motor commands the next robot state is predicted by:

$$\begin{aligned}\mathbf{x}_{k|k-1} &= f(\mathbf{x}_{k-1|k-1}, u_{k-1}) \qquad (1)\\ &= \begin{bmatrix} x_{k-1|k-1} + \delta t * u_{v,k-1} * \cos(\theta_{k-1|k-1} + \delta t/2 * u_{\omega,k-1}) \\ y_{k-1|k-1} + \delta t * u_{v,k-1} * \sin(\theta_{k-1|k-1} + \delta t/2 * u_{\omega,k-1}) \\ \theta_{k-1|k-1} + \delta t * u_{\omega,k-1} \end{bmatrix}\\ \theta_{k|k-1} &= (\theta_{k|k-1} + \pi)(\mathrm{mod}\ 2\pi) - \pi. \qquad (2)\end{aligned}$$

Where u_v and u_ω stands for the linear respectively angular velocity commands, and δt as the elapsed time between the two states. The index $k|k-1$ denotes the state k estimated out of the given state $k-1$. In (2) the predicted yaw angle $\theta_{k|k-1}$ is limited to the range $[-\pi, \pi)$.

B. Measurement Model

There are two types of measurement which have to be processed differently. The first and simplest case is when a marker attached to the environment is detected as shown in Fig. 3 on the left in blue. The second case is a relative measurement when one robot is looking at another robot. (Fig. 3 on the right side in red.)

1) Absolute Measurement Model: To model such a measurement a total of three coordinate systems are involved. Where O is the observing robot frame, C the camera frame, W the world frame and M the marker frame.

The first coordinate transformation is the static position of the camera mounted on the robot. It is named T_C^O and measured out of the CAD model of the robot. The position of the observing robot in the world frame is T_O^W. And the marker position and orientation in relation to the map frame T_M^W.

The marker transformations are stored in the environment map database and looked up by the detected marker ID.

$$T_M^C = T_C^{O\,-1} * T_O^{W\,-1} * T_M^W \,. \tag{3}$$

Using these transformations the 3D measurement provided by ARToolKitPlus can be modeled:

$$h_{m,abs} = \begin{bmatrix} z_m \\ x_m \\ \theta_m \end{bmatrix} = \begin{bmatrix} x_{MO}\cos(\theta_O) - x_C + y_{MO}\sin(\theta_O) \\ y_{MO}\cos(\theta_O) - x_{MO}\sin(\theta_O) \\ \theta_{MO} \end{bmatrix} \tag{4}$$

where: $x_{MO} = x_M - x_O$, $y_{MO} = y_M - y_O$, $\theta_{MO} = \theta_M - \theta_O$.

2) Relative Measurement Model: The second case occurs when a marker on another robot is sensed. In a relative robot to robot measurement a total of four affine coordinate system transformations have to be performed. The observing robot frame O and its camera frame C can be used as shown before. The two new frames are the coordinate system of the sighted robot named S and the marker frame M in the sighted robot frame. T_S^W can be calculated using the states of the sighted robot and T_M^S is measured out of the CAD model and stored in a lookup table for all five on-board markers. The complete transformation between the observer camera and the marker on a sighted robot is:

$$T_M^C = T_C^{O\,-1} * T_O^{W\,-1} * T_S^W * T_M^S. \tag{5}$$

Out of this transformation, the measurement vector can be calculated:

$$h_{m,rel} = \begin{bmatrix} z_m \\ x_m \\ \theta_m \end{bmatrix} \tag{6}$$
$$= \begin{bmatrix} x_{SO}\cos(\theta_O) - x_C - y_{SO}\sin(\theta_O) + x_M\cos(\theta_{SO}) + y_M\sin(\theta_{SO}) \\ y_{SO}\cos(\theta_O) - x_{SO}\sin(\theta_O) + y_M\cos(\theta_{SO}) - x_M\sin(\theta_{SO}) \\ \theta_S + \theta_M - \theta_O \end{bmatrix}$$

where: $x_{SO} = x_S - x_O$, $y_{SO} = y_S - y_O$, $\theta_{SO} = \theta_S - \theta_O$.

V. Filter Design

This section describes the design of the EKF. The EKF fundamentals as well as the required design steps that were taken are described in [11].

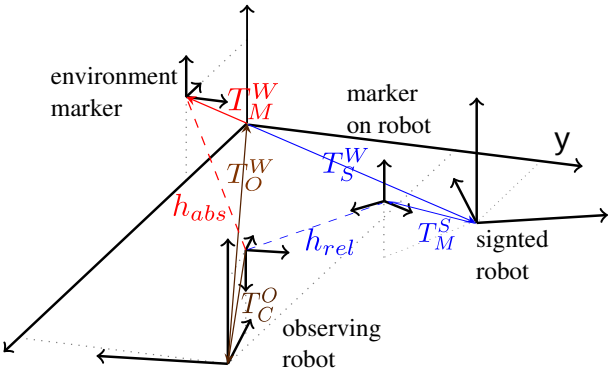

Fig. 3. Coordinate frame transformations for two robots equiped with marker and camera.

The EKF state consists of the states of all r robots concatenated to one large state vector.

$$\mathbf{x}_k = \left[\mathbf{x}_{k,1}, \mathbf{x}_{k,2}, \ldots, \mathbf{x}_{k,r}\right]^\top \tag{7}$$

In the Kalman prediction step the new robot positions $\mathbf{x}_{k|k-1}$ and covariances $\mathbf{P}_{k|k-1}$ are estimated using the motion model of the robot as derived in Equation (1) where k is the discrete time and $\mathbf{u}_{k-1}$ is the robot command which was active during the time step.

$$\mathbf{x}_{k|k-1} = f(\mathbf{x}_{k-1|k-1}, \mathbf{u}_{k-1}) \tag{8}$$
$$\mathbf{P}_{k|k-1} = \mathbf{F}_{k-1}\mathbf{P}_{k-1|k-1}\mathbf{F}_{k-1}^\top + \mathbf{U}_{k-1}\mathbf{Q}_{k-1}\mathbf{U}_{k-1}^\top \tag{9}$$

For the covariance update in (9) the Jacobian of the robot model $\mathbf{F}_{k-1}$ is used to propagate the uncertainty to the current state. The Jacobian of the system model is used to propagate the system error introduced during this step [12]. The Jacobian derivations are shown in Section V-A.

All measurements from all robots z_n' are collected in a measurement vector z with length $n * r$. Where is n the number of measurements (which varies with time), and r the number of robots.

$$\mathbf{z}_k = \left[\mathbf{z}'_{k,1}, \ldots \mathbf{z}'_{k,n}\right]^\top \tag{10}$$

The EKF update equations are used to correct the estimated position using the measurement information received during this time step:

$$\mathbf{x}_{k|k} = \mathbf{x}_{k|k-1} + \mathbf{K}_k(\mathbf{z}_k - h(\mathbf{x}_{k|k-1})) \tag{11}$$
$$\mathbf{P}_{k|k} = (I - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}, \tag{12}$$

where $\mathbf{K}_k$ is the Kalman gain derived as follows:

$$\mathbf{S}_k = \mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^\top + \mathbf{R}_k \tag{13}$$
$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k^\top\mathbf{S}_k^{-1}, \tag{14}$$

where $\mathbf{H}_k$ is the measurement Jacobian of this time step and $\mathbf{R}_k$ describes the error of the measurements.

The Jacobians of each individual measurement are concatenated to $\mathbf{H}_k$ as used in the update equations:

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{H'}_{1,1} & \ldots & \mathbf{H'}_{r,1} \\ \vdots & \vdots & \vdots \\ \mathbf{H'}_{1,n} & \ldots & \mathbf{H'}_{r,n} \end{bmatrix}_k . \tag{15}$$

Each column corresponds to a measurement and the rows to the robots. Depending on the type of measurement $\mathbf{H'}_{i,m}$ is chosen differently for the involved robots. The remaining $\mathbf{H}'_{i,m}$ are set to zero. Therefore, if an absolute measurement m is observed by robot i, the corresponding row results in:

$$\left[0, \ldots, 0, \mathbf{H'}_{i,m}^{observer}, 0, \ldots, 0\right].$$

For or a relative measurement m between robot i and j, the corresponding row results in:

$$\left[0, \ldots, 0, \mathbf{H'}_{i,m}^{observer}, 0, \ldots, 0, \mathbf{H'}_{j,m}^{sighted}, 0, \ldots, 0\right].$$

The different measurement Jacobians are derived in Section V-A.

The matrix $\mathbf{R}_k$ consists of the error covariances of each measurement placed on the diagonal. For each measurement $\mathbf{R'}_{m,m}$ is derived as described in (21). Therefore $\mathbf{R}_k$ is given by:

$$\mathbf{R}_k = \begin{bmatrix} \mathbf{R'}_{1,1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \mathbf{R'}_{n,n} \end{bmatrix}. \tag{16}$$

A. Derivations of the Jacobians

The fundamental principle of the EKF is to linearize the nonlinear system and measurement model at the estimated system state to calculate the Jacobian matrices. In this section the Jacobians are derived for the system model and the measurement model for all the necessary modes.

1) Robot Motion Jacobian: This Jacobian depends only on the robot orientation and the active robot command during this timestep. This is a direct consequence of the simplified robot motion model without taking the inertia into account and is given by

$$\mathbf{F}_{k-1} = \left.\frac{\partial f}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{k-1|k-1},\mathbf{u}_{k-1}}. \tag{17}$$

2) Measurement Jacobians: The partial derivation of the measurement model with respect to the robot states gives the EKF the information in which direction and magnitude a measurement has to correct the robot states. This Jacobians have to be calculated for both measurement types.

The case for an absolute measurement when sensing an environment marker involves only the observing robot and is calculated from (4):

$$\mathbf{H}'^{observer}_{k-1} = \left.\frac{\partial h_{m,abs}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{k-1|k-1},\mathbf{u}_{k-1}}. \tag{18}$$

When a relative measurement between two robots is sensed, both robots gain information and their position can be updated. Therefore for both robots a Jacobian has to be calculated with respect to their states and the measurement model.

For the observing robot the measurement model in 6 is derived with respect to its state:

$$\mathbf{H}'^{observer}_{k-1} = \left.\frac{\partial h_{m,rel}}{\partial \mathbf{x}_{observer}}\right|_{\hat{\mathbf{x}}_{k-1|k-1},\mathbf{u}_{k-1}}. \tag{19}$$

For the sighted robot the same measurement model (6) can be used but derived with resphas to be partiallyect to the sighted robot:

$$\mathbf{H}'^{sighted}_{k-1} = \left.\frac{\partial h_{m,rel}}{\partial \mathbf{x}_{sighted}}\right|_{\hat{\mathbf{x}}_{k-1|k-1},\mathbf{u}_{k-1}}. \tag{20}$$

B. Time Delay Handling

Considering the relative big amount of data the on-board computer of the robots has to handle and the unstable nature of the Wifi connection (delays caused by disturbances in the crowded frequency band) it becomes clear that time delays are not negligeable. In fact the lag between a picture is taken by the on-board camera and the final measurement output from the ARToolKitPlus, including transmission over Wifi, is around 150ms. If the Wifi gets disturbed during the transmission, it can become easily 300ms.

To handle these lags in the sensor data a buffered Kalman filter structure is designed. Therefore a defined number of system states are saved together with a time stamp and each image taken by the camera is marked with an exact time stamp. After sending and processing the image, the measurement is assigned to the corresponding state by searching for the closest time stamp. To use this delayed measurements, the next Kalman filter run has to start at the latest measurement and propagate the new information up to the current system state. Fig. 4 shows a schematic of the EKF state buffer with delayed measurements.

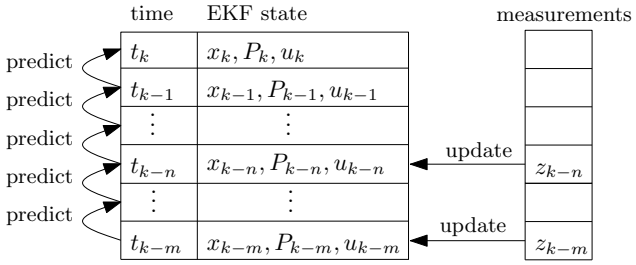

Fig. 4. Function scheme of the filter buffer and the information propagation from delayed measurement to the current state

VI. Experiments

In order to record the ground truth of the robots the motion capture system built by Vicon[1] has been used. Fig. **??** shows five e-puck robots equipped with retro-reflective balls for the external ground truth measurement system driving a eight figure. The parameters of the robot and the system are shown in Table I. Localization is solely performed by on-board vision.

TABLE I
E-puck System Data

wheel radius	$r_w = 0.0205[m]$
wheel separation	$d = 0.053[m]$
motor steps per revolution	$steps = 20$
gear reduction	$gr = 1/50$
max steps per seconds	$steps_{max} = 1000[s^{-1}]$
max drive speed	$v_{max} = 0.129[m/s]$
max rotation speed	$r_{wmax} = 0.129 * 2/d = 4.86[rad/s]$

A. Sensor Error Identification

A robot was driven arround randomly by hand and in an precomputed eight figure and the data where recorded. The result of the data analysis showed no big influence of the distance to the marker (z_m) on the measurement error for you small distances of our setup. The error of x_m grows for less than 10% when comparing a measurement distance to

[1] http://www.vicon.com

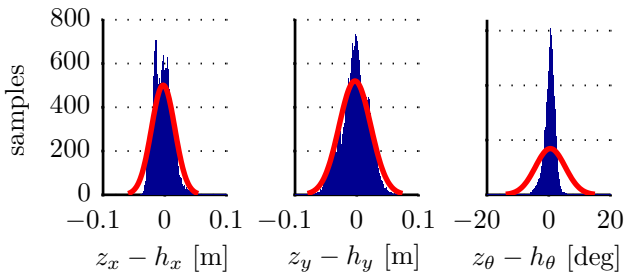

Fig. 5. The error histograms of all three measurement variables and the overlayed gaussian approximations calculated by MATLAB. The measured variances are σ_z: 0.026, σ_x: 0.029 and σ_θ: 0.508

the marker from 0.5m to 1.5m. In the marker orientation measurement the major errors also have no dependency on the distance, but there are a few outliers which grow linearly with the distance. More details can be found in [13].

Considering the negligible influence of the distance it was decided to use a static measurement error model for the marker sensory system.

$$R_m = \begin{bmatrix} \sigma_z^2 & 0 & 0 \\ 0 & \sigma_x^2 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{bmatrix} \tag{21}$$

As initial parameter for the error model the measurement errors of the experiment were fitted to the approximated gaussian as shown in Fig. 5.

B. Single Robot

To compare the single robot localization performance of the filter, an eight trajectory was driven at two different speeds. One run was driven with the velocity of $0.06m/s$ which is about 50% of the maximum speed the robot can drive. The same eight was driven with $0.08m/s$ to find out how the odometry quality affects the position estimation. For the system error model the parameters k_s and k_r are set to $2 \cdot 10^{-4}$ respectively $2 \cdot 10^{-3}$, which were found by tuning when driving an eight at $0.06m/s$ speed.

The scatter plot in Fig. 6 for the fast run shows the difficulties the estimator had during the curve on both ends of the eight. This is a direct consequence of the odometry assumptions which do not match the reality any more in this velocity range because of wheel slip during curves and stepper motor slips during rapid velocity changes.

In Fig. 7 the position and orientation errors calculated out of the recorded estimated position and the ground truth are shown. The position error of the slow run is between $5mm$ and $7cm$, for the fast run the position error lies between $4cm$ and $13cm$. The difference is even bigger in the orientation error. For the slow run the error lies between $-4°$ and $3°$, for the fast run it increases to $-12°$ and $10°$.

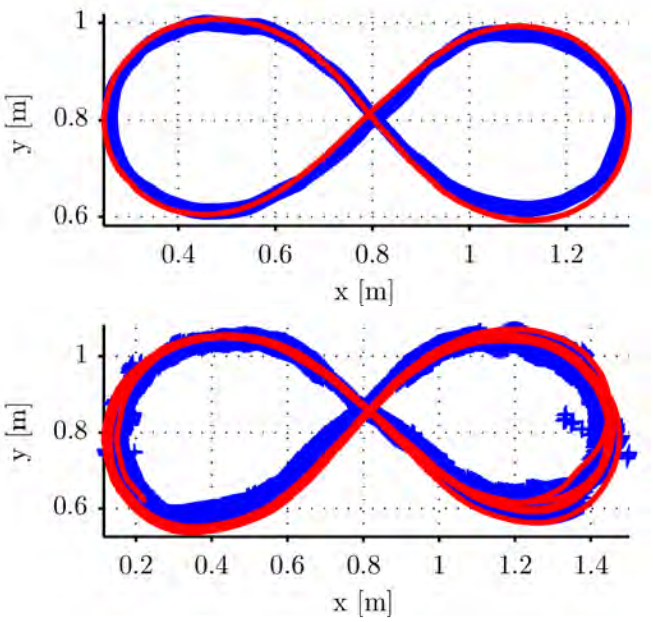

Fig. 6. Estimated (blue) and true (red) position of a robot driving an eight at different speeds. Top $0.06m/s$ and bottom $0.08m/s$.

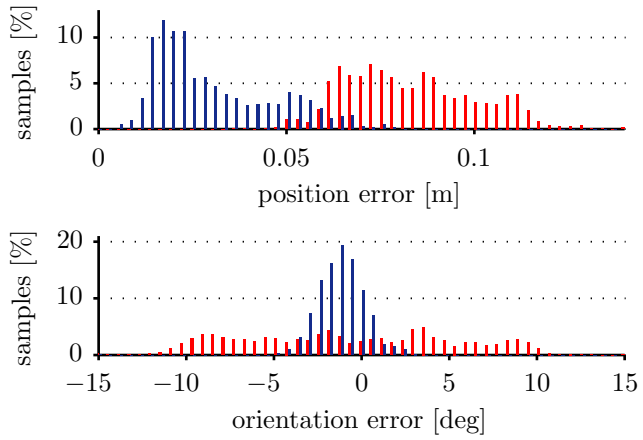

Fig. 7. Localization error while driving multiple eights with one robot at a speed of $0.06m/s$ (blue) and $0.08m/s$ (red)

The main part of the position error is caused by the wrong odometry assumptions, especially the orientation error. Furthermore there is another error source which is strongly dependent on rotational velocity. It is the shutter timing of the camera which cannot be controlled by the camera driver and therefore the timestamp of the image is not always absolute precisely set. For example when rotating with $0.4rad/s$ and given an image delay of one frame at 7.5Hz can result in an image which is taken at a robot orientation of up to 3° before the moment of the estimators assumption.

C. Multiple Robots

To measure the benefit of relative measurements in non clustered trajectories, the same eight as before was driven

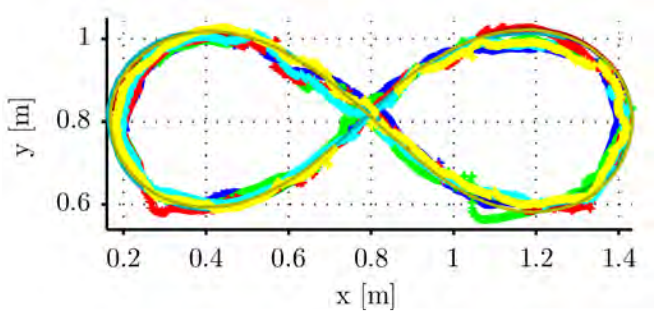

Fig. 8. Path of five robots while driving multiple eights at velocity of 0.06m/s. Each path in one color.

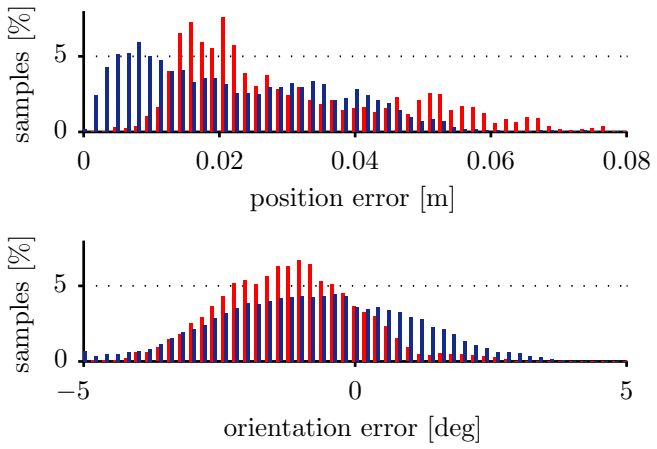

Fig. 9. Comparison of the localization error for one (red) and five (blue) robots driving an eight at 0.06m/s

with five robots. This allowed relative measurements in the middle of the eight during the crossing. Again all the estimated positions and orientation as well as the ground truth measured by Vicon were recorded. The robots were driving with the calibrated speed of $0.06m/s$. Fig. 8 shows the scatter plot of the estimated and true position of all five robots driving five times the complete eight.

This experiment is shown in the accompanying video. Where the view of one of the on-board cameras of a robots is displayed as well as the real time position erstimation and its covariance matrix.

The resulting position and orientation estimations are displayed in Fig. 9 overlayed with the data recorded during the slow run. The algorithm showed a reasonable good performance. The average error of all robots over all driven eight is at $2.02cm$ and the average orientation error at $1.56°$ degree.

Comparing the two identical trajectories with one or five robots in Fig. 9 a decrease of the position and orientation error is visible. The average position error decreases from $2.90cm$ to $2.02cm$ and the orientation error from $1.86°$ to $1.56°$. This is a direct effect of the additional relative measurement.

VII. CONCLUSION

This paper has described an on-board localization method for a set of mobile robots using computer vision. The presented algorithm was developed for an application in entertainment robotics in which mobile robots are deployed on a planar surface to create images. Fixed landmarks are available around the robot arena to provide a known coordinate frame. The localization method is able to propagate localization information throughout the swarm, including to robots which see their neighbors but which are occluded from seeing the fixed landmarks. This is a suitable solution for large sets of robots where occlusions are frequent. The computer vision component uses fiducial markers, but a more sophisticated vision system could be readily subsituted in future work.

The algorithm uses a central computer, which introduces known problems of scaling to large sets of robots. This could be addressed by using a decentralized EKF. The main challenges of such an approach are (a) to decide what data is to be communicated between robots and what can be neglected and (b) to avoid the need for each robot to need the state and measurements of every other robot. These challenges are suitable for future work.

REFERENCES

[1] J. Alonso-Mora, A. Breitenmoser, M. Rufli, R. Siegwart, and P. Beardsley, "Image and animation display with multiple robots," *Int. Journal of Robotics Research*, vol. 31, pp. 753–773, May 2012.

[2] A. Martinelli, F. Pont, and R. Siegwart, "Multi-robot localization using relative observations," in *International Conference on Robotics and Automation, Barcelona, Spain*, 2005.

[3] M. Tüchler, V. Schwarz, and A. Huber, "Accuracy of an uwb localization system based on a cmos chip," in *WPNC'05*, 2005.

[4] N. O. Tippenhauer and S. Capkun, "Uwb-based secure ranging and localization," tech. rep., ETH Zürich, 2008.

[5] S. J. Kim and B. K. Kim, "Dynamic localization based on ekf for indoor mobile robots using discontinuous ultrasonic distance measurements," in *International Conference on Control, Automation and Systems*, 2010.

[6] L. E. Navarro-Serment, C. J. J. Paredis, and P. K. Khosla, "A beacon system for the localization of distributed robotic teams," tech. rep., Carnie Mellon University, 1999.

[7] H. Kato and M. Billinghurst, "Marker tracking and hmd calibration for a video-based augmented reality conferencing system," tech. rep., University of Washington and Hisoshima City University, 1999.

[8] D. Wagner and D. Schmalstieg, "Artoolkitplus for pose tracking on mobile devices," tech. rep., Graz University of Technology, 2007.

[9] M. Fiala, "Artag, a fiducial marker system using digital techniques," tech. rep., National Research Council of Canada, 2005.

[10] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J. christophe Zufferey, D. Floreano, and A. Martinoli, "The e-puck, a robot designed for education in engineering," in *In Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, pp. 59–65, 2009.

[11] H. Durrant-Whyte, "Introduction to estimation and the kalman filter," tech. rep., Australian Centre for Field Robotics; The University of Sydney NSW 2006, 2001.

[12] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*. The MIT Press, 2010.

[13] P. J. Gohl, "Localization of a swarm of mini-robots," Master's thesis, ETH Zürich, 2012.