Tell me about dynamics! Mapping velocity fields from sparse samples with Semi-Wrapped Gaussian Mixture Models

Tomasz Piotr Kucner, Martin Magnusson, Erik Schaffernicht, Victor Hernandez Bennetts, Achim J. Lilienthal
Center of Applied Autonomous Sensor Systems (AASS),
Örebro University,
Sweden.
firstname.lastname@oru.se

***Abstract*—Autonomous mobile robots often require information about the environment beyond merely the shape of the work-space. In this work we present a probabilistic method for mapping *dynamics*, in the sense of learning and representing statistics about the flow of discrete objects (e.g., vehicles, people) as well as continuous media (e.g., air flow). We also demonstrate the capabilities of the proposed method with two use cases. One relates to motion planning in populated environments, where information about the flow of people can help robots to follow social norms and to learn implicit traffic rules by observing the movements of other agents. The second use case relates to Mobile Robot Olfaction (MRO), where information about wind flow is crucial for most tasks, including e.g. gas detection, gas distribution mapping and gas source localisation. We represent the underlying velocity field as a set of Semi-Wrapped Gaussian Mixture Models (SWGMM) representing the learnt local PDF of velocities. To estimate the parameters of the PDF we employ a formulation of Expectation Maximisation (EM) algorithm specific for SWGMM. We also describe a data augmentation method which allows to build a dense dynamic map based on a sparse set of measurements. In case only a small set of observations is available we employ a hierarchical sampling method to generate virtual observations from existing mixtures.**

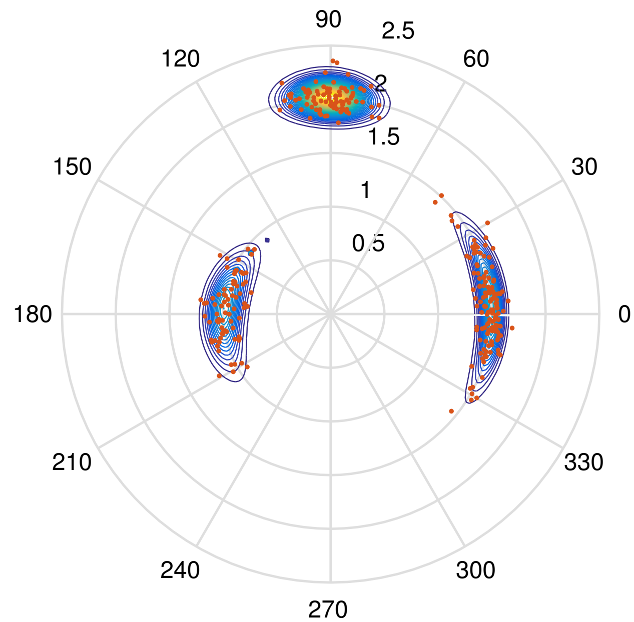

Fig. 1. Polar plot of Semi-Wrapped Gaussian Mixture Model (SWGMM) of synthetic, circular-linear data.

I. Introduction

A. Motivation

Many robotics applications can benefit from maps that go beyond mere occupancy and instead explicitly model the dynamics in an environment. The range of applications for such maps is broad, from Human Robot Interaction (HRI) to Mobile Robot Olfaction (MRO) problems. Statistics describing dynamics of people is important for HRI while navigating in a populated environment [7]. Such information can enhance planning and navigation of an autonomous system, by making it possible to plan around areas that are known to be busy, or not to go against the expected flow of people. It will allow the robot to comply to social norms and operate in an unobtrusive way. The improvement in planning and navigation will not only result in higher and more human friendly performance, but also increase the safety while executing tasks. Both high performance and safety are crucial for any commercial robotic system working in vicinity of people. In the field of MRO and environmental monitoring, statistics describing dynamics can provide a better understanding about gas distribution. Gas plumes are heavily influenced by the local airflow. However the task of wind flow modelling in small scales is yet to be solved. Probabilistic modelling of wind flow can improve gas distribution prediction and allow to improve gas source localisation techniques.

In this paper, we introduce Circular Linear Flow Field map (CLiFF-map), a general purpose method for mapping of dynamics. In this paper we define dynamics as motion within the environment. CLiFF-map consists of:

- a discrete representation of a continuous velocity field,
- a method combining Mean Shift and Expectation Maximisation for parameter estimation of Semi-Wrapped Gaussian Mixture Models (SWGMM),
- an interpolation method for filling in unobserved regions of the map in an informed way.

To each location in the map we associate an SWGMM [16] describing the PDF of velocities in a local neighbourhood. Any method providing explicit velocity measurements can be used to build the map. In this way, the method is independent of any particular representation. CLiFF-map computes the PDF for each location independently, and therefore allows to build the map using partial, incomplete data. Moreover the proposed interpolation technique allows to reconstruct missing data.

B. Related Work

In the past decades, multiple methods were developed to tackle the problem of map building for mobile robots [19]. There are numerous robust methods to deal with noisy data, where dynamics are treated as faulty measurements [21, 23, 6, 2]. These algorithms aim to remove measurements caused by moving objects, and filter them out from the final maps. Such maps can be useful for localisation and global path planning. However, in this way we are limiting our knowledge about the environment to only the statically stable parts, such as walls and fixed objects.

In recent years, a number of approaches explicitly addressing mapping of dynamics in the environment have been developed. We can split these methods into two groups. The first group is tied to a grid map representation of the environment [13]. These methods try to learn from a history of past observations how the occupancy likelihood has changed [1, 12]. The Temporal Occupancy Grid (TOG) introduced by Arbuckle et al. [1] is a layered occupancy grid map. Each layer incorporates the set of measurements within a specific time window, up to the most recent ones. Arbuckle et al. used TOG to classify the dynamics (if both the long term and medium term maps are empty but the short term map is occupied that means we observe a dynamic object). The major limitation of this approach is memory complexity. It requires storing all observations up to the longest time scale. Mitsou et al. [12] follow the same paradigm. They store the full history of occupancy changes for each cell in the map. This representation is especially interesting as a method of storing and accessing the data for further analysis. However, its memory requirements grows without bound over time. The major limitation of these methods is the memory complexity of the representations and the fact that it is impossible to reason about possible future states of the environment. To address these problems there are methods that aim to learn spatiotemporal relation between subsequent states of the map [22, 8, 17]. The problem of reasoning about future states of the environment is addressed by Saarinen et al. [17] and Luber et al. [9]. All these approaches aim to learn the dynamics on the cell level, implicitly stating that the state of each cell does not depend on the state of the neighbourhood. This assumption is relaxed in the work of Wang et al. [22] and Kucner et al. [8]. These methods build models that can grasp the spatial relation between the state of adjacent cells. The aforementioned methods treat dynamics as a change of the occupancy of the cells in the map. In contrast CLiFF-map models the velocity field of the environment and how it affects the motion of objects.

The second group of methods come from the computer vision community [3, 14]. They require information about the complete trajectory of an object as input. The input trajectories are clustered and classified. The major limitation is that such methods cannot work with inputs that only provide velocity estimates. Moreover, they assume that the input data consist of all possible or pertinent trajectories and that each one of those was fully observed from source to the goal. This assumption makes it impossible to build maps from incomplete data.

The aforementioned methods perform well in their respective fields. However, the limitations on the type of input data make it impossible to employ them to any other task.

As we have already mentioned, dynamics of discrete objects is often considered while planning and executing tasks autonomously. However, there are research areas requiring information about other kinds of motion patterns. Currently in the field of Mobile Robot Olfaction (MRO), micro-scale wind mapping is an active research topic. Probabilistic modelling of wind flow can aid the trajectory planning of the robot during exploration and improve the gas distribution model [15].

Wind is a flow of a continuous medium. Therefore, none of the aforementioned methods used for mapping motion of discrete objects can be used for wind mapping. There are a number of algorithms coping with wind mapping at macro (distance up to 5000 km) and meso (distances up to 200 km) scales. However, the problem of wind mapping at micro (distances below 2 km) scale is crucial for MRO and has not yet been fully addressed. The CLiFF-map method presented in this paper allows to build a model of local wind flow.

C. Outline

The remainder of the paper is organised as follows. In Section II, we describe the SWGMM and present a specific formulation of EM for learning the parameters of the model. The same section contains an interpolation method for sparse data. In Section III, we show and discuss results for people tracking and wind data. Finally in section IV, we summarise our work and present directions for future work.

II. Algorithm

We express the velocity of a point size object as a point in a two-dimensional continuous state space $\vec{V} = (\theta, \upsilon)^T$ where $\upsilon \in \mathbb{R}^+$ denotes the speed and $\theta \in [0, 2\pi)$ is the direction. In the chosen representation each of the components is meaningful in a physical way. Representing velocity in polar coordinates has the advantage of fitting well to the "banana-shaped" distribution of motion that appears when the variance in rotation is high. A bivariate Gaussian in Cartesian (x, y) coordinates would need to be excessively inflated in order to fit such a distribution. The direction is a circular quantity, whose PDF can be modelled as a noncircular PDF that has been wrapped around the unit circle. That is, the PDF of the wrapped variable $\theta = \varphi \mod 2\pi$ is $p_w(\theta) = \sum_{k=-\infty}^{\infty} p(\theta + 2k\pi)$ where k is an integer winding number. A wrapped PDF is a probability distribution describing data on an n-dimensional sphere. In one dimension a wrapped distribution consists of points on the unit circle and can be visualised as "wrapping" a continuous distribution around the circle, while the winding number denotes the current revolution around the unit circle. Thus, the wrapped normal distribution is:

$$\theta \sim \mathcal{N}^{\mathcal{W}}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{k=-\infty}^{\infty} \mathcal{N}(\boldsymbol{\mu} + 2k\pi, \boldsymbol{\Sigma}) \tag{1}$$

Since $\theta \in [0, 2\pi)$, we have

$$\int_0^{2\pi} \mathcal{N}^{\mathcal{W}}(\theta|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = 1 \tag{2}$$

Velocity, as we have already mentioned, is a heterogeneous quantity, We build its probabilistic model by wrapping only the directional component. Thus, its density is given by:

$$\vec{V} \sim \mathcal{N}^{\mathcal{SW}}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{k=-\infty}^{\infty} \mathcal{N}(\boldsymbol{\mu} + 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}, \boldsymbol{\Sigma}) \tag{3}$$

In Equations (1) and (3) we can see that to model the wrapped distribution it is required to use an infinite sum. However Mardia and Jupp [11] comment that, for practical purposes, the density can be approximated by truncation to $k \in \{-1, 0, 1\}$ for cases where $\Sigma > 2\pi$ and for $\Sigma < 2\pi$ the term with $k = 0$. Equation (3) defines a uni-modal multivariate semi-wrapped normal distribution. To model multi-modal phenomena, such as the wind or pedestrian flow velocity, we define a mixture of J semi-wrapped normal distributions.

$$\vec{V} \sim p(\Xi) = \sum_{j=1}^{J} \pi_j \mathcal{N}^{\mathcal{SW}}(\boldsymbol{\xi}_j) \tag{4}$$

where π_j ($0 < \pi_j < 1$ and $\sum_{j=1}^{J} \pi_j = 1$) are the mixing factors, and $\boldsymbol{\xi}_j$ is the set of the parameters describing the jth component, namely $\boldsymbol{\xi}_j = (\boldsymbol{\mu}, \boldsymbol{\Sigma})_j^T$. The symbol $\Xi = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_J, \pi_1, \ldots, \pi_J)$ denotes the complete set of parameters of the multimodal distribution.

1) Parameter estimation with EM: To estimate the parameters of the SWGMM, we use Expectation Maximisation (EM) [5]. For the sake of brevity, we omit the derivation of update rules, which for the general, n-dimensional case can be found in the work of Roy and Puri [16]. For the 2D case we consider here, the update rules look as follows.

a) Expectation Step:

$$\eta_{ijk}^t = \frac{\pi_j^{t-1} \mathcal{N}\left(\vec{V}_i; \boldsymbol{\mu}_j^{t-1} + 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}, \boldsymbol{\Sigma}_j^{t-1}\right)}{\sum\limits_{j=1}^{M} \sum\limits_{k=-\infty}^{\infty} \pi_j^{t-1} \mathcal{N}\left(\vec{V}_i; \boldsymbol{\mu}_j^{t-1} + 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}, \boldsymbol{\Sigma}_j^{t-1}\right)} \tag{5}$$

In the expectation step, we compute the *responsibility* η that cluster j takes for the ith data point for the kth round of wrapping, based on the parameters estimated in the previous iteration of the algorithm.

b) Maximisation step: In the maximisation step, we compute the new set of parameters Ξ using the following update rules.

$$\pi_j^t = \frac{1}{N} \sum_{i=1}^{N} \sum_{k=-\infty}^{\infty} \eta_{ijk}^t \tag{6}$$

$$\boldsymbol{\mu}_j^t = \frac{\sum\limits_{i=1}^{N} \sum\limits_{k=-\infty}^{\infty} \left(\vec{V}_i - 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}\right) \eta_{ijk}^t}{\sum\limits_{i=1}^{N} \sum\limits_{k=-\infty}^{\infty} \eta_{ijk}^t} \tag{7}$$

$$\boldsymbol{\Sigma}_j^t = \frac{\sum\limits_{i=1}^{N} \sum\limits_{k=-\infty}^{\infty} \left(\vec{V}_i - \boldsymbol{\mu}_j - 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}\right) \left(\vec{V}_i - \boldsymbol{\mu}_j - 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}\right)^T \eta_{ijk}^t}{\sum\limits_{i=1}^{N} \sum\limits_{k=-\infty}^{\infty} \eta_{ijk}^t} \tag{8}$$

2) Mean Shift for EM initialisation: EM has to be initialised with a set of initial modes, which are then modified to fit the data when running the algorithm. We employ Mean Shift [4] as a mode seeking algorithm to obtain the number and initial positions of modes and covariances.

Mean Shift treats each data point as the mean of its neighbourhood. The neighbourhood is defined as all the points within a given window. In each step the algorithm computes a new value of the mean based on the shape and size of the window and then shift the centre of the window to the computed mean. In this way we obtain maxima of the underlying density function and clusters corresponding to each maximum. We define the window as an isotropic Gaussian whose bandwidth is estimated with Silverman's rule [18]

$$\sigma = \left(\frac{4\hat{\sigma}^5}{3N}\right)^{\frac{1}{5}} \tag{9}$$

where $\hat{\sigma}$ is the standard deviation and N is the number of samples for the whole data set. We can see that Equation (9) requires the data to be univariate. To satisfy this constraint we express each point as its distance from the origin of the coordinate frame. In Eq (10) we define a distance between two points denoting some velocities.

$$d_{1,2}^w = \sqrt{(\theta_1 \ominus \theta_2)^2 + (\upsilon_1 - \upsilon_2)^2} \tag{10}$$

$$\theta_1 \ominus \theta_2 = \pi - \{(\theta_1 - \theta_2) - 2\pi \cdot [(\theta_1 - \theta_2) \mod 2\pi]\} \tag{11}$$

Ttherefore the distance from the origin of the coordinate frame is:

$$d^w(\vec{V}) = \sqrt{(\theta)^2 + (\upsilon)^2} \tag{12}$$

Thus the sample mean we use is

$$m(\vec{V}) = \frac{\sum\limits_{i=1}^{N} \mathcal{N}(d^w(\vec{V})|0, \sigma)\vec{V}_i}{\sum\limits_{i=1}^{N} \mathcal{N}(d^w(\vec{V})|0, \sigma)} \tag{13}$$

In Fig 2 we can see the initial set of clusters built with Mean Shift, while in Fig 3 we can see the resulting PDF after applying EM initialised with these clusters. We can see that clusters estimated by Mean Shift are symmetric and sometimes smaller than expected. The biases on size and shape of the clusters are artefacts coming from the size and shape of the used window. Applying EM helps to remove these biases and also remove unnecessary clusters, which might be created if the size of the window used by Mean Shift was too small.

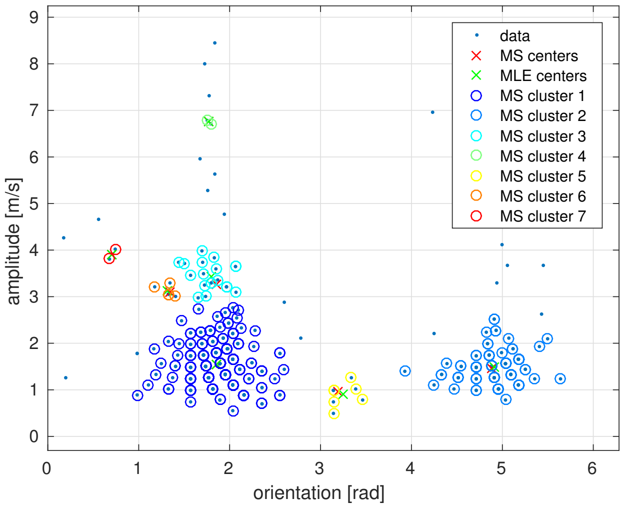

Fig. 2. Clusters obtained using Mean-Shift algorithm for one of the locations.

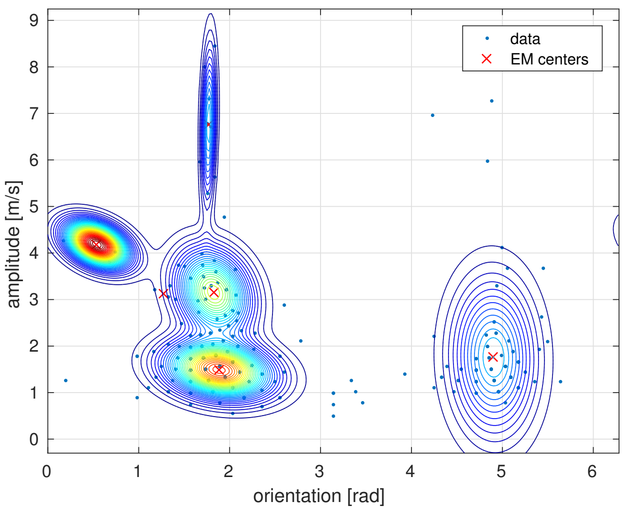

Fig. 3. Distribution obtained using EM algorithm for one of the locations.

Comparing Figures 2 and 3 we can see an example of such a change in shape and size for clusters 4 and 7.

3) Interpolation: Until now we have described how to build a map of dynamics using dense sets of observations. However it is not always feasible to collect measurements for all relevant locations. One example of such cases are wind measurements. Building dense enough sensor networks for wind measurements is too expensive or unfeasible in practice. Similar problems might occur also when mapping the dynamics of discrete objects, when some parts of the environment are constantly occluded during data collection. To address such cases we extend our framework with an interpolation method based on data augmentation.

Let's denote a location as $l = (x, y)$ and associate each with a SWGMM denoted as Ξ. The set of sparsely distributed measurement locations is denoted as $L^M = \{l_1^M, \dots, l_N^M\}$. Our aim is to compute Ξ for each element of a set of densely distributed locations $L^A = \{l_1^A, \dots, l_K^A\}$. (As an example, L^M is shown in Figure 8 as green dots, and Figure 9 shows L^A for the same scenario.)

To estimate Ξ for l_i^A we build a set of virtual observations based on L^M using hierarchical sampling. We associate a weight with each element of L^M based on its distance to l_i^A:

$$w_{l_i^A}(l_j^M) = \mathcal{N}(|l_1^M - l_1^A|; 0, \rho) \tag{14}$$

where ρ denotes the size of the neighbourhood, we want to consider. In the first step we sample an element of L^M according to its weight as in Equation (14). Afterwards we sample a virtual measurement from the corresponding distribution Ξ. We repeat these two steps until we obtain the desired number of samples. Then we fit an SWGMM as described in Section II. We repeat this process for each l^A.

Our method treats each sample as equally important and is not modelling the uncertainties in the estimation directly. It is appealing to apply Gaussian Process here. However, using GP introduces new limitations. In the first place we loose the ability to model multimodal distribution. To each location we will be able to associate only one mode with increased covariance matrix. Such limitation will significantly decrease the flexibility of our method or even lead to wrong estimates (e.g. instead of multiple modes pointing in different directions we will obtain a distribution close to uniform).

III. Results

To demonstrate the capabilities of the method for learning and representing motion maps that was outlined in Section II we have conducted experiments using people tracks from overhead camera images [10] and wind data measured with an anemometer mounted on a mobile robot [20]. For the wind data, there is a set of measurements associated with measurement locations where robot stopped during data collection while for the pedestrian dataset measurements are distributed unevenly over the considered area. To build a model of the flow fields we have split data according to its locations. First we have built a set of points of interests. Second to each point we have associated neighbouring measurements (which are within the radius of 0.5 m) as if the pedestrians passed through the point.

A. People tracking data

In the camera data set, velocity estimates are not available explicitly. We have computed them assuming that the velocities between subsequent detections are constant. In this way we have obtained a set of observations distributed over the map. Figure 4 shows the obtained tracks. The colour corresponds to the direction of the motion. In Fig 4 we can observe paths followed by the pedestrians. The dominant paths are: A-E, A-D, B-C, E-D, E-C, C-D. If we compare the learned motion directions in Fig 5 we can see that aforementioned patterns are still visible.

In Fig 5 we can see a high variance in the mapped velocities. It looks like people tend to run between points A and E and move rather slowly between the points D and C. This impression comes from the fact that we are not visualising the mixing factors of the modes in the SWGMMs. However,

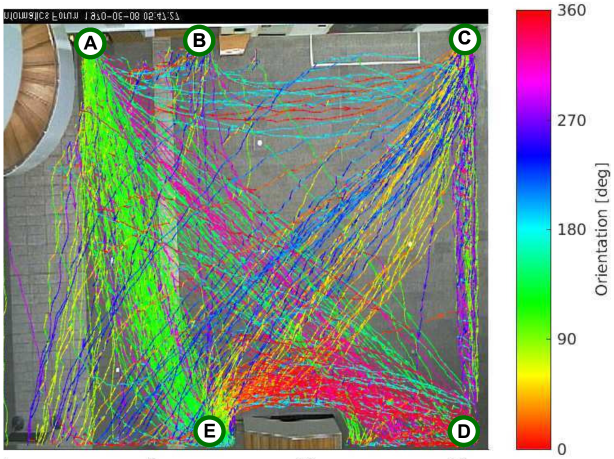

Fig. 4. Tracks extracted from the raw tracking data, used as input to CLiFF-map.

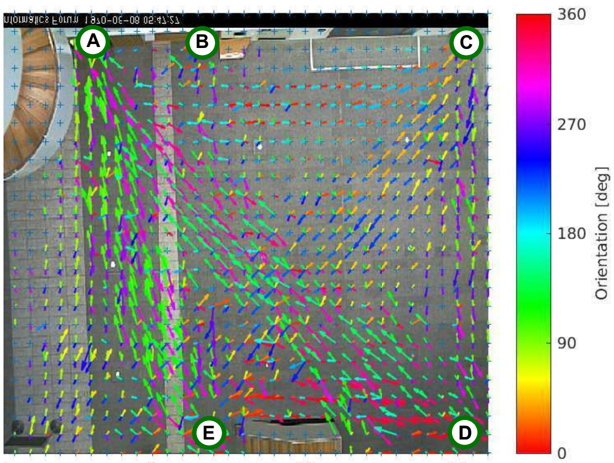

Fig. 5. Map including all the learned motion directions. The distance between nodes is 0.5 m.

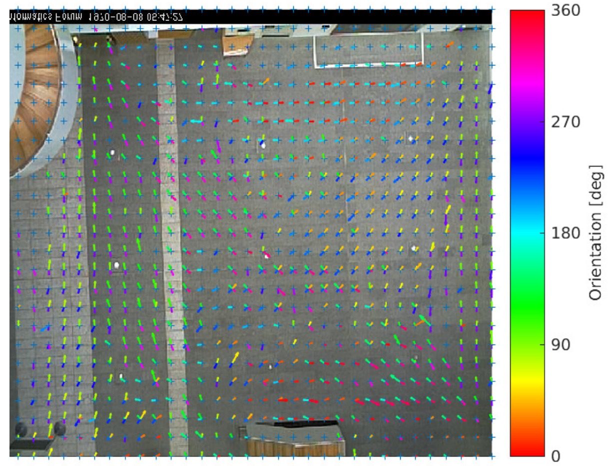

Fig. 6. Map including the learned motion directions whose mixing factor is higher than 0.1. The distance between nodes is 0.5 m.

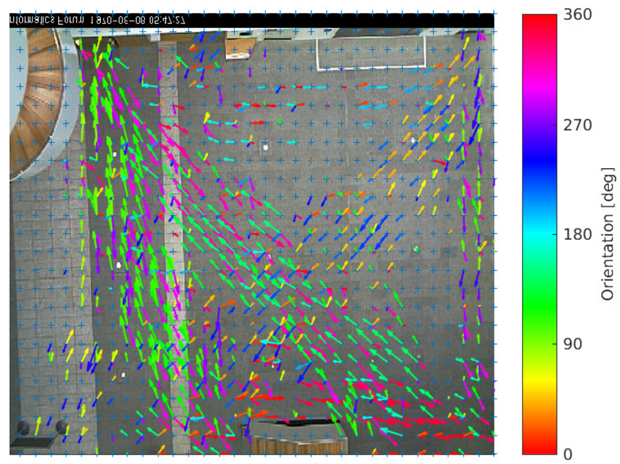

Fig. 7. Map including the learned motion directions whose mixing factor is lower than 0.1. The distance between nodes is 0.5 m.

if we split the modes based on the mixing factor we can see an interesting phenomenon. In Fig 6 we present the modes whose mixing factor is higher than 0.1; that is, the modes that contribute most to the SWGMM. There is much less discrepancy in arrow length, therefore the speed in each direction looks homogeneous. In Fig 7 we can see the modes with mixing factor lower than 0.1. These two figures show that by using this simple filtering approach we can extract dominant motion patterns. These results also show that CLiFF-map is able to learn and represent rare events in an appropriate way.

B. Wind data

To further demonstrate the generality of the approach, we have also conducted experiments with wind data. Fig 8 shows a data set with sparsely sampled wind measurements from a mobile robot, which are insufficient to model wind flow properly. To tackle this problem we have employed the interpolation method presented in section II-3. We show the resulting map in Fig 9. It is visible that wind direction estimates for locations close to the edge are replicating the wind flow directions of the most external measurement locations. The closer the location to the centre the more turbulent the estimate is. Our interpolation method combines sampled measurements from adjacent locations. This results in creating a map of possible turbulent wind flow.

It is difficult to quantify the quality of interpolation. In case of the wind data set presented here the measurements were conducted in sequence and therefore we have no guarantee that the wind flow was stable over time. In such a case cross validation will result in removing an important chunk of data, that might directly affect the whole mapping process. One possible evaluation method would be to use a different data set, or to use a simulated wind data set.

IV. Summary and Future Work

In this paper we have presented our ongoing work on Circular Linear Flow Field maps (CLiFF-maps). CLiFF-mapis

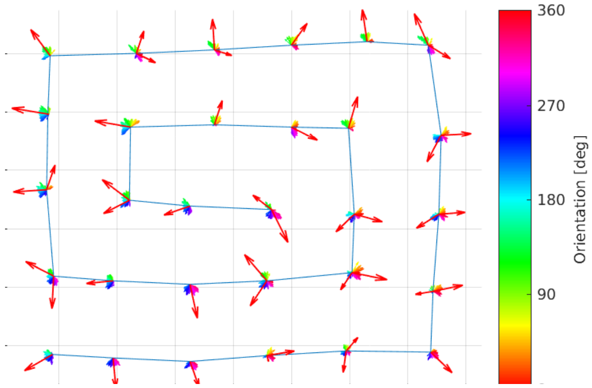

Fig. 8. Distributions obtained for a set of wind measurements using EM algorithm. The red arrows represent the directions of the modes while colour codded ones represent the raw measurements.

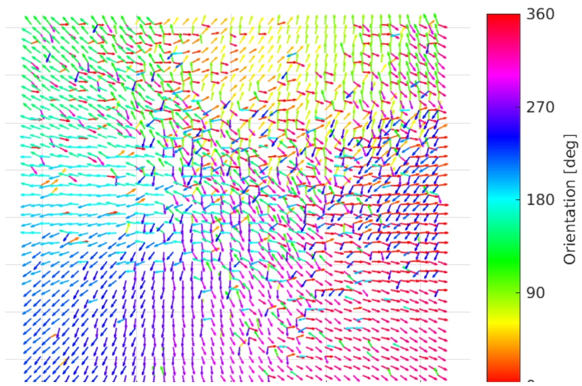

Fig. 9. A fully reconstructed wind map based on the set of measurements shown in figure 8.

a general method for mapping dynamics (as opposed to the structure of an environment), and locally models the underlying, unknown, velocity field with Semi-Wrapped Gaussian Mixture Models (SWGMM).

We have shown how to estimate the parameters of SWGMM PDFs using velocity observations within a predefined spatial window. This estimation is done in two steps: first, we generate a set of modes within the data set with Mean Shift; and second, this set is used to initialise Expectation Maximisation (EM). We show in detail how to apply these two methods to heterogeneous (circular-linear) data analogous to velocity.

We have also discussed the problem of sparse and missing data. We addressed this problem with interpolation based on data augmentation of the available observations. To estimate missing PDFs (from unobserved regions) we generate virtual observations by kernel-based sampling, and fit an SWGMM to the samples.

We have demonstrated the applicability of CLiFF-map for two significantly different data sets: people tracks and wind flow. In this way we have demonstrated that CLiFF-map is independent from the type of input data as long as it is possible to extract velocity samples from them.

In ongoing work we are developing methods to evaluate and verify the generated maps. The most promising direction is to rule out a number of estimated locations and reconstruct them using the remaining ones. The quality might be measured as a difference between the true distributions and the interpolated ones. We are also developing a method to include a confidence estimation for the interpolation method. Moreover we also work on methods for motion prediction.

Acknowledgements

This work was funded in part by the EU FP7 project SPENCER (ICT-2011-600877), the Swedish Knowledge Foundation project AIR (20140220) and H2020-ICT project SmokeBot (645101).

References

[1] D. Arbuckle, A. Howard, and M. Mataric. Temporal occupancy grids: a method for classifying the spatio-temporal properties of the environment. In *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, volume 1, pages 409–414. IEEE, 2002.

[2] Wolfram Burgard, Cyrill Stachniss, and Dirk Hähnel. Mobile robot map learning from range data in dynamic environments. *Adv. Robot.*, 35:3–28, 2007.

[3] Simone Calderara, Andrea Prati, and Rita Cucchiara. Mixtures of von Mises distributions for people trajectory shape analysis. *IEEE Trans. Circuits Syst. Video Technol.*, 21(4):457–471, apr 2011.

[4] Yizong Cheng. Mean Shift, Mode Seeking, and Clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(8): 790–799, 1995.

[5] A.P. AP Dempster, N.M. NM Laird, and DB Donald B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B Methodol.*, 39 (1):1–38, 1977.

[6] D. Hähnel, R. Triebel, W. Burgard, and S. Thrun. Map building with mobile robots in dynamic environments. In *2003 IEEE Int. Conf. Robot. Autom. (Cat. No.03CH37422)*, volume 2, pages 1557–1563, 2003.

[7] Thibault Kruse, Amit Kumar Pandey, Rachid Alami, and Alexandra Kirsch. Human-aware robot navigation: A survey. *Rob. Auton. Syst.*, 61(12):1726–1743, 2013.

[8] Tomasz Kucner, Jari Saarinen, Martin Magnusson, and Achim J. Lilienthal. Conditional transition maps: Learning motion patterns in dynamic environments. In *IEEE Int. Conf. Intell. Robot. Syst.*, pages 1196–1201, 2013.

[9] Matthias Luber, Gian Diego Tipaldi, and Kai Oliver Arras. Place-dependent people tracking. In *Int. J. Rob. Res.*, pages 557–572, 2009.

[10] Barbara Majecka. *Statistical models of pedestrian behaviour in the forum*. PhD thesis, University of Edinburgh, Edinburgh, 2009.

[11] Kantilal Varichand Mardia and Peter E. Jupp. *Directional Statistics*. Wiley, 2000.

[12] Nikos C. Mitsou and Costas S. Tzafestas. Temporal Occupancy Grid for mobile robot dynamic environment mapping. In *2007 Mediterr. Conf. Control Autom. MED*, pages 1–8. IEEE, 2007.

[13] Hans Moravec and A. Elfes. High resolution maps from wide angle sonar. In *IEEE Int. Conf. Robot. Autom.*, volume 2, pages 116–121, 1985.

[14] Tahir Nawaz, Andrea Cavallaro, and Bernhard Rinner. Trajectory clustering for motion pattern extraction in aerial videos. In *2014 IEEE Int. Conf. Image Process. ICIP 2014*, pages 1016–1020, 2014.

[15] Matteo Reggente and Achim J. Lilienthal. The 3D-Kernel DM+V/W algorithm: Using wind information in three dimensional gas distribution modelling with a mobile robot. In *2010 IEEE Sensors*, pages 999–1004. IEEE, nov 2010.

[16] Anandarup Roy, Swapan K. Parui, and Utpal Roy. SWGMM: a semi-wrapped Gaussian mixture model for clustering of circular???linear data, oct 2014.

[17] Jari Saarinen, Henrik Andreasson, and Achim J. Lilienthal. Independent Markov chain occupancy grid maps for representation of dynamic environment. In *2012 IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pages 3489–3495, Vilamoura, oct 2012. IEEE.

[18] Bernard W Silverman. *Density estimation for statistics and data analysis*, volume 26. CRC press, 1986.

[19] Sebastian Thrun. Robotic mapping: a survey. In Gerhard Lakemeyer and Bernhard Nebel, editors, *Explor. Artif. Intell. New Millenium*, pages 1–35. Morgan Kaufmann Publishers Inc., 2003.

[20] Yuta Wada, Marco Trincavelli, Yuichiro Fukazawa, and Hiroshi Ishida. Collecting a Database for Studying Gas Distribution Mapping and Gas Source Localization with Mobile Robots. In *Int. Conf. Adv. Mechatronics*, pages 183–188, 2010.

[21] Chieh-Chih Wang Chieh-Chih Wang and C. Thorpe. Simultaneous localization and mapping with detection and tracking of moving objects. In *Proc. 2002 IEEE Int. Conf. Robot. Autom.*, volume 3, pages 2918–2924, 2002.

[22] Zhan Wang, Rares Ambrus, Patric Jensfelt, and John Folkesson. Modeling motion patterns of dynamic objects by IOHMM. In *IEEE Int. Conf. Intell. Robot. Syst.*, pages 1832–1838, Chicago, sep 2014. IEEE.

[23] Denis F. Wolf and Gaurav S. Sukhatme. Mobile robot simultaneous localization and mapping in dynamic environments. *Auton. Robots*, 19(1):53–65, 2005.