Enabling Flow Awareness for Mobile Robots in Partially Observable Environments

Tomasz Piotr Kucner[1], Martin Magnusson[1], Erik Schaffernicht[1], Victor Hernandez Bennetts[1], and Achim J. Lilienthal[1]

***Abstract*— Understanding the environment is a key requirement for any autonomous robot operation. There is extensive research on mapping geometric structure and perceiving objects. However, the environment is also defined by the movement patterns in it. Information about human motion patterns can, e.g., lead to safer and socially more acceptable robot trajectories. Airflow pattern information allow to plan energy efficient paths for flying robots and improve gas distribution mapping. However, modelling the motion of objects (e.g., people) and flow of continuous media (e.g., air) is a challenging task. We present a probabilistic approach for general flow mapping, which can readily handle both of these examples. Moreover, we present and compare two data imputation methods allowing to build dense maps from sparsely distributed measurements. The methods are evaluated using two different data sets: one with pedestrian data and one with wind measurements. Our results show that it is possible to accurately represent multimodal, turbulent flow using a set of Gaussian Mixture Models, and also to reconstruct a dense representation based on sparsely distributed locations.**

I. Introduction

A. Motivation

Many robotics applications can benefit from maps going beyond mere occupancy by explicitly modelling the motion in an environment. To allow robots to operate efficiently and in an acceptable way among people, we have to provide methods to both map and exploit information about motion patterns. Motion pattern information can be useful for many tasks; e.g., task and motion planning, object tracking and intention recognition. These tasks are relevant for areas as diverse as Human Robot Interaction (HRI) and Mobile Robot Olfaction (MRO).

In HRI, information about how people usually move should affect the robot's motion planning. A motion planner can use this knowledge to build socially more acceptable and predictable paths [12]. Motion patterns implicit embed information about traffic rules and also mark frequented paths. Such information will help to improve motion planning in complex environments, as demonstrated in [16, 17], and allow robots to move in a safer way.

Similarly, in MRO and environmental monitoring, statistics about airflow can facilitate better understanding of gas distributions [20]. Gas plumes are heavily influenced by the local airflow. However, the task of wind flow modelling in small scales is rarely addressed [20]. Moreover, for tasks considering long term surveillance with aerial vehicles it is important to plan energy efficient paths considering air flow. Probabilistic modelling of wind flow can aid trajectory planning of a robot during exploration.

B. Problem statement

We address the problem of statistically modelling flows in non-trivial environments using sparse and noisy data.

It is rarely possible to have access to data completely describing the motion in an environment, e.g., building a very dense sensor network to monitor wind flow is usually unfeasible or too expensive. Another problem is randomness in motion observations, which can be a result of noisy measurements, or an inherent feature of the motions. Both cases require a proper statistical analysis in order to identify and model the motion patterns.

Our proposed Circular–Linear Flow Field map (CLiFF-map), introduced in Sec III-A, represents motion patterns using multimodal statistics to represent speed and orientation jointly. The approaches presented in Sec III-D allow to reconstruct a dense map from spatially and temporally sparse data.

C. Contribution

The contributions of the paper are as follows.

First, we describe CLiFF-map, a probabilistic framework for mapping velocity observations independently from their underlying physical processes. In this way we gain generality that allows to build a comprehensive and coherent model. The model consists of a set of Gaussian mixture models (GMMs) representing local observations, and variables describing the confidence in each part of the model, as well as the likelihood of the motion. In contrast to other methods, CLiFF-map represents speed and orientation in a joint space. This allows to properly address multimodality in the data.

Second, we investigate two data imputation methods for generating maps from sparsely distributed measurements.

Third, we present a validation of the representation and the quality of map reconstruction using real world data with the k-NN divergence estimator from [25].

II. Related Work

Our work is connected to the field of *mapping of dynamics*, which is different from *dynamic mapping* (creating a geometric map in a changing environment). Recently, a number of approaches for *mapping of dynamics* have been developed.

One general approach is to store past observations in a compressed way (e.g., work of Arbuckle et al. [1] and

This work was funded in part by the EU FP7 project SPENCER (ICT-2011-600877) and H2020-ICT projects SmokeBot (645101) and ILIAD (732737), the Swedish Knowledge Foundation projects AIR (20140220) and RAISE (20130196).

[1]The authors are with the MRO lab of the AASS research centre at Örebro University, Sweden. firstname.lastname@oru.se

Mitsou et al. [14]). Such methods provide tools to analyse the history of the environment, but they are not able to predict how it will evolve in the future. On the other hand, work on generative models, which try to predict future states of the environment, can be found in work of Wang et al. [26], Kucner et al. [10] and Saarinen et al. [23]. All of these methods treat dynamics as a change of occupancy in grid map cells and most of them assume that state changes of neighbouring cells are independent. In contrast, CLiFF-map models the flow field of the environment and how it affects the motion of objects.

The related problem of analysing *trajectories* of moving objects has been addressed, e.g., in work of Calderara et al. [3] and Nawaz et al. [15]. The authors try to detect similarities between trajectories and cluster them. An extension of this approach, presented by Ellis et al. [7], clusters trajectories based on their entry point and uses a Gaussian Process to model the observed motion patterns. The shared limitation of the trajectory analysing methods listed above is that they cannot work with inputs providing only velocity estimates or incomplete data. Moreover, they rely on the assumption of complete observability, which is difficult to ensure for large environments (e.g., airports). Furthermore, the work of Ellis et al. [7] requires to build a separate model for each entry point, which can lead to a complex multilevel representation. They also do not address the problem of multimodality, assuming that intersections will only happen between trajectories with different entry points.

Joseph et al. [9] describe trajectories as flow fields where each position has a corresponding path derivative. The motion in the environment is modelled as a mixture of Gaussian Processes. The major difference between CLiFF-map and the work of Joseph et al. is that CLiFF-map does not depend on the knowledge of trajectories. CLiFF-map aims to address a wide spectrum of problems focusing on cases where trajectory information is not available (air flow) or is scarce (data obtained with mobile platform), while the work of Joseph et al. focus on cases where it is possible to associate trajectories with objects (even if the information is sparse). Moreover the trajectory derivatives are modelled independently along the x and y axes while CLiFF-map treats velocity as joint random variable.

Chen et al. [4] propose a method to solve trajectory mapping as a constrained dictionary learning problem. However, there is no information about velocity and only heading is modelled.

One of the goals of CLiFF-map is to capture the dependency between the speed and the direction. The von Mises distribution, which is broadly used for modelling circular data (e.g., work of Calderara et al. [3], is not suitable for heterogeneous data where one of the components is circular while the other is linear. There have been attempts to overcome this obstacle by building Independent von Mises–Gaussian distribution, proposed by Roy et al. Roy et al. [21]. Such distributions assume that there is no correlation between orientation and magnitude of the velocity vector, which is an invalid assumption in many real world applications. Instead, CLiFF-map uses the idea of a Semi-Wrapped Gaussian Mixture Model presented in a later work of Roy et al. [22]. A similar idea on how to jointly analyse speed and orientation was suggested in work of Calderara et al. [2], which points out that using a normal distribution *on a line* and a wrapped normal distribution allows to compute a covariance matrix explaining the correlation between the speed and orientation.

In this paper we also address the data imputation problem [11]. Data imputation is a set of methods that are used to fill in missing records in a data set. We use non-parametric random kernel imputation [18, 19]. In this approach missing data are drawn, according to weights computed using a kernel function, from already computed models and then imputed into unobserved locations.

III. Algorithms

A. Representation

Velocity $\boldsymbol{V}$ can be described as a heterogeneous quantity combining orientation (θ) and speed (ρ), (see Eq. 1). In contrast to a representation using a 2D Euclidean vector (V_x, V_y), in this representation each component has a physical meaning and can be analysed separately, and the covariance matrix has a clear physical interpretation.

$$\boldsymbol{V} = (\theta, \rho)^T, \ \rho \in \mathbb{R}^+ \wedge \theta \in [0, 2\pi) \tag{1}$$

To build a probabilistic model of velocities we use a *semi-wrapped normal distribution* (SWND), which is a distribution on a cylinder. One of the dimensions is wrapped around the circumference of it and the other one is along its height.

$$\mathcal{N}^{\mathcal{SW}}_{\boldsymbol{\mu},\boldsymbol{\Sigma}}(\boldsymbol{V}) = \sum_{k\in\mathbb{Z}} \mathcal{N}_{\boldsymbol{\mu},\boldsymbol{\Sigma}}\left(\begin{bmatrix}\theta\\ \rho\end{bmatrix} + 2\pi\begin{bmatrix}k\\ 0\end{bmatrix}\right) \tag{2}$$

In Eq. 2 we see that density function of a semi-wrapped normal distribution is a periodic sum of period 2π of normal distributions on a surface defined by $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. We can imagine that we are wrapping a distribution on a cylinder and k, the winding number, indicates the number of revolutions.

To preserve the multimodal character of, e. g., wind or pedestrian flow we employ Semi-Wrapped Gaussian *mixture models* (SWGMM). A SWGMM is a PDF represented as a weighted sum of J SWNDs

$$p(\boldsymbol{V}|\boldsymbol{\xi}) = \sum_{j=1}^{J} \pi_j \mathcal{N}^{\mathcal{SW}}_{\boldsymbol{\Sigma}_j,\boldsymbol{\mu}_j}(\boldsymbol{V}), \tag{3}$$

where $\boldsymbol{\xi}$ denotes a finite set of components:

$$\boldsymbol{\xi} = \{\xi_i = (\boldsymbol{\Sigma}_i, \boldsymbol{\mu}_i, \pi_i) | i \in \mathbb{Z}^+\}. \tag{4}$$

Each component ξ_i is defined by its mean $\boldsymbol{\mu}_i$, covariance $\boldsymbol{\Sigma}_i$ and mixing factor π_i. The Circular–Linear Flow Field map (CLiFF-map) is a set of SWGMMs coupled with their location ($\boldsymbol{l}_j$), *motion probability* ($\bar{\mathsf{p}}_j$) and *observation ratio* ($\bar{\mathsf{q}}_j$), denoted as

$$\Xi = \{(\boldsymbol{\xi}_s, \bar{\mathsf{p}}_s, \bar{\mathsf{q}}_s, \boldsymbol{l}_s) | s \in \mathbb{Z}^+ \wedge \boldsymbol{l}_s \in \mathbb{R}^2\}. \tag{5}$$

The *motion probability* estimates how often motion occurs at $\boldsymbol{l}_j$, given our observations. We compute it as the ratio of the duration when motion has been observed and the total observation duration for $\boldsymbol{l}_j$:

$$\bar{\mathsf{p}}_j = \frac{T_m}{T_o}. \tag{6}$$

The *observation ratio* represents how much information we have collected at $\boldsymbol{l}_j$, relative to the rest of the map. It is the ratio between the observation duration for $\boldsymbol{l}_j$ and T_t, which is the duration of the whole experiment or the longest observation time for the data set:

$$\bar{\mathsf{q}}_j = \frac{T_o}{T_t}. \tag{7}$$

The *motion probability* carries information on how intensive the motion is in a given location. It is particularly useful for motion planning, marking busy areas. The *observation ratio* quantifies the relative confidence of a given distribution.

To estimate the parameters of a SWGMM ($\boldsymbol{\xi}$) we use a two step approach. First, we estimate the number of clusters and their positions using Mean Shift (MS). These clusters are initial conditions for Expectation Maximisation (EM).

1) Parameter estimation with EM: To estimate the parameters of the SWGMM, we use Expectation Maximisation (EM) [6]. For the general, n-dimensional case the derivation of update rules can be found in the work of Roy and Puri [22]. For the 2D case the update rules look as follows.

a) Expectation Step:

$$\eta^t_{ijk} = \frac{\pi^{t-1}_j \mathcal{N}\left(\vec{V}_i; \boldsymbol{\mu}^{t-1}_j + 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}, \boldsymbol{\Sigma}^{t-1}_j\right)}{\sum\limits_{j=1}^{M} \sum\limits_{k=-\infty}^{\infty} \pi^{t-1}_j \mathcal{N}\left(\vec{V}_i; \boldsymbol{\mu}^{t-1}_j + 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}, \boldsymbol{\Sigma}^{t-1}_j\right)} \tag{8}$$

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: Here we compute the new set of parameters Ξ using the following update rules.

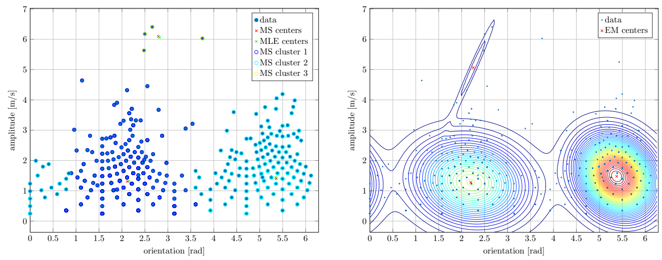
$$\pi^t_j = \frac{1}{N} \sum_{i=1}^{N} \sum_{k=-\infty}^{\infty} \eta^t_{ijk} \tag{9}$$

$$\boldsymbol{\mu}^t_j = \frac{\sum\limits_{i=1}^{N} \sum\limits_{k=-\infty}^{\infty} \left(\vec{V}_i - 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}\right) \eta^t_{ijk}}{\sum\limits_{i=1}^{N} \sum\limits_{k=-\infty}^{\infty} \eta^t_{ijk}} \tag{10}$$

$$\boldsymbol{\Sigma}^t_j = \frac{\sum\limits_{i=1}^{N} \sum\limits_{k=-\infty}^{\infty} \left(\vec{V}_i - \boldsymbol{\mu}_j - 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}\right) \left(\vec{V}_i - \boldsymbol{\mu}_j - 2\pi \begin{bmatrix} k \\ 0 \end{bmatrix}\right)^T \eta^t_{ijk}}{\sum\limits_{i=1}^{N} \sum\limits_{k=-\infty}^{\infty} \eta^t_{ijk}} \tag{11}$$

2) Mean Shift for EM initialisation: We employ Mean Shift [5] as a mode seeking algorithm to obtain the number and initial positions of modes and covariances for EM.

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 shifts the centre of the window to the computed mean. In this way we obtain maxims of



(a) Clusters obtained using MS algorithm for one of the locations.

(b) Distribution obtained using EM algorithm for one of the locations.

Fig. 1. Visualisation of mapping steps.

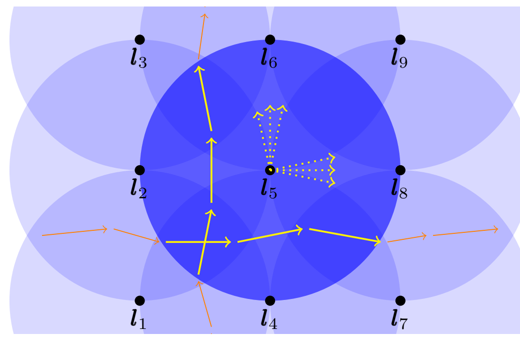


Fig. 2. Measurement discretisation procedure. Each blue circle represents an area with radius r from which measurements are associated with the location in its centre. Yellow arrows represent the measurements that are associated with location $\boldsymbol{l}_5$. The orange arrows represent measurements not taken into account for $\boldsymbol{l}_5$.

the underlying density function and clusters corresponding to each maximum. We define the window as an non-isotropic 2D Gaussian whose covariance matrix is defined as $\Sigma = \mathrm{diag}(\delta_\theta, \delta_\rho)$. Where δ_θ and δ_ρ are estimated with Silverman's rule [24] $\sigma = \left(4\hat{\sigma}^5/3N\right)^{\frac{1}{5}}$. Where $\hat{\sigma}$ is the standard deviation and N is the number of samples for the whole data set.

In Fig 1a we can see the initial set of clusters built with Mean Shift, while in Fig 1b we can see the resulting PDF after applying EM initialised with these clusters.

B. Data discretisation

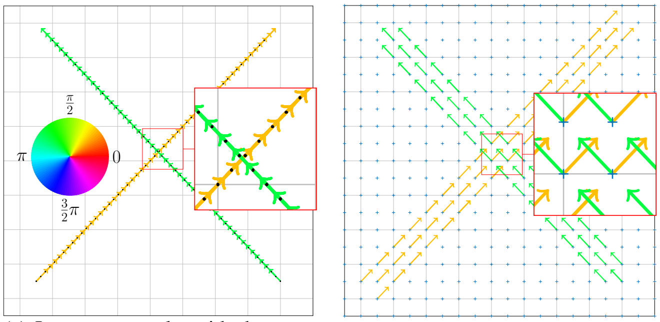
CLiFF-map builds a map of a flow field as if all the observations were obtained at a discrete set of measurement locations ($\boldsymbol{l_j}$). For wind mapping, such locations are places where the robot stopped to acquire data, while measures of people's velocities are associated to different locations. In order to compute a CLiFF-map, we aggregate measurements at a discrete set of locations, by assuming that each measurement within radius r of $\boldsymbol{l}_j$ was obtained at $\boldsymbol{l}_j$. See Fig. 2.

C. CLiFF-map analysis

CLiFF-map allows to grasp the multimodal characteristics of flows. Fig 4 demonstrates key characteristics of the CLiFF-map representation with two toy examples.

Fig. 3a shows two tracks. Black dots represent the measurements locations and arrows represent the measured velocities. To build the distributions shown in Fig. 3b we have discretised the data from Fig. 3a as described in Sec. III-B. We can see that CLiFF-map correctly models the two crossing flows.

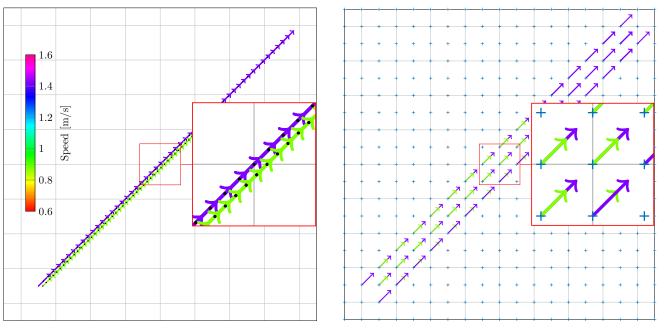
The tracks in Fig. 4a have different speeds.Fig. 4b shows that CLiFF-map correctly models the two modes coming



(a) Input: two tracks with the same velocity but different orientation.

(b) CLiFF-map for Fig. 3a, with modes for different directions.

Fig. 3. A toy example demonstrating how CLiFF-map addresses the problem of multiple modes.



(a) Input: two tracks with different velocity along the same line.

(b) CLiFF-map for Fig. 4a, with modes for different speeds.

Fig. 4. A toy example demonstrating how CLiFF-map addresses the problem of multiple modes.

from different speeds at the same location. In Sec IV we analyse CLiFF-map with real life data.

D. Data Imputation

As already mentioned, it is rare to have velocity data densely covering an area. We can define such problem as an incomplete data set. Therefore, our goal is to estimate missing data points[11], which will later on allow us to build a dense representation.

We compare two imputation methods employing a data model, in contrast to re-sample already obtained observations: *Monte Carlo Imputation* (MC) tends to preserve multimodal characteristics of the data and keep the sharp borders between different motion directions, while *Nadaraya Watson Imputation* (NW) smooths the data and introduces gradual changes between different data points.

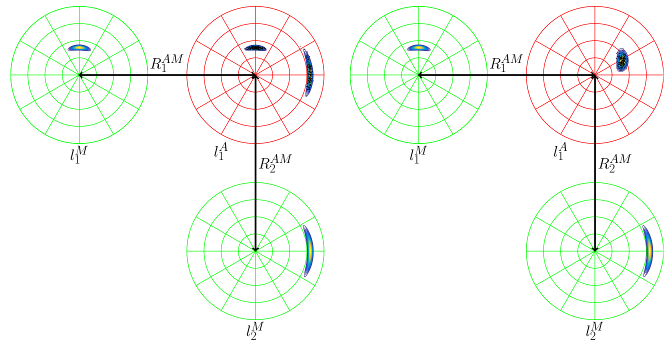
1) Monte Carlo Imputation: We reconstruct a distribution in an unobserved location by sampling *virtual observations* ($\hat{\boldsymbol{V}} = (\theta, \rho)$) from the surrounding, already learned, distributions as shown in Fig 5a. For obtaining *virtual observations*, we use hierarchical sampling (as follows).

First, we pick one of the existing SWGMMs ($\boldsymbol{\xi}$). The likelihood of picking an SWGMM is proportional to the distance (R_i^{AM}) between the location to be estimated ($\boldsymbol{l}_i^M$) and the existing one ($\boldsymbol{l}_1^A$):

$$R_i^{AM} = |\boldsymbol{l}_1^A - \boldsymbol{l}_i^M| \tag{12}$$

$$p(\boldsymbol{\xi} = \hat{\boldsymbol{\xi}}_{\boldsymbol{i}} | \bar{\mathsf{q}}_i, K(R_i^{AM})) = \bar{\mathsf{q}}_i K(R_i^{AM}) \tag{13}$$

In Eq 13, a kernel function $K(\bullet)$ defines the sampling window. In this work we use a Gaussian kernel $K(R_i^{AM}) = \mathcal{N}(R_i^{AM}|0, \sigma)$.



(a) MC imputation.

(b) NW imputation.

Fig. 5. Comparison of imputation methods. Green plots describe the SWGMM for *observed* locations, and red is *unobserved*. Black dots represent virtual observations sampled from locations $l_{(2,1)}^M$ and $l_{(1,2)}^M$. The number of observations is proportional to the distance R_1 and R_2.

After, we sample a virtual observation from this SWGMM. Depending on the *motion probability* $\bar{\mathsf{p}}$ we can sample an *empty observation* (no motion) or a realisation of $\boldsymbol{V}$.

Finally, we obtain the realisation of $\boldsymbol{V}$. The probability of realisation $\hat{\boldsymbol{V}}$ is presented in Eq 14.

$$p(\boldsymbol{V} = \hat{\boldsymbol{V}} | \hat{\boldsymbol{\xi}}) = \sum_{i=1}^{J} \mathcal{N}^{SW}(\hat{\boldsymbol{V}} | \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\pi_i \tag{14}$$

To each of the newly estimated SWGMMs we associate a *trust factor*, which denotes how confident we are in the reconstruction. The intuition behind this factor is that we have more trust in reconstructed locations which are closer to observed ones.

$$\hat{\mathsf{t}} = \frac{M}{\sum_{i=1}^{N}(R_i^{AM} + 1)}, \tag{15}$$

where M is the number of realisations of $\boldsymbol{V}$ and R_i^{AM} is the distance between $\boldsymbol{l}_1^A$ and the SWGMM selected in step 1.

2) Nadaraya Watson Imputation: The first step is to sample one observation from each existing SWGMM. We discard observations with *zero* velocity.

The realisation $\hat{\boldsymbol{V}}$ of variable $\boldsymbol{V}$ was presented in Eq 14. In the following step we compute the weighted mean of observations. The weights are proportional to the distance and the *observation ratio* (see Eq 16).

$$\bar{\boldsymbol{V}} = \frac{\sum_{i=1}^{N} K(R_i^{AM} \bar{\mathsf{q}}_{\mathsf{i}}^{-1}) \hat{\boldsymbol{V}}_i}{\sum_{i=1}^{N} K(R_i^{AM} \bar{\mathsf{q}}_{\mathsf{i}}^{-1})} \tag{16}$$

For this imputation method we compute the *trust factor* as the number of observations having a high impact on $\boldsymbol{l}_1^A$, divided by the distance from $\boldsymbol{l}_1^A$,

$$\hat{\mathsf{t}} = \frac{\sum_{i=1}^{N} I(K(R_i^{AM}))}{\sum_{i=1}^{N} I(K(R_i^{AM}))R_i^{AM} + I(K(R_i^{AM}))}, \tag{17}$$

where $I(\bullet)$ is an indicator function

$$I(d) = \begin{cases} 1 & \text{if } d < 3\sigma \\ 0 & \text{if } d \geq 3\sigma \end{cases} \tag{18}$$

IV. Results

As mentioned in Sec I-C the evaluation will be twofold. First, we evaluate the divergence between the observations and the CLiFF-map. Second, we present an evaluation of the map reconstruction (using imputation).

A. Pedestrian Data

We assume that to each location in the map there is associated a true and unknown PDF $\boldsymbol{\xi}^t$. It is impossible to access this distribution directly. We can only access a set of observations $\{\hat{\boldsymbol{V}}_1^o, \ldots, \hat{\boldsymbol{V}}_n^o\}$. One method to estimate the divergence between two d-dimensional distributions, while having access only to samples was proposed by Wang et al. [25]. They propose an estimator that employs only the samples coming from the two distributions. The estimator is computed as follows:

$$\hat{D}_{n,m}(\boldsymbol{\xi}^t || \boldsymbol{\xi}) = \frac{d}{n} \sum_{i=1}^{n} \log_2 \frac{\nu_k(i)}{\rho_k(i)} + \log_2 \frac{m}{n-1} \tag{19}$$

The idea is to compare the distance $\rho_k(i)$ between $\hat{\boldsymbol{V}}_i^o$ and its k-NN in $\{\hat{\boldsymbol{V}}_j^o\}_{j \neq i}$ to the distance $\nu_k(i)$ between $\hat{\boldsymbol{V}}_i^o$ and its k-NN in $\{\hat{\boldsymbol{V}}_j^s\}$, where $\{\hat{\boldsymbol{V}}_j^s\}$ denotes virtual observations sampled from the CLiFF-map.

Using this divergence estimator to evaluate the model is beneficial because it allows to estimate the distance between the true distribution and the reconstructed one, even though we do not have direct access to a true distribution.

The pedestrian data set [13] does not contain velocities. We approximate them assuming that motion between subsequent detections is linear and constant, thus obtaining a dense set of measurements as shown in Fig. 10a, where we can see approx. 250 000 velocity estimates obtained during 12 h of observation. In the figure we can observe a diagonal motion pattern connecting the top left and bottom right corners. Moreover, there is a vertical motion pattern on the right and on the left side of the image. These patterns are also visible in the reconstructed map in Fig. 10b. Fig. 10c shows a map of *motion probability*. Brighter colours correspond to cells with higher motion intensity.

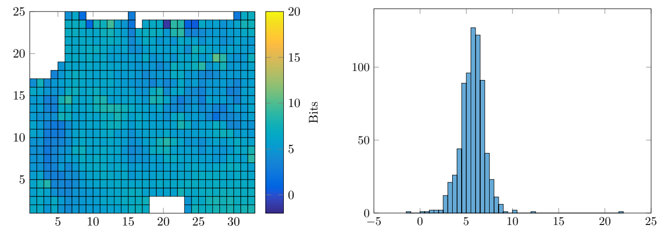
Fig. 11a shows a CLiFF-map obtained by discarding 75% of the locations shown in Fig 10b. The dominant motion patterns are still clearly visible.

Fig. 11b shows a map reconstructed using MC with kernel size 0.25 m. The dominant diagonal direction is preserved, and the big motionless area on the left side of the map was reconstructed correctly. Also the vertical and horizontal motion patterns along the edges of the map remain visible.

Fig. 11c shows a reconstruction of the map with NW, also with kernel size 0.25 m. The quality of the resulting map is much lower. There is a clear line dividing the map in to two areas, and the motion patterns are averaged over a big area.

In fig. 12a a more challenging example is presented. Only 3% of the locations remained and were used for map reconstruction. The result of reconstruction with MC (see Fig. 12b) still appears to be qualitatively correct. The motion patterns on the left and in the upper right corner are still visible, but the information about the motion in the lower left corner is lost. In Fig. 12c, the results of reconstruction with NW looks similar to the previous one. We can see a clear cut into two areas and over-smoothing.

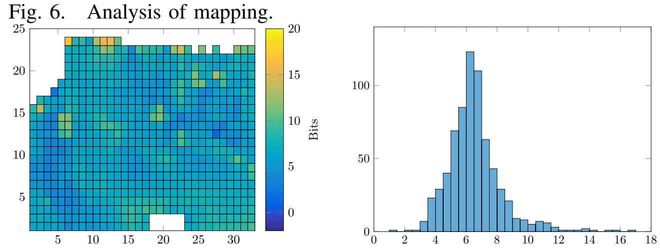
To quantitatively measure the quality of the reconstruction, we compute the divergence between the learned SWGMMs and the original data using the estimator of Wang et al [25].



(a) A plot of the distance between the measurements and the map 10b

(b) Histogram of divergence values based on 6a.

Fig. 6. Analysis of mapping.



(a) A plot of the distance between the velocity measurements and the result of interpolation.

(b) Histogram of divergence values based on 7a.

Fig. 7. Analysis of reconstruction with Monte Carlo imputation.

Fig 6 shows a map of divergence and the histogram of the computed distances. The distances are concentrated around 5 bits. Fig 7 shows results of map reconstruction based on MC. The distances are concentrated as in Fig 6b, however the average bit count is higher. There is a similar tendency for maps reconstructed with NW imputation. In Fig. 8, the histogram and divergence map look similar to the previous examples.

To properly evaluate the quality of reconstruction, we have measured the divergence for reconstructed maps for 9 days, each consisting of 825 cells, i. e., 7425 data points. It took on average 8 s to estimate parameters for each location.

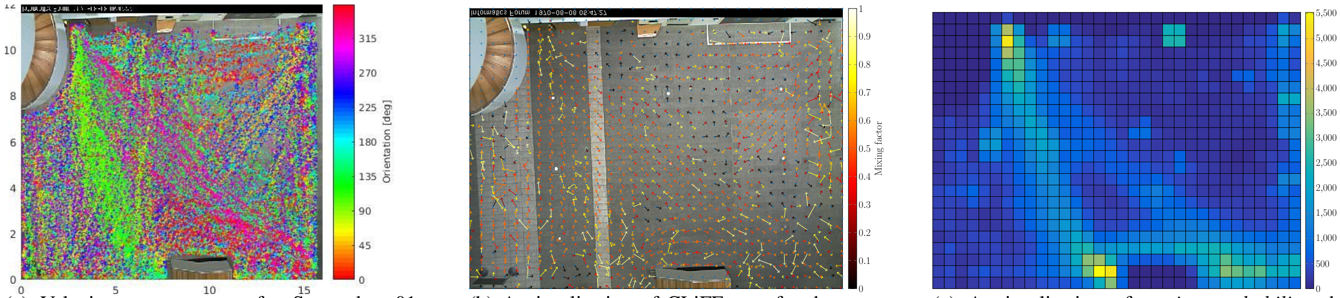
In Fig. 9a and 9b we present how the reconstruction quality depends on the imputation method and the kernel size. For this experimental setup the reconstruction quality depends on the amount of input data. However, in both cases the results are close to the quality of the original map. The MC method shows to be better over all than NW method, except one case in 3% data set, in which, for a small kernel, NW is better.

However, while the quality of reconstruction with MC improves with larger kernel sizes, the quality with NW decreases.

As a baseline comparison to the CLiFF-map representation and the MC and NW reconstruction methods, we have implemented a histogram method. Each location stores a histogram of velocity observations, with bin sizes 0.12 rad and depending on the day from 1-2.25 m/s.[1] Results are shown in the "Dense" column of Tab I. In each case, CLiFF-map has about 0.5 bits less divergence than the histogram maps.

As a baseline for map reconstruction, we have linearly interpolated the 4 nearest histograms to recreate the missing locations. In this case, Tab I shows that the reconstruction quality depends on the density of the input data. For 1 m

[1]Histograms were computed using MATLAB®'s histcounts2 function.

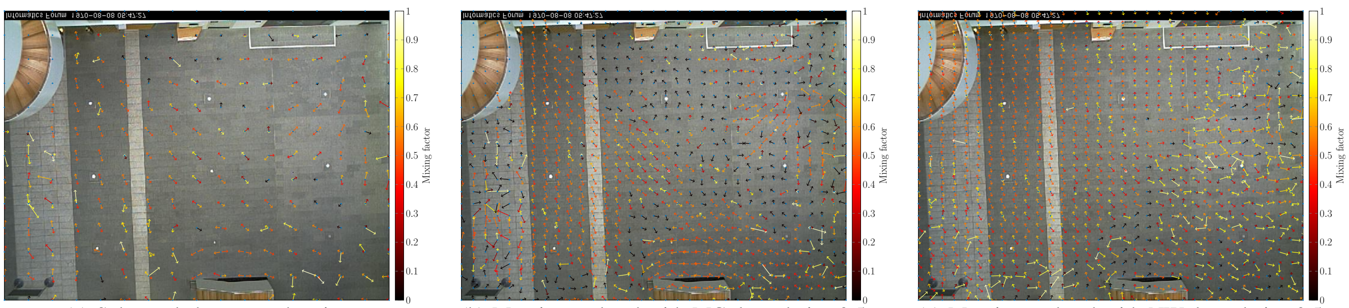


(a) Velocity measurements for September 01, 2009. Arrows are coloured based on their orientation, and the lengths are proportional to the speed.

(b) A visualisation of CLiFF-map for the same data. The distance between locations is 0.5 m and the discretisation radius is also 0.5 m. (We show only means of modes with mixing factor higher than 0.05).

(c) A visualisation of *motion probability*, the colours correspond to how many times motion was observed in each location.

Fig. 10. CLiFF-map constructed from all available observations. We can see that the patterns from (a) are preserved in (b).

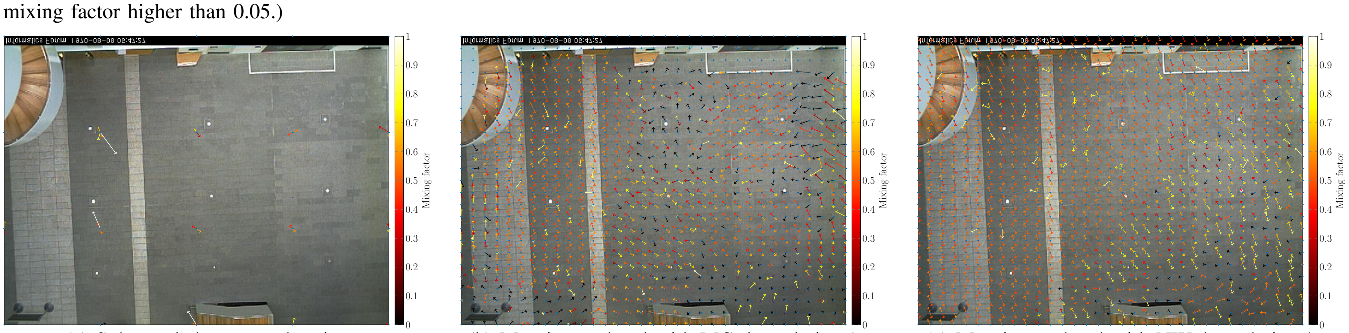


(a) Subsampled map used as input.

(b) Map interpolated with MC, kernel size 0.5 m.

(c) Map interpolated with NW, kernel size 0.5 m.

Fig. 11. Sparse reconstructions from 221 locations (1 m between input locations), shown as blue crosses in (a). (We show only means of modes with mixing factor higher than 0.05.)

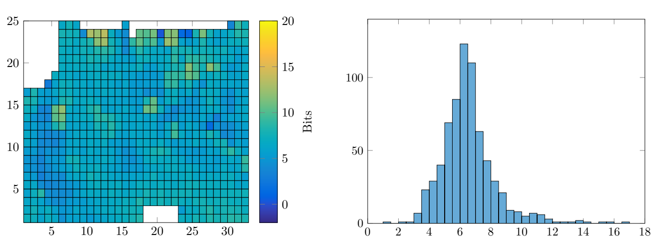


(a) Subsampled map used as input.

(b) Map interpolated with MC, kernel size 1 m.

(c) Map interpolated with NW, kernel size 1 m.

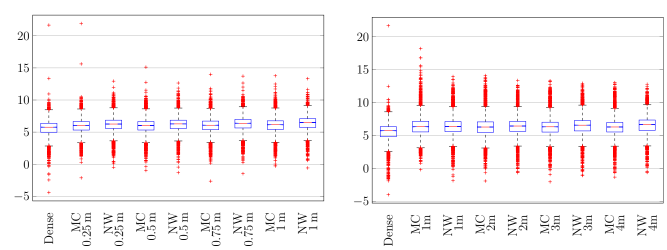
Fig. 12. Sparse reconstructions from 20 locations (4 m between input locations), shown as blue crosses in (a). (We show only means of modes with mixing factor higher than 0.05.)



(a) A plot of the distance between the velocity measurements and the result of interpolation.

(b) Histogram of divergence values based on 8a.

Fig. 8. Analysis of reconstruction with Nadaraya Watson imputation.



(a) 75% of initial data

(b) 3% of initial data.

Fig. 9. The box-plot of divergence for reconstructed maps

resolution, both MC and NW outperforms histogram interpolation. For 4 m, resolution, the quality of MC and NW reconstruction is similar to interpolation. We assume, that this drop in quality is caused because there is too little data for proper reconstruction and is independent from the reconstruction method.

B. Wind Data

The wind data set [8] does not provide ground truth, because there are no denser measurements to compare to. This data set will instead be evaluated qualitatively using *stability maps*. A stability map shows which regions contain a stable wind flow over multiple sessions.

To evaluate stability we compute a summed pairwise

date	Dense (0.5m)		Hist. Interp.		CLiFF+MC		CLiFF+NW	
	CLiFF	Hist.	1m	4m	1m	4m	1m	4m
01.09	5.424	5.824	6.191	6.217	6.079	6.531	6.149	6.470
02.09	5.198	5.570	6.150	6.243	5.777	6.175	5.993	6.123
04.09	5.559	5.774	6.134	6.144	5.952	6.397	6.225	6.372
05.09	4.320	5.756	6.295	6.294	5.996	6.396	6.229	6.408
06.09	4.068	5.717	6.356	6.505	5.961	6.289	6.213	6.377
10.09	4.928	5.589	5.869	5.972	5.529	6.025	5.725	6.230
11.09	4.551	4.952	5.509	5.568	5.161	5.664	5.410	5.755
12.09	4.548	4.886	6.511	6.590	4.345	5.633	5.009	5.781
13.09	4.297	4.957	5.398	5.504	5.031	5.671	5.543	5.776
Mean	4.872	5.447	6.046	6.115	5.527	6.087	5.833	6.144

TABLE I. Comparison of divergence for CLiFF-map and baseline histogram maps, showing divergence [25] from original data (in bits) for maps with 0.5 m discretisation ("Dense" column) and reconstructions from sparsely sampled maps, using several days form the pedestrian data set.

symmetric KLD between all corresponding map cells.

$$I(k) = \sum_{i,j \in n} s\text{KLD}(k)_{i,j} \tag{20}$$

The KLD is computed in a discretised state space $\Gamma_{\Theta,P}$. Each state in $\Gamma_{\Theta,P}$ is a $\gamma = (\Theta, P)$ tuple of direction and speed where $\Theta \in \{\Theta_{min}, \Theta_{min} + \Delta\Theta, \ldots, \Theta_{max}\}$ and $P \in \{P_{min}, P_{min}+\Delta P, \ldots, P_{max}\}$. We denote a discretised PDF for each distribution for each location ($\boldsymbol{\xi}$) as $\boldsymbol{x} = (x_\gamma | \gamma \in \Gamma)$. Therefore we can define KLD for kth element in the map as

$$\text{KLD}(\boldsymbol{x}^a(k) || \boldsymbol{x}^b(k)) = \sum_{\gamma \in \Gamma} x^a_\gamma(k) \log \frac{x^a_\gamma(k)}{x^b_\gamma(k)}. \tag{21}$$

The symmetric KLD (sKLD) for each cell is computed as

$$\text{sKLD}(k) = \text{KLD}(\boldsymbol{x}^r(k) || \boldsymbol{x}^{gt}(k)) + \text{KLD}(\boldsymbol{x}^{gt}(k) || \boldsymbol{x}^r(k)) \tag{22}$$

The wind data was collected in a foundry, using a mobile platform. The size of the environment is 40 m x 70 m. The robot was deployed for several data collection tours. In each tour, the robot stopped at different way points and collected data for 120 s. Using data from these locations we have reconstructed the airflow in the environment with MC and NW imputation, respectively.

Fig. 13a shows a map reconstructed with MC. The colour corresponds to the mixing factor. It is possible to clearly distinguish areas with different wind directions. Moreover there are clear borders between areas with different airflow directions. Fig. 13b shows a map representing the confidence in the reconstruction. The highest confidence is near the measurement locations and clearly decreases with the distance from the measurement location.

Fig. 14a shows the result of reconstruction with NW. The areas with similar motion directions are as visible as in Fig. 13a, however in this case it is important to mention that the borders separating the regions are not so visible. We can observe a gradual, smooth change between the flows.

Fig. 14b shows the trust map for NW. The positions of measurement locations are still visible, however the overall score is much lower and more evenly distributed.

Figs. 13c and 14c show stability maps built from data obtained during 7 different sessions. Even though the maps were obtained using different interpolation methods, large parts of them present the same level of stability. However, what is most interesting is the areas with different stability levels. We can see that in Fig. 13c the centre of the map is stable while in Fig. 14c the same area has a worse score. It is caused by the differences between the imputation methods. MC keeps all virtual samples, therefore it is able to model turbulent wind behaviour. NW, on the other hand, averages the virtual observations finding the dominant wind direction. This results in different types of distributions in this area. MC builds a set of models of turbulent flow which are more similar, while NW builds a set of models which have a clear dominant direction. In the latter case such smoothed out distributions can vary a lot between the different data sets and are the reason for the worse stability score. It depends on the application, which behaviour is more preferable.

V. Summary and Future Work

We have presented CLiFF-map, an approach for flow mapping. CLiFF-map is a complete probabilistic model, which also accounts for motion probability and model confidence. We have also introduced two methods for map reconstruction from spatially sparse measurements (MC and NW). Finally we have evaluated the accuracy of the representation and of the reconstruction methods.

The evaluation was threefold. First, based on dense pedestrian data we have built a CLiFF-map model and computed the divergence [25] between the original data and the model. The results have also shown that CLiFF-map accurately represents multimodal flow of people.

In the second experiment with pedestrian data we have reconstructed maps using 75% and 3% of the original data. We have evaluated the quality of reconstruction using the same divergence estimator. The results show that MC reconstruction performs better than NW. The quality of reconstruction with MC is less sensitive to the kernel size than NW.

The third experiment, with wind data, shows an interesting phenomenon. We have built stability maps showing changes in the flow across several days – one using MC and another with NW. The stability maps show large similarities, which implies that the PDFs reconstructed with MC and NW are similar, even though the methods emphasise different features of the flow.

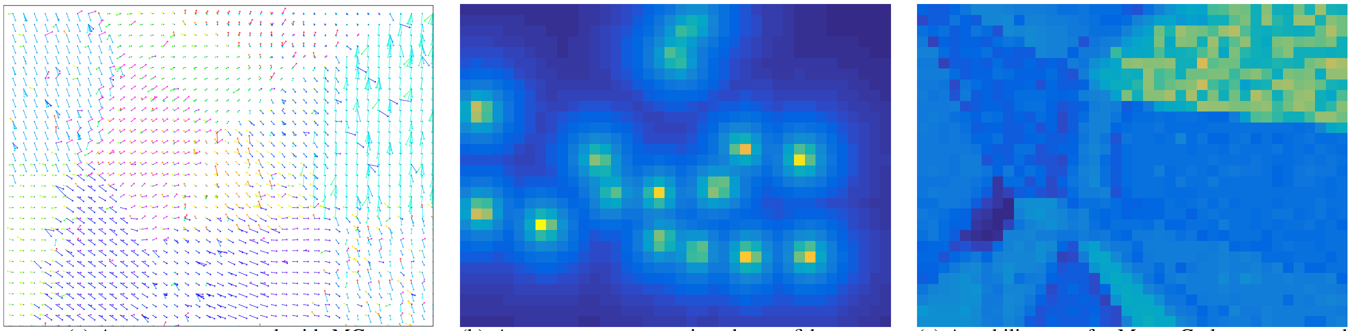
The conducted experiments show, that CLiFF-map can accurately model multimodal, turbulent flow of people and air. The results also support the hypothesis that it is possible to reconstruct a map of flow based on sparse measurements.

Knowledge of the flow in the environment influences motion planning, allowing to build more efficient trajectories.

A future direction of research is to address flow changes in time. A time dependent model will allow to grasp flow variability over time. It will also be important to include information about the environment's shape. Additional spatial information will improve the reconstruction procedure.

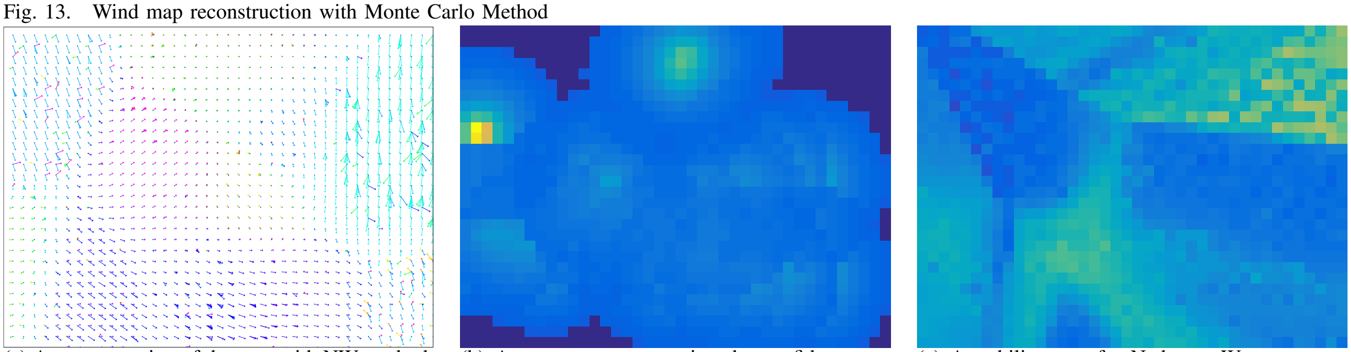
References

[1] D. Arbuckle, A. Howard, and M. Mataric. Temporal occupancy grids: a method for classifying the spatio-temporal properties of the environment. In *IROS*, volume 1, pages 409–414, 2002.



(a) A map reconstructed with MC. (b) A trust map representing the confidence per location in the map for MC reconstruction. (c) A stability map for Monte Carlo reconstructed map.

Fig. 13. Wind map reconstruction with Monte Carlo Method



(a) A reconstruction of the map with NW method. (b) A trust map representing the confidence per location in the map for NW reconstruction. (c) A stability map for Nadaraya Watson reconstructed map.

Fig. 14. Wind map reconstruction with Nadaraya Watson Method

[2] S. Calderara, A. Prati, and R. Cucchiara. Learning People Trajectories Using Semi-directional Statistics. In *AVSS*, pages 213–218, sep 2009.

[3] S. Calderara, A. Prati, and R. Cucchiara. Mixtures of von Mises distributions for people trajectory shape analysis. *TCSVT*, 21(4):457–471, Apr. 2011.

[4] Y. Chen, M. Liu, , and J. P. How. Augmented Dictionary Learning for Motion Prediction. In *ICRA*, 2016.

[5] Y. Cheng. Mean Shift, Mode Seeking, and Clustering. *TPAMI*, 17(8):790–799, 1995.

[6] A. A. Dempster, N. N. Laird, and D. D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J-R-STAT-SOC-SER-A-GENERAL*, 39(1):1–38, 1977.

[7] D. Ellis, E. Sommerlade, and I. Reid. Modelling pedestrian trajectory patterns with gaussian processes. In *ICCV Workshops*, pages 1229–1234, 2009.

[8] V. Hernandez, E. Schaffernicht, A. J. Lilienthal, H. Fan, T. P. Kucner, L. Andersson, and A. Johansson. Towards occupational health improvement in foundries through dense dust and pollution monitoring using a complementary approach with mobile and stationary sensing nodes. In *IROS*, Daejeon, Korea, 2016.

[9] J. Joseph, F. Doshi-Velez, A. S. Huang, and N. Roy. A Bayesian nonparametric approach to modeling motion patterns. *AURO*, 31(4):383–400, 2011.

[10] T. Kucner, J. Saarinen, M. Magnusson, and A. J. Lilienthal. Conditional transition maps: Learning motion patterns in dynamic environments. In *IROS*, pages 1196–1201, 2013.

[11] R. J. Little and D. B. Rubin. *Statistical analysis with missing data*. John Wiley & Sons, 2014.

[12] M. Luber, L. Spinello, J. Silva, and K. O. Arras. Socially-aware robot navigation: A learning approach. In *IROS*, pages 902–907, 2012.

[13] B. Majecka. Statistical models of pedestrian behaviour in the forum. Master's thesis, University of Edinburgh, Edinburgh, 2009.

[14] N. C. Mitsou and C. S. Tzafestas. Temporal Occupancy Grid for mobile robot dynamic environment mapping. In *MED*, pages 1–8, 2007.

[15] T. Nawaz, A. Cavallaro, and B. Rinner. Trajectory clustering for motion pattern extraction in aerial videos. In *ICIP*, pages 1016–1020, 2014.

[16] S. T. O'Callaghan, S. P. N. Singh, A. Alempijevic, and F. T. Ramos. Learning navigational maps by observing human motion patterns. *ICRA*, pages 4333–4340, 2011.

[17] L. Palmieri, T. P. Kucner, M. Magnusson, A. J. Lilienthal, and K. O. Arras. Kinodynamic motion planning on Gaussian mixture fields. In *ICRA*, 2017.

[18] N. Pettersson. Multiple kernel imputation: A locally balanced real donor method. 2013.

[19] Y. Qin, S. Zhang, X. Zhu, J. Zhang, and C. Zhang. {POP} algorithm: Kernel-based imputation to treat missing values in knowledge discovery from databases. *Expert Systems with Applications*, 36(2, Part 2):2794 – 2804, 2009.

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

[21] A. Roy, S. K. Parui, and U. Roy. A Mixture Model of Circular-Linear Distributions for Color Image Segmentation. *IJCA*, 58 (9):6–11, 2012.

[22] A. Roy, S. K. Parui, and U. Roy. SWGMM: a semi-wrapped Gaussian mixture model for clustering of circular-linear data, oct 2014.

[23] J. Saarinen, H. Andreasson, and A. J. Lilienthal. Independent Markov chain occupancy grid maps for representation of dynamic environment. In *IROS*, pages 3489–3495, Vilamoura, oct 2012.

[24] P. Taylor and K. Dehnad. *Density Estimation for Statistics and Data Analysis*, volume 26. CRC press, 2012.

[25] Q. Wang, S. R. Kulkarni, and S. Verdú. Divergence estimation for multidimensional densities via k-nearest-neighbor distances. *EEE Trans. Inf. Theory*, 55(5):2392–2405, 2009.

[26] Z. Wang, R. Ambrus, P. Jensfelt, and J. Folkesson. Modeling motion patterns of dynamic objects by IOHMM. In *IROS*, pages 1832–1838, Chicago, sep 2014.