# Extracting Statistically Significant Behaviour from Fish Tracking Data With and Without Large Dataset Cleaning

# Extracting Statistically Significant Behaviour from Fish Tracking Data With and Without Large Dataset Cleaning

Cigdem Beyan[1,*], Vasiliki-Maria Katsageorgiou[1], Robert B. Fisher[2]

[1]Pattern Analysis and Computer Vision Department (PAVIS), Istituto Italiano di Tecnologia (IIT), Genoa, Italy
[2]School of Informatics, University of Edinburgh, Edinburgh, UK
[*]cigdem.beyan@iit.it

**Abstract:** Extracting a statistically significant result from video data of natural phenomenon can be difficult for two reasons: *i)* there can be considerable natural variation in the observed behaviour, and *ii)* computer vision algorithms applied to natural phenomena may not perform correctly on a significant number of samples. This paper presents one approach to cleaning of a large noisy visual tracking dataset to allow extracting statistically sound results from the image data. In particular, the paper presents an analysis of a dataset of 3.6 million underwater trajectories of a species of fish, which are also labelled with the water temperature at the time of acquisition. Although there are many false detections and incorrect trajectory assignments, by a combination of data binning and robust estimation methods, we demonstrate reliable evidence for an increase in fish speed as water temperature increases. We also present a method for data cleaning which removes outliers arising from false detections and incorrect trajectory assignments using an effective deep learning based clustering algorithm. The corresponding results show a rise in fish speed as temperature goes up. Several statistical tests applied to both cleaned and not-cleaned data confirm that both results are statistically significant and show an increasing trend (not random). However, the latter approach also generates a cleaner dataset suitable for other analysis.

## 1. Introduction

There has been increasing interest in the use of computer vision methods for the analysis of natural world phenomenon, such as for species abundance and variety inventory, farm animal behaviour monitoring, and many specific scientific investigations. Examples of this increased interest are seen at special workshop series (*e.g.,* [1, 2]), the LifeCLEF competitions (*e.g.,* [3]), and in special books [4, 5].

One advantage of computer vision methods is the ability to acquire and analyse large amounts of data automatically, which can lead to more statistically sound results based on larger datasets. The downside of the computer vision approach is also linked to the large datasets acquired: it may not be possible to ensure that all of the data comes from the phenomenon of interest or is correctly measured, even with the use of mass ground-truthing capabilities such as facilities like Mechanical Turk.

Typical sources of error include:

- Undetected transient sensor failures, such as compression or communication artefacts.
- Target detection failures, such as missed detections, detection of overlapping individuals as a single target, false detections due to moving background items, illumination variations, unrelated or unexpected foreground objects, etc.
- Target tracking failures, such as mis-assignment errors arising when multiple individuals are present, detection failures occur, or from occlusions between multiple targets.
- Target mis-identification errors, such as the confusion of one individual or species for another, a common occurrence given the unconstrained target pose and environmental conditions, *e.g.,* dust and lighting.
- Measurement errors, such as sizes or positions, which can arise from special cases not considered by the algorithms, *e.g.,* partial occlusions or partial detection failures or unusual poses.

It is possible to develop post-processing filters to detect and remove some of these failures [6], and manual review of the data is certainly possible. But, as we move to the era of 'big data', it is no longer feasible to ensure that the datasets are 100% clean. Hence, computer vision needs to develop methods that are more robust to a wide variety of sources of data errors, not just sensor error (which are generally handled well by current robust methods).

Herein, a dataset consisting of about 4 million fish trajectories is analysed to explore how fish speed varied with respect to water temperature. Investigating the behaviour of coral reef fish species at different temperatures can help to assess their sensitivity to climate change [7]. There are marine biology studies (such as [8, 9]) which explored the fish activity in different water temperatures. In their analysis, a fish tank model, which allows to modify the temperature of the water [8], was used in contrast to our study which explores the data obtained from underwater videos in a natural setting that reflects the seasonal changes in water temperature. Some of our results have been reported previously in [7] from an ecological perspective where interested readers can find a deeper discussion regarding our findings and the findings of other studies. In [7], it was shown that one can still obtain statistically sound results, even with a dataset that contains examples of all the errors listed above. It shows that by a combination of data binning and robust statistics over a large amount of data, statistically sound inferences can be made from the very noisy data (*i.e.,* the trend that the fish speed increases with water temperature). Different from [7], this paper focuses on the computer vision and big qualitative data analysis methods. The main contribution of this paper is a method for cleaning noisy tracking data and estimation of a dataset property (speed) robustly with and without cleaning the dataset. The found trend using all data was also validated by removing outliers (i.e. by cleaning the data) which could represent errors especially false detections and incorrect trajectory assignments. To detect outliers, we propose an effective outlier detection algorithm which is based on cluster cardinality. Clusters are obtained applying a mean-covariance Restricted Boltzmann Machine (mcRBM) which groups the data such

that data points in the same group are more similar to each other than to those in other clusters.

The rest of this paper is organized as follows. Section 2 discusses some current considerations about big data, in particularly data cleaning and outlier detection methods that have been used for data cleaning. The dataset used in this work is introduced in Section 3 including the error estimation that was performed using random subset selection. In Section 4, the outlier detection method which is based on mean-covariance Restricted Boltzmann Machine (mcRBM), the data binning method which is used to analyse the whole data and the cleaned data are presented. We present the experimental results in Section 5. Finally, we conclude the paper with a discussion in Section 8.

## 2. Background

Thanks to improvements in computational and storage resources and data acquisition tools, big data analysis has become a central topic in data science research. Similarly, for investigation of animal biometrics, recognizing different species and animal behaviour understanding, scientists have started to undertake their analysis using big data which contains not only huge amounts of data but also the data which includes variety such as different species or behavior classes. Among several works, Palazzo and Murabito [10] proposed a semi-supervised fine-grained fish recognition algorithm which utilized a dataset having 20 million fish images. In [11], convolutional neural networks were used to classify the two different behaviors of *Drosophila*. In that study [11] *i)* standing/walking and *ii)* not in physical contact with the substrate were recognized using over 21600 labeled training data. To analyse the social behaviour of mice, in [12], a single target tracking method was presented. Once the mice were tracked continuously for five days, the corresponding trajectories were analysed to measure the social behavior of mice. In total, 100k samples were collected and used for the evaluation of the proposed system. Interested readers can refer to a recent survey on visual animal biometrics [13] which also mentioned the importance of developing better platforms to be able to present more efficient algorithms to process massive data.

Although it is not applied to big data, the research most related to this paper is [14] which presented a method for automatic detection of the erroneous fish trajectories (which exist due to object occlusions, tracker mis-associations and background movements). In that work [14], trajectories were represented using Hidden Markov Models (HMMs). Later, Multi-Dimensional Scaling (MDS) was applied to project all trajectories onto a low-dimensional fixed length vector space. K-means clustering was then applied to the resulting vectors to model the correct trajectories which were used to detect erroneous trajectories. In detail, to decide whether a new trajectory is erroneous or not, for each cluster (obtained after applying k-means) a corresponding HMM was built. The resulting k-HMMs were then used to evaluate the likelihood that the new trajectory belongs to one of the HMMs and if the maximum likelihood is smaller than a threshold, that trajectory was identified as erroneous. Although, this method looks promising, it requires training data to build the HMMs. Additionally, it is not totally parameter free, at least for k-means the number of clusters should be learnt from the training data. Applying this method to big data is not very feasible since the needed size of training data is not clear and there is no guarantee that the selected training data will be representative

enough.

The following part of this section focuses on describing big data and addressing the current considerations mainly about data cleaning. One of the most well-known methods for big data cleaning is applying outlier detection [15]. As the approach presented here is based on outlier detection, we also review outlier detection methods for the purpose of data cleaning.

### *2.1. Big Data and Current Considerations*

The definition of big data is domain specific even though the importance of analysing big data has been generally recognized [6]. Although big data mainly refers to vast volumes of raw data, there are also some other concerns. Chen et al. [16] defined big data as masses of unstructured data that traditional information technology equipment such as desktop workstations, non-clustered nodes, typical software tools are not able to gather, store, process, manage and analyse. In [17], big data was defined not only in terms of volume but also in terms of velocity and variety. In that paper [17], the velocity refers to producing and processing the data rapidly and on time to satisfy the demand while variety means various modalities such as image data from a sensor source, and text data from social networks, etc. In addition to volume, velocity and variety in [18, 19] veracity (accuracy, trustability of the data), variability (inconsistency in the data), value (usefulness of the data for decision making) and complexity (the degree of interconnectedness) were also added to the features of big data.

Big data analytics cover extracting meaningful patterns from massive raw data for prediction and decision making. Especially in the last decade, various companies utilized big data analytics to monitor and analyse their business needs and had a better understanding about their business which can lead to better customer service, improved products, increased sales, etc. However, the challenges of big data analytics still continue. Key problems are: exponential growth of data, need for suitable data storage, compression, transmission and data indexing. There are also problems such as variety of the raw data, highly distributed and various sources, high dimensionality, scalability of existing algorithms, imbalanced data, limited labelled data, lack of efficient information retrieval, noisy data and so forth [20].

As more video cameras exist in our lives, such as cameras embedded in our mobile phones, laptops, surveillance cameras in the buildings, ATMs, traffic, etc., the image and video generated by such devices has become the largest big data source [21]. These large amounts of video and image data have become attractive to the computer vision and machine learning research community, and which present great opportunities and new challenges. Automatic video annotation [22], parallel computing, developing scalable algorithms for scene understanding [23, 24], object detection [25], object tracking [26], object recognition [27, 28], video data visualization, image search, image retrieval and indexing [29] can be seen as some applications which become even more challenging with big data.

As mentioned above, one of the big challenges of big data is data cleaning which was also stated in [30] and this can require efficient data querying. Fan et al. [31] proposed a method to querying big data using a small amount of it. In that study [31], a concept called bounded envelope retrieves a sufficiently accurate subset which is a good approximation of the full data set. Similarly, in their previous work on scale independence

[32, 33] it was shown that the size of small representative dataset is more dependent on the query than the size of the full dataset. In [6], methods to remove false fish detections from a large fish image dataset were presented. It was shown that evaluating big data by sampling smaller subsets is the most convenient way of evaluating the cleaned dataset.

An alternative approach to handing errors in big datasets is not to clean the data, but rather to model the types and results of the errors on a smaller ground-truth dataset, and then use the model to correct the statistical results [34]. While this does not correct individual errors, it improves the overall results, under the assumptions that a given level and type of errors are expected in the full dataset.

Recently, active learning based data cleaning methods were proposed as well. Krishnan et al. [35, 36], proposed an iterative data cleaning tool which focuses on the errors such as missing data, incorrect or inconsistent values in the data. The proposed method [35, 36] cleans the data while preserving provable convergence properties.

Another popular approach for data cleaning is utilizing outlier detection methods which are also efficient on very large datasets [15]. An outlier is defined as a datum which deviates significantly from other data points where the quantity of outliers is less than the quantity of inliers [37]. In [5], to obtain a clean dataset for the purpose of fish recognition, clustering based outlier detection (see Section 2.2 for more information) was performed. In that study [5], the fish images were first clustered. The outliers were detected as the images which were not similar to the representative image (cluster centre) of the cluster that the outlier belongs to. Once an outlier was detected, they were removed from the data and the clean data were used for fish recognition task. In the following section, we review recent outlier detection methods which were utilized for data cleaning.

### *2.2. Outlier Detection for Data Cleaning*

Outlier detection methods have been proposed for various applications such as fraud detection, network intrusion, weather forecasting and many other data mining tasks. In this work, we focus on outlier detection based on a deep learning technique specifically for data cleaning.

In [15], data cleaning methods were addressed for different types of data for instance both qualitative and categorical. Particularly, for big qualitative data (which is the data type in this study as well), outlier detection was presented as the basis of data cleaning. In that study [15], different outlier detection methods were investigated with their advantages and disadvantages. Outlier detection methods were categorized as *i)* non-normality assumption based methods, *ii)* data modeling based methods, *iii)* data partitioning (clustering) based methods, *iv)* model-free detection, *v)* distance based methods, *vi)* time-series methods, *vi)* re-sampling based methods and *vi)* frequency based methods [15].

Among many different categories, clustering based, distance based and density based outlier detection methods are the most popular approaches for data cleaning. As an example of distance based outlier detection methods, Kollios, et al. [38] proposed a generic method which can be used for data cleaning as well. In that study [38], a data point is defined as an outlier if at most $k$ other points lie within a defined distance. Similarly, Knorr and Ng [39] proposed a metric to define an outlier threshold which is a percentage of data points at a distance from a data point. Breunig, et al. [40] defined the density as the average distance of a data point to its $k$ nearest neighbors where data points having

low density are defined as outliers. Loureiro, et al. [41] presented an approach which is based on hierarchical clustering to detect the errors in foreign trade data. In that study [41], small clusters were defined as the clusters which contain outliers assuming that they are distinct from the majority of the data. In a very recent study [42], outlier detection based on k-means and k-mediods clustering methods were presented which were applied for data cleaning.

Overall, although the proposed methods are useful, in practice, they usually require data point to point distance (similarity) calculations which makes them not scalable and their application for big data becomes not feasible. Motivated by this, in this study we propose an outlier detection method which is based on the clustering capability of mean-covariance Restricted Boltzmann Machine which does not require any similarity (distance) calculations.

## 3. Dataset

The dataset used here was based on video data captured as part of the Fish4Knowledge project [43]. The videos were captured from one of four fixed cameras (3.6 mm focal length, 2/3 inches CCD) in uncontrolled open sea conditions in the intake bay of the third Nuclear Power Plant (NPP) inside Kenting National Park. Simultaneous water temperature readings were stored with each video. The videos were analysed to detect and track fish using a covariance based tracker [26] while species recognition of individual fish was based on a balance-guaranteed optimized decision tree classifier [27]. We selected the data associated with the damselfish *Dascyllus reticulatus*, which lives in colonies, commonly feeding on zooplankton near coral heads. The camera used in this study was at a depth of 2 meters, and a typical image from the video data is shown in Figure 1. Because of the shallow camera and coral depth, the scene is greatly affected by illumination variations, arising from both changes in the sky lighting (sun position, clouds) and, more importantly, refraction of the light by the ocean surface causing caustics that can be mistaken for fish, or which might cause fish to be undetected.

In total, 12247 videos ($640 \times 480$ resolution, 10 minutes each, 24 frames per second) which is 2041.2 hours of data were analysed. 3649007 trajectories of *Dascyllus reticulatus* were identified and used in the analysis. To assess the quality of the automatically detected and analysed data, 1000 of the 3.6 million fish trajectories were manually examined, with 100 trajectories from each of the 10 temperature intervals chosen randomly (see Section 5 for the definition of intervals). Manual examination was performed as follows. For each detection of a given trajectory, we examined whether there was any false detection (the detected object is not a fish) or not. This condition included the requirement that the majority of the bounding box should contain the detected fish and if the detected object was a fish then it should have also been the same fish detected in the previous frames. We also checked whether there was any false recognition (the detected fish is not *Dascyllus reticulatus*). If there was at least one false detection or one false recognition, the corresponding trajectory was classified as an erroneous trajectory, otherwise it was classified as a correct trajectory. By looking at consecutive frames in the video, we also assessed whether the linked detections were likely to be from the same fish i.e. consistent with the fish's direction, motion and neighbours. These 1000 trajectories led to 16504 detections, of which 16210 were actually fish. 745 trajectories (11602

detections) of the 1000 trajectories were correctly tracked from frame to frame. All 745 trajectories were *Dascyllus reticulatus*, although we expect that there are some instances of mis-recognition in the full dataset. Consequently, we estimate that 74.5% of the 3.6 million trajectories are valid.

## 4. Methods

The central question to be answered was whether fish speeds (for this species) increased, decreased or had no trend as a function of water temperature. The temperature was measured directly on a per-video basis.

Each trajectory was represented by a set of $Traj = \{(c_t, r_t)\}_{t=1}^{T}$ such that $(c, r)$ are image column and row positions of the centres of the detection bounding boxes for a trajectory composed of $T$ detections. Although there were no direct 3D position estimates, the observed fish were largely mature adults, with a typical height of 33 mm (see [7] for the reasoning). These fish swim, on average, horizontally, with the result that there is often foreshortening of the apparent length of the fish due to its pose relative to the camera. On the other hand, the foreshortening of the apparent height of the fish is largely due to the distance of the fish from the camera. Consequently, we estimated the 3D position of the fish from the measured height of the bounding box, from which a set of $\{\vec{p}_t\}_{t=1}^{T} = \{(x_t, y_t, z_t)\}_{t=1}^{T}$ scene positions were estimated, and then the fish speed was estimated from the time taken to traverse the total distance travelled $\Sigma_{t=1}^{T-1} \mid\mid \vec{p}_{t+1} - \vec{p}_t \mid\mid$. Complete details of the speed estimation method are given in [7].

The general principles of the method are straightforward. Unfortunately, there are many factors that contribute to cause erroneous speed estimates, some of which are identified here:

1. A small percentage of detections (est. 2% based on the analysis in the previous section) are false positives, which can lead to unrealistic distances when combined with consecutive true positive detections.

2. A significant percentage of trajectories (est. 25%) are formed by connecting detections incorrectly. This can arise easily as *Dascyllus reticulatus* tend towards aggregating into schools, and thus leads to incorrect pairings. It is also possible for the tracking algorithm to connect detections that are substantial image distances away from the true trajectory. Both lead to erroneous 3D distances between consecutive detections.

3. Some fish are larger or smaller than the nominal sizes, which lead to erroneous depth and thus position and speed estimates.

4. Some fish are not swimming horizontally and thus their bounding box is larger than expected. This leads to closer position estimates and thus slower speed estimates.

5. Some bounding boxes are larger or smaller than the true fish, which again lead to erroneous size, distance and speed estimates.

6. There are natural variations in the behaviour (due to changing circumstances) and capability (due to the difference in metabolism).

One might be discouraged based on all these influences and it is quite likely that many of the measurements from individual fish are affected. However, we take the view that one can analyse the data from the general perspective of the 'Law of Large Numbers', whereby individual variations tend to cancel out to expose the underlying trends. Our data does not satisfy the conditions for the Law specifically, but follows the general perspective that enough data cancels fluctuations to reveal the underlying distribution. Figure 2 shows the histogram of speeds for the 28.1-30.3 centigrade temperature band (for more evidence, i.e. showing similar trend, interested readers can refer to supplementary material of [7]). It is clear that there is a significant peak irrespective of the considerable variation due to the factors identified above (and possibly more).

Below, we describe an approach that uses a mean-covariance Restricted Boltzmann Machine to find clusters which have few samples and are expected to contain fish trajectories that are significantly different from the vast majority of the trajectories, *i.e.,* erroneous trajectories (outliers). We claim that the trend found for fish speed and temperature using the remaining fish trajectories will be the same with the trend found using all fish trajectories. Showing the evidence of such a claim will prove that it is still possible to obtain statistically reasonable results, even with a dataset that contains much noise, once you have very big data.

### *4.1. Validation of the Method*

In this section, we present an outlier detection method which utilize clustering based on a mean-covariance Restricted Boltzmann Machine. Thanks to effectiveness of mean-covariance Restricted Boltzmann Machine, 3.6 million trajectories can be clustered without need of any similarity calculation between samples. Then, the outliers are detected and removed to apply the same pipeline (3D position estimation, speed calculation and data binning) presented in [7] to be able to compare the trends.

### *4.2. Trajectory Representation*

To cluster the trajectories, the mean-covariance Restricted Boltzmann Machine requires each trajectory to be represented with a vector of fixed length. Among several trajectory representation methods (such as vector quantization [44], Discrete Fourier Transform [45], Chebyshev Polynomial Approximation [46], the Haar Wavelet Transform [46], Principal Component Analysis (PCA) [47], and Hidden Markov Model (HMM) [48]). In this study, Cubic B-spline Control Points [49, 46] is applied.

This is mainly because *i)* Cubic B-spline Control Points is able to encode the shape and the spatiotemporal property of trajectory, *ii)* the control points and weight factors are flexible to encode any simple or complicated trajectory, *iii)* its better performance has been shown in studies such as [49, 46, 50] to detect the abnormal trajectories (which are outliers in other words) and last *iv)* it is not based on learning. The disadvantage of this method is to be based on the chosen number of control points which might result in ignoring sharp changes in trajectory if an incorrect number of control points is chosen but the similar disadvantages exist in other popular trajectory representation methods such as HMM and PCA.

Given that each trajectory is defined by a set of $Traj = \{(c_t, r_t)\}_{t=1}^{T}$ image column and row positions of the centres of the detection bounding boxes, the approximate of

cubic B-spline curve is

$$S = \{ \sum_{i=1}^{p} C_i^c B_{i,4}(t), \sum_{i=1}^{p} C_i^r B_{i,4}(t) \} \tag{1}$$

where $p$ is the number of control points, 4 is fixed because the order of the function is 3 for cubic spline, $\vec{C}^r$ and $\vec{C}^c$ are the unknown control points while every control point is based on a B-spline function which is shown as $B_{i,4}(t)$.

The B-spline basis functions are defined by a knot vector $\vec{\tau}$ (which determines where and how the control points affect the spline curve, see [46] for the values used) as follows [49, 46, 50]:

$$\begin{aligned} B_{i,1}(t) &= \begin{cases} 1 & if \tau_i \leq t < \tau_{i+1} \\ 0 & otherwise \end{cases} \\ B_{i,m}(t) &= \frac{t - \tau_i}{\tau_{i+m-1} - \tau_i} B_{i,m-1} + \frac{\tau_{i+m} - t}{\tau_{i+m} - \tau_{i+1}} B_{i+1,m-1} \end{aligned} \tag{2}$$

The $p$ coefficients which minimise the sum of squared errors between the original trajectory ($Traj$) and its approximation $S$ are found by the Moore-Penrose pseudoinverse operator ($\Phi$) that can be defined as:

$$\begin{aligned} \Phi &= \begin{bmatrix} B_{1,4}(t_1) & ... & B_{p,4}(t_1) \\ ... & ... & ... \\ B_{1,4}(t_T) & ... & B_{p,4}(t_T) \end{bmatrix} \\ \text{and} \qquad \Phi^\dagger &= (\Phi^T \Phi)^{-1} \Phi^T \end{aligned} \tag{3}$$

while the control points are defined as $F^{CR} = \Phi^\dagger Traj^{CR}$ where $F^{CR}$ is the final trajectory representation.

In this work, we set the number of control point to seven as applied in [49, 46, 50]. Seven is a reasonable number given that the median and the mean values of the trajectory lengths are 12 and 20, respectively.

### *4.3. Mean-covariance Restricted Boltzmann Machine (mcRBM)*

Using Restricted Boltzmann Machines (RBMs) for clustering has some advantages over other clustering methods: *i)* Clustering methods like k-means and hierarchical clustering require pairwise similarity (distance) calculations, whereas RBMs do not require this. *ii)* There exist clustering methods, e.g. Gaussian Mixture Model (GMM) and Dirichlet Process Mixture Model (DPM), that also do not need pairwise comparisons. However, for GMM, the number of mixtures (i.e. the number of clusters) should be known. On the other hand, for DPM, although it is non-parametric and can learn the number of mixture components without being specified in advance, the behaviour of the model is sensitive to the choice of prior base measure. Moreover, it needs to calculate mean and covariance for each component, and update covariance with Cholesky decomposition, which may lead to high space and time complexity [51]. *iii)* As mentioned in [52], other

clustering methods (such as GMM) as opposed to RBMs need a huge number of clusters to capture all the variations in the input whereas a reasonably small RBM can capture very complicated distributions, since RBM is able to discover a rich representation of the input. This is because $N$ hidden units can represent up to $2^N$ different regions in input space. With other clustering techniques, one would need $O(2^N)$ parameters (and/or examples) to capture that many regions, unlike RBMs which require $O(N)$ parameters.

The mean-covariance Restricted Boltzmann Machine (mcRBM) is a type of Restricted Boltzmann Machine (RBM) that is capable of feature learning from real-valued data [53]. Similar to all RBM models, mcRBM has a bipartite undirected graph structure. It contains two layers of stochastic random variables, which are also called units. The first layer is a visible layer that represents the observed data, namely *visible units* ($v$). The second layer has latent variables that are also referred to as *hidden units* ($h$). Different than standard RBM models, the mcRBM has two sets of hidden units: *i)* mean units ($h_m$) and *ii)* covariance units ($h_c$). The $h_m$ units model the mean of the input elements, while the $h_c$ ones represent the pairwise dependencies between the visible units, hence modeling their covariance structure. There are no connections between variables within the layers, thus, the variables of a layer are independent of each other (see Figure 3 for an illustration of the mcRBM).

The mcRBM is a combination of a Gaussian RBM (gRBM) and a covariance RBM (cRBM) [53]. Its energy function is composed of two terms and is defined as follows:

$$E_{mc}(v, h_c, h_m) = E_c(v, h_c) + E_m(v, h_m). \tag{4}$$

$E_c$ defines a zero-mean Gaussian distribution over the visible variables and is given by:

$$E_c(v, h_c) = -d^T h_c - (v^T R)^2 P h_c \tag{5}$$

where $R$ is the visible-factor weight matrix, $P$ is the factor-hidden (or "pooling") matrix and $d$ is the hidden bias vector. $E_m$ is defined as:

$$E_m(v, h_m) = \frac{1}{2}(v - b)^T(v - b) - c^T h_m - v^T W h_m \tag{6}$$

with $W$ denoting the direct connections from the hidden mean units ($h_m$) to the visible units ($v$), $b$ is visible bias and $c$ is the hidden mean bias.

By having $E_m$, different than the cRBM, the mcRBM can produce conditional distributions over the visible units given the hidden units, that have non-zero means. The conditional distribution of the hidden covariance units given the visible unit states $v$ is:

$$p(h_c|v) = \sigma(d + ((v^T R)^2 P)^T) \tag{7}$$

where $\sigma(x) = 1/(1 + exp(-x))$ is the logistic function and the conditional distribution over the mean hidden variables is:

$$p(h_m|v) = \sigma(c + W v^T) \tag{8}$$

The resulting conditional distribution over the visible variables is a Gaussian distribution which is in terms of the hidden covariance and hidden mean latent states and is defined as:

$$p(v|h_c, h_m) \propto N\left(\Sigma W h_m, \Sigma\right) \tag{9}$$

where $\Sigma$ is given by:

$$\Sigma = \left(R(\text{diag}(-P^T h_c))R^T\right)^{-1}. \tag{10}$$

In this paper, as recommended in [53] and used in [54] for mice behaviour analysis, the normalized version of mcRBM is used to avoid the feature based disparity. It is trained using the common $RBM$ parameter $\theta \in (R, P, W, d, b)$ with stochastic gradient ascent:

$$\Delta\theta \propto \langle -\frac{\partial E}{\partial \theta}\rangle_{data} + \langle \frac{\partial E}{\partial \theta}\rangle_{model} \tag{11}$$

where $\langle\cdot\rangle$ denotes expectations under the distributions specified by the subscript. To compute the expectations under the model distribution, we use Hybrid Monte Carlo on the mcRBM's free energy which allows obtaining reconstructions (as suggested in [54]).

Lastly, by using the mcRBM, clusters are obtained by using the different binary feature configurations of the model's latent variables which represent different modes of the input data distribution.

The hyperparameters used to train the mcRBM are given in Section 5. All training parameters were randomly initialized to small values as suggested in [53] and match the example configuration of the code accompanying [53]. Normalization of weight matrices ($R$, $P$) also matches the code accompanying [53]. Regarding convergence, looking at the energy function, we observed that for the current data the network was converging after 1000 training epochs. In order to avoid fast convergence, differently than [53] we didn't use annealing of the learning rates, because we observed that after few epochs, the network was not learning anymore.

### *4.4. Outlier Detection*

An outlier is a datum which is far away from the other data points in a dataset. Usually, the cardinality of the outliers is less than the other data samples. In this study, we adapted the outlier detection algorithm presented in [37] such that outliers are the data points located in small clusters. Small clusters are defined as the clusters having fewer trajectories which can be identified by a threshold. This threshold is calculated as follow. The cardinality of each cluster is calculated and the median of them are found. Later, the threshold is defined as A% the found median value (see Section 5 for the values used as $A$). If a cluster's cardinality is smaller than that threshold then that cluster is determined as a small cluster. All trajectories which belong to a small cluster are detected as outliers and removed for further analysis (3D position estimation, speed calculation and data binning) assuming that they are false detections and incorrect trajectories and the remaining dataset is cleaned.

### *4.5. Data Binning*

Once 3D position estimation and speed calculation are performed for each individual fish trajectory using the method given in [7], the temperature data and the corresponding speeds are grouped into 10 bins. Data binning is used to find whether fish speeds (for this species) increased, decreased or had no trend as a function of water temperature. In detail, each bin contains a similar number of trajectories. Since there were more data for some temperatures and much less data available for other temperatures (see Figure 4),

such a data binning was the only reasonable way compared to another alternative which is dividing data into equal temperature intervals. Additionally, if the number of bins was more than 10, the resulting number of trajectories in each bin is not as similar as when we set the number of bins equal to 10. On the other hand, if the number of bins was less than 10, this merges the data such that some bins correspond to a very wide temperature interval as compared to other bins. This might make the analysis less accurate in case there are fluctuations in the speed as temperatures rise.

The temperature intervals and the number of trajectories (whole data (not cleaned)-without outlier detection) in each interval are given in Table 1.

## 5. Results

As stated previously, the central question to be answered was whether fish speeds increased, decreased or had no trend as a function of water temperature. As discussed in [7], the swimming speed of *Dascyllus reticulatus* was found to increase as the water temperature increased. What is interesting in this paper are the methods used to ensure that this conclusion was sound.

### *5.1. Results Without Data Cleaning*

Data binning as described in the previous section helped to cope with the unequal distribution of data, particularly at the higher temperatures where the standard deviation of speeds are higher. Table 2 summarizes the data in terms of mean, median, mode (extracted after histogram of the data is smoothed) and the standard deviation of speeds associated with this binning. One can observe that: mean, median and mode speeds are generally increasing with temperature. The trend for speed increase is weak for bins 4-6, however note that these bins combined cover only about 1 degree, whereas bins 1 and 10 cover several degrees individually. Thus, it is not surprising that bins 4-6 have similar mean, median and mode speeds.

In Figure 4a, the speed data versus temperature is given. For each bin, the mean (shown with white circle) and the median (shown with white diamond) were marked by taking the middle temperature value of each bin as the vertical coordinate. Additionally, the box plot representation of the same data is shown in Figure 4b (adapted from [7]). In that Figure, speeds are truncated at 100 mm/sec to present the trend clearer (the maximum speed is visible in Figure 4a). The horizontal axis is not uniformly spaced because we chose to use bins that had approximately equal numbers of samples. The boxes are bounded by the upper and lower quartile of the data, the mid-bar is the data median value, and the whiskers show the most extreme speeds. As seen, as well as the increasing mean, the median had an increasing trend with temperature. Because there was a significant amount of data in each bin, powerful statistical tests could be applied to confirm the hypothesis of an increasing speed trend. The Kruskal-Wallis significance test (which does not assume either normality or homogeneity of variance, i.e. having approximately equal variance on the scores across groups) was applied to assess whether the speeds in the different bins were significantly different. The test showed that the mean ranks for each temperature interval are significantly different from each other, which implies that the speeds for each temperature interval are significantly different ($p < 0.05$). Tukey-

Kramer post-hoc analysis was applied to analyse the speeds of each pair of temperature intervals. The results again showed that the speed distributions are significantly different for each pair of temperature intervals. To answer the central question of this paper which is that the speed had a trend, or was random as a function of water temperature, the Mann-Kendall test ($p < 0.05$) was applied to the mean, median and mode speeds for each temperature interval. As a result, it was obtained that an **increasing trend** existed for the mean, median and mode speeds with a p-value of 0.0056, 0.0095 and 0.0056, respectively.

### *5.2. Results With Data Cleaning*

The same data binning procedure was applied to the cleaned data which was obtained after the proposed outlier detection method was applied. The temperature intervals were kept the same for comparison while a similar number of trajectories were obtained for each bin as well. mcRBM was trained with different parameters for hidden covariance units ($h_c$ = 14), hidden mean units ($h_m$ = $\{10, 20\}$), batch size = 256 and epochs = $\{900, 1600\}$ (see Section 4.3 for the definition of parameters). For outlier detection, the *A* parameter (see Section 4.4 for the definition) was taken as 1, 5, 10, 15, 20 and 25. By using different parameter settings for mcRBM, we obtained different numbers of clusters. However, the total number of remaining trajectories (even the number of trajectories per bins) did not change much (and therefore the mean, median and standard deviation were changed only in the 0.001 place) no matter which outlier detection parameter was taken. This is perhaps because mcRBM is good at clustering especially when there is a large amount of data (3.6 million). In detail, we observed that by increasing the number of the latent variables, the number of clusters was gradually increasing. We also realized that there is a certain number of hidden units beyond which there is no point of adding more, since they will stay in a non-active state during the experiment, meaning they do not affect the clustering results in anyway. We reached a good balance between computational cost and quality in results using the aforementioned set of latent variables as well.

Figure 5 shows the histogram of speeds for the 28.1-30.3 centigrade temperature band after data cleaning is applied. When Figure 2 is compared with Figure 5, it is seen that the mode of the histograms did not change much although the mean and the median decreased slightly after data cleaning. In Table 3, the number of trajectories, mean, median, mode (extracted after histogram of the data is smoothed) and the standard deviation of speeds associated with each bin for the cleaned data are given. After removing outliers, in total 2239021 trajectories ($\sim 61\%$ of the whole data) remained for analysis (which are not detected as an outlier). These results were obtained when 123 clusters (123 unique activations) were found after applying mcRBM when $h_c$, $h_m$, batch size and epochs were set as 14, 10, 256 and 1600, respectively and A was taken as 10. The number of visible to hidden covariance factors was kept equal to the number of hidden covariance units in all the experiments, as suggested in [54]. When we changed A from 25 to 1 only 10276 trajectories (0.28% of whole data) more were detected as outliers. Additionally, distributions of control points (features; see Section 4 for definition) in some of the dense and small clusters (not all due to space limitation) are shown in Figure 6 which corresponds to results given in Table 3. As seen, the behaviour of different clusters are distinctive (there are different patterns) while each cluster (especially the dense clusters which are not outliers) has a low standard deviation per control points. This means that

the intra-cluster similarity is high while inter-cluster similarity is low as expected from a successful clustering algorithm.

The Mann-Kendall test ($p < 0.05$) was also applied to the mean, median and mode speeds of cleaned data. As a result, an **increasing trend** was obtained for the mean, median and mode speeds with a p-value of 0.0032, 0.0056, and 0.0032, respectively. The same p-values were also obtained when different mcRBM parameters and the outlier detection parameters were used.

When Table 2 and 3 are compared, it can be seen that the mean values of each bin changed which is expected since the proposed method removed (a lot of) outliers. The median values also all went down after data cleaning which can be interpreted as the data cleaning removed a lot of data from upper tail as well. On the other hand, the modes remained almost the same after data cleaning.

### *5.3. Evaluation of Data Cleaning Results*

As mentioned in many studies and also in Section 2, one way of estimating errors in a dataset is evaluating the performance of it on random small subsets of the data. As given in Section 3, such an analysis, which performed by manual examination, showed 25.5% error which is notably close to the number of outliers ($\sim$ 29% of the whole data) that were automatically detected by the proposed outlier detection algorithm. In detail, *i)* 52% of the data which were identified as erroneous trajectories by manual investigation were also detected as outliers by the proposed outlier detection algorithm, *ii)* 78% of the data which were identified as correct trajectories by manual investigation were also detected as inliers (not outliers) by the proposed outlier detection algorithm. *iii)* 22% of correct trajectories identified by manual investigation were removed as outliers by the proposed algorithm. *iv)* 48% of erroneous trajectories identified by manual investigation were detected as correct trajectories by the proposed algorithm. The net result in the cleaned dataset is now estimated to consist of 82.6% correct trajectories (as compared to 74.5%).

## 6. Figures

***Fig. 1.*** *Typical camera view.*

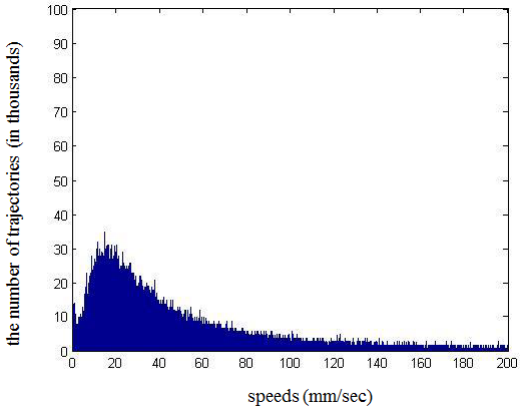


***Fig. 2.*** *Distribution of trajectories (in thousands) vs speeds (in mm/sec, shown until 200 mm/sec) for 364645 trajectories in the 28.1-30.3 centigrade temperature band.*

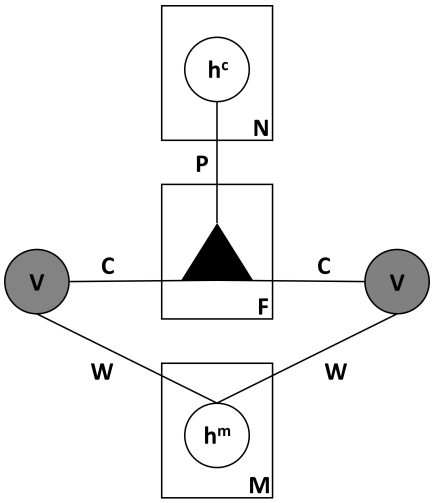


***Fig. 3.*** *Illustration of Mean-covariance Restricted Boltzmann Machine (mcRBM). Adapted from [54].*

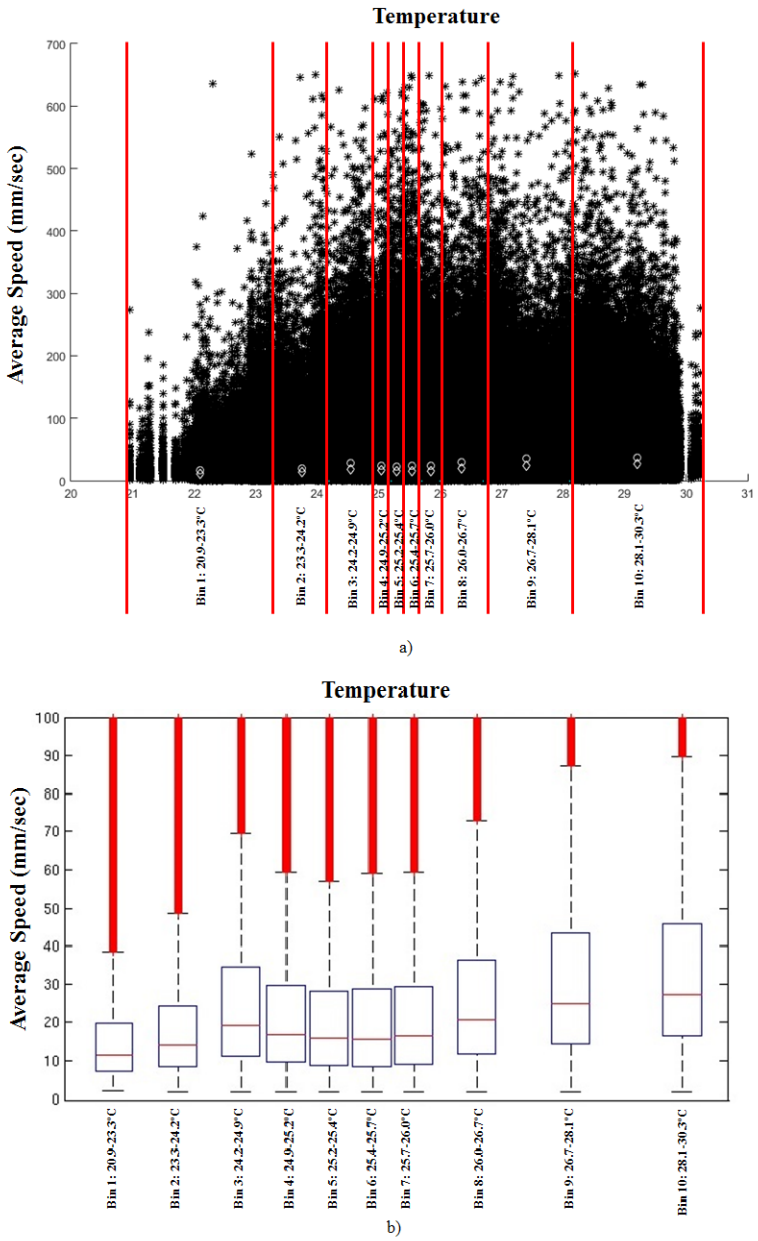


***Fig. 4.*** *Fish Speed vs. Temperature without data cleaning. a) Mean (shown with white circle) and median (shown with white diamond) were marked by taking the middle temperature value of each bin as the vertical coordinate. There are more data for some temperatures and much less data available for other temperatures. b) Box plots for each bin. Speeds are truncated at 100 mm/sec. Outliers found by robust statistics are shown individually with red plus signs (appears as thick bars). The boxes are bounded by the upper and lower quartile of the data, the mid-bar is the data median value, and the whiskers show the most extreme speeds excluding the outliers. This figure is based on [7].*

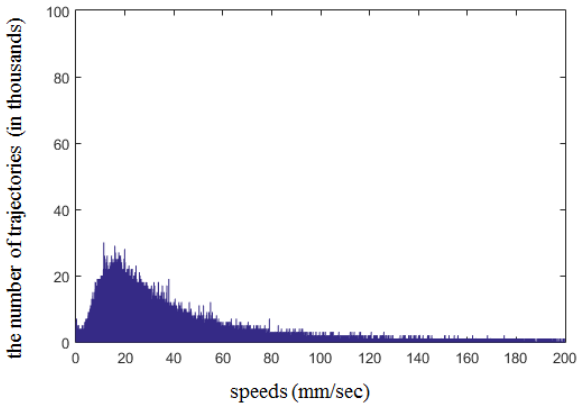


***Fig. 5.*** *Distribution of trajectories (in thousands) vs speeds (in mm/sec, shown until 200 mm/sec) after data cleaning was applied in the 28.1-30.3 centigrade temperature band.*

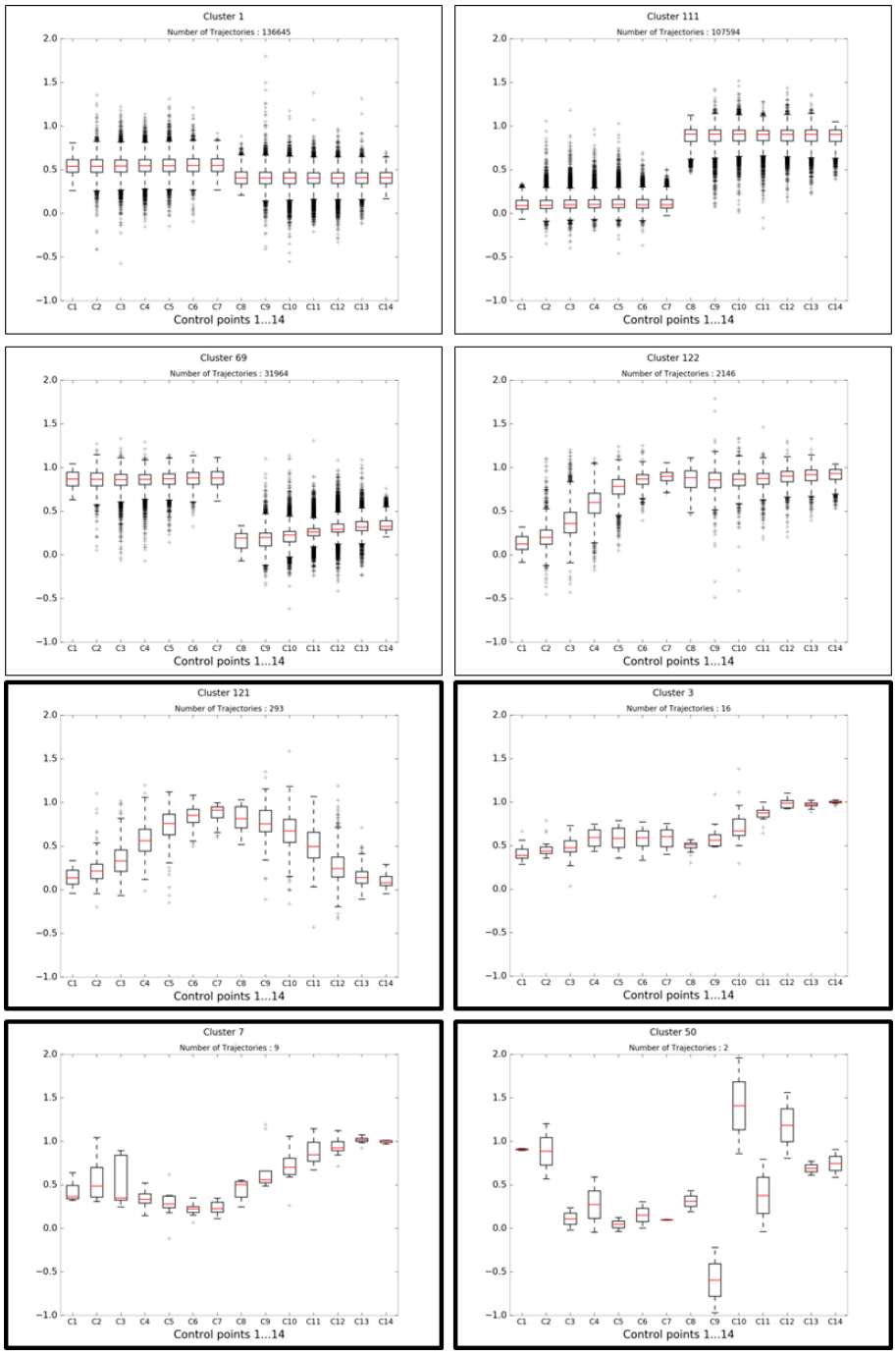


***Fig. 6.*** *Distributions of control points which belong to randomly selected dense clusters and small clusters (outliers). Clusters having less than 975 trajectories were classified as outliers, and are outlined with a thicker border. The control points C1-C7 are the column coordinates and C8-C14 are the row coordinates for the seven real 2D control points (shown as $F^{CR}$ in Section 4.2). This figure shows that the behaviour of different clusters are distinctive, hence there are different patterns per cluster. Particularly the dense clusters have a low standard deviation per control points, meaning that their intra-cluster similarity is high.*

### 6.1. Figure Captions

Figure 1: Typical camera view.
Figure 2: Distribution of trajectories (in thousands) vs speeds (in mm/sec, shown until 200 mm/sec) for 364645 trajectories in the 28.1-30.3 centigrade temperature band.
Figure 3: Illustration of Mean-covariance Restricted Boltzmann Machine (mcRBM). Adapted from [54].
Figure 4: Fish Speed *vs.* Temperature without data cleaning. a) Mean (shown with white circle) and median (shown with white diamond) were marked by taking the middle temperature value of each bin as the vertical coordinate. There are more data for some temperatures and much less data available for other temperatures. b) Box plots for each bin. Speeds are truncated at 100 mm/sec. Outliers found by robust statistics are shown individually with red plus signs (appears as thick bars). The boxes are bounded by the upper and lower quartile of the data, the mid-bar is the data median value, and the whiskers show the most extreme speeds excluding the outliers. This figure is based on [7]
Figure 5: Distribution of trajectories (in thousands) vs speeds (in mm/sec, shown until 200 mm/sec) after data cleaning was applied in the 28.1-30.3 centigrade temperature band.
Figure 6: Distributions of control points which belong to randomly selected dense clusters and small clusters (outliers). Clusters having less than 975 trajectories were classified as outliers, and are outlined with a thicker border. The control points C1-C7 are the column coordinates and C8-C14 are the row coordinates for the seven real 2D control points (shown as $F^{CR}$ in Section 4.2). This figure shows that the behaviour of different clusters are distinctive, hence there are different patterns per cluster. Particularly the dense clusters have a low standard deviation per control points, meaning that their intra-cluster similarity is high.

## 7. Tables

**Table 1** Data Binning: Temperature range of each bin and the number of trajectories corresponds to each bin (whole data).

| Bin | Temperature Range (C) | Number of Trajectories |
|---|---|---|
| 1 | 20.9-23.3 | 364946 |
| 2 | 23.3-24.2 | 364884 |
| 3 | 24.2-24.9 | 365258 |
| 4 | 24.9-25.2 | 365011 |
| 5 | 25.2-25.4 | 364543 |
| 6 | 25.4-25.7 | 364845 |
| 7 | 25.7-26.0 | 365035 |
| 8 | 26.0-26.7 | 365975 |
| 9 | 26.7-28.1 | 363865 |
| 10 | 28.1-30.3 | 364645 |

**Table 2** Summary of estimated speed in each temperature bin when whole data is analysed.

| Bin | Mean Speed (mm/sec) | Median Speed (mm/sec) | Mode Speed (mm/sec) | Standard Deviation (mm/sec) |
|---|---|---|---|---|
| 1 | 16.3 | 10.7 | 6.39 | 19.2 |
| 2 | 20.2 | 13.3 | 7.53 | 24.7 |
| 3 | 27.8 | 18.7 | 9.02 | 30.1 |
| 4 | 23.9 | 16.0 | 7.93 | 26.6 |
| 5 | 22.7 | 15.2 | 7.65 | 25.7 |
| 6 | 23.2 | 15.0 | 8.27 | 28.0 |
| 7 | 23.8 | 15.8 | 7.81 | 27.1 |
| 8 | 29.4 | 20.1 | 12.12 | 31.8 |
| 9 | 34.8 | 24.4 | 13.47 | 35.9 |
| 10 | 36.9 | 26.5 | 14.44 | 35.3 |

**Table 3** Summary of estimated speed in each temperature bin when cleaned data is analysed.

| Bin | Number of trajectories | Mean Speed (mm/sec) | Median Speed (mm/sec) | Mode Speed (mm/sec) | Standard Deviation (mm/sec) |
|---|---|---|---|---|---|
| 1 | 220608 | 13.6 | 10.2 | 7.09 | 12.4 |
| 2 | 259718 | 16.9 | 12.6 | 8.47 | 15.3 |
| 3 | 206958 | 23.3 | 17.8 | 10.12 | 19.8 |
| 4 | 251967 | 19.1 | 14.5 | 9.76 | 17.1 |
| 5 | 227706 | 18.6 | 14.1 | 9.34 | 17.1 |
| 6 | 208107 | 18.9 | 14.0 | 8.66 | 18.6 |
| 7 | 186589 | 20.3 | 15.4 | 8.08 | 18.8 |
| 8 | 220496 | 23.6 | 18.2 | 13.00 | 20.5 |
| 9 | 232394 | 27.5 | 21.4 | 14.56 | 23.4 |
| 10 | 224478 | 29.9 | 23.7 | 15.46 | 22.6 |

### 7.1. Table Captions

Table 1: Data Binning: Temperature range of each bin and the number of trajectories corresponds to each bin (whole data).
Table 2: Summary of estimated speed in each temperature bin when whole data is analysed.
Table 3: Summary of estimated speed in each temperature bin when cleaned data is analysed.

## 8. Discussion

The analysis presented in this paper showed that it was possible to obtain statistically sound conclusions from image data, even in the presence of many sources of errors that corrupt the data and subsequent results. At a minimum approximately 25% of the data suffered from detection and tracking errors, but almost certainly a large amount of the correctly tracked data was also corrupted to some degree by failing to conform to the model of the ideal horizontally moving fish accurately delimited by its bounding box.

Nonetheless, we showed that the conclusions are sound, essentially because the major errors led to outlier values that could be easily eliminated, and many other errors led to over and under estimations that tended to balance out, leaving a significant and obvious mode to the estimated speeds. In other words, having on the order of 300 thousand data points allows the mode value to be obvious. Although we had no ground truth to validate the correctness of the estimated speeds, the speeds were reasonable from a marine ecology perspective. More importantly, even if there was a systematic error in the scaling of the speeds (*i.e.,* if the real speeds were greater than the estimated speeds), this error would affect all of the results, but the conclusions about the trend were based on a relationship that was invariant to the actual result scaling.

This paper has presented a specific example of 'big visual data' analysis, but the question is what can readers take from this example. Clearly, the time of having completely clean training and test data has passed, simply because of the volume of data now available. It still makes sense to develop and evaluate algorithms using small clean datasets, but when large data sets start to be analysed, such forms of precision computer science are no longer feasible. Hence, one has to: *i)* develop algorithms that aim for unbiased error distributions (so the Law of Large Numbers could apply), *ii)* aim for measurements and properties that are robust to the many uncontrollable factors that affect the data and *iii)* evaluate performance on random small subsets of the data. A fourth approach, is to develop outlier detection methods capable of cleaning out a lot of erroneous data although this also risks eliminating unanticipated subclasses of true positive data whose behaviour did not conform to the data model.

In addition to all the discussions given above, it should not be ignored that this work presents reliable evidence for an increase in fish speed as water temperature increases (here, readers should consider the temperature interval used and also the fish species investigated). Robust statistics (i.e. median values, also presented in our early work [7]) and the proposed pipeline for data cleaning (trajectory representation, clustering using mcRBM and outlier detection) are the approaches applied to prove that the found trend is reliable. It is shown that this scientific conclusion is significant even the data includes

erroneous trajectories. In this paper, the key point is to show that the cleaning process gives the same conclusion, but using a cleaner dataset has its own intrinsic value because one can now do other analyses better with the cleaner data e.g. abundance estimation, size distribution estimation, etc.

The justified fish swimming speed trend is directly related to fish behaviour understanding which is an important research topic for marine biology. The trend is complementary with findings of marine biologists (see [7] for more information), which in a way shows that using fully automated computer vision systems like [43] can be helpful to marine biologist in their analysis.

In addition, one can adapt the presented work (trajectory representation, mcRBM and outlier detection parts) for trajectory analysis (not only for analysis of fish trajectories but for other types of trajectories) for instance to detect the abnormal trajectories (in other words; rare trajectories, unusual trajectories or anomaly detection). Another application can be using the trajectory representation algorithm and the mcRBM for automatic biometric identification (recognition) based on handwriting.

## 9. References


[1] 'MAED 2014: The 3rd ACM Int. Regular and Data Challenge Workshop on Multimedia Analysis for Ecological Data', http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=37434, accessed November 2016

[2] 'Visual observation and Analysis of Vertebrate and Insect Behavior 2014', http://homepages.inf.ed.ac.uk/rbf/vaib14.html, accessed November 2016

[3] 'ImageCLEF/ LifeCLEF- Multimedia Retrieval in CLEF', http://www.imageclef.org/node/181, accessed November 2016

[4] Zhou, J., Bai, X., Caelli, T. (Eds.): 'Computer Vision and Pattern Recognition in Environmental Informatics' (IGI-Global, 2015)

[5] Fisher, R.B., Chen-Burger, Y.-H., Giordano, D., Hardman, L., Lin, F.-P. (Eds.): 'Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data' (Springer, 2016)

[6] Pugh, M.: 'Removing False Detections from a Large Fish Image Dataset'. Master Thesis, School of Informatics, University of Edinburgh, 2015

[7] Beyan, C., Boom, B.J., Liefhebber, J.M.P, Shao, K.-T., Fisher, R.B.: 'Natural Swimming Speed of Dascyllus reticulatus Increases with Water Temperature', ICES Marine Science, 2015, 72, (8), pp. 2506-2511

[8] Johansen, J.L., Jones, G.P.: 'Increasing ocean temperature reduces the metabolic performance and swimming ability of coral reef damselfishes', Global Change Biology, 2011, 17, pp. 2971-2979

[9] Johansen, J.L., Messmer, V., Coker, D.J., Hoey, A.S., Pratchett, M.: 'Increasing ocean temperatures reduce activity patterns of a large commercially important coral reef fish', Global Change Biology, 2014, 20, pp. 1067-1074

[10] Palazzo, S., Murabito, F.: 'Fish Species Identification in Real-Life Underwater Images', Proc. 3rd ACM International Workshop on Multimedia Analysis for Ecological Data, 2014

[11] Stern, U., He, R., Yang C.-H.: 'Analyzing animal behavior via classifying each video frame using convolutional neural networks', Scientific Reports, 2015, 5, (14351), pp. 1-13

[12] Ohayona, S., Avni, O., Taylor. A.L., Peronaa, P., Egnor, S.E.R.: 'Automated multiday tracking of marked mice for the analysis of social behaviour', Journal of Neuroscience Methods, 2013, 219, pp. 10-19

[13] Kumar, S., Singh, S.K.: Visual animal biometrics: survey, 'IET Biometrics', 2017, 6, (3), pp. 139:156

[14] Spampinato, C., Palazzo, S.: 'Hidden Markov Models for detecting anomalous fish trajectories in underwater footage', Proc. Int. Workshop on Machine Learnig for Signal Processing, 2012

[15] Hellerstein, J.M.: 'Quantitative data cleaning for large databases', United Nations Economic Commission for Europe, 2008, pp. 1-42

[16] Chen, M., Mao, S., Liu, Y.: 'Big Data: A Survey', Mobile Networks and Applications, 2014, 19, (2), pp. 171-209

[17] Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., Byers, A.H.: 'Big data: the next frontier for innovation, competition, and productivity', McKinsey Global Institute, 2011

[18] Kaisler, S., Armour, F., Espinosa, J.A., Money, W.: 'Big Data: Issues and Challenges Moving Forward', Proc. IEEE Hawaii International Conference on System Sciences, 2012, pp. 995-1004

[19] Gani, A., Siddiqa, A., Shamshirband, S., Hanum, F.: 'A survey on indexing techniques for big data: taxonomy and performance evaluation', Knowledge and Information Systems, 2016, 46, pp. 241-284

[20] Najafabadi, M.M., Villanustre, F., Khoshgoftaar, T.M., Seliya, N., Wald, R., Muharemagic, E.: 'Deep learning applications and challenges in big data analytics', Journal of Big Data, 2015, 2, (1), pp. 1-21

[21] Huang, T.: 'Surveillance Video: The Biggest Big Data. Computing Now', IEEE Computer Society (online), 2014, 7, (2)

[22] Kavasidis, I., Palazzo, S., Salvo, R., Giordano, D., Spampinato, C.: 'An innovative web-based collaborative platform for video annotation', Multimedia Tools and Applications, 2013, 7, (2), pp. 1-20

[23] Alexander, J.: 'Scene Understanding for Real Time Processing of Queries over Big Data Streaming Video', Department of Electrical Engineering and Computer Science, The University of Central Florida, PhD Thesis, 2013

[24] Xiao, J.: 'A 2D + 3D Rich Data Approach to Scene Understanding', PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 2013

[25] Kumar, P.: 'High Performance Object Detection on Big Video Data using GPUs', Proc. Int. Conf. on Multimedia Big Data, 2015, pp. 383-388

[26] Spampinato, C., Palazzo, S., Giordano, D., Lin, F.P., Lin, Y.T.: 'Covariance-based fish tracking in real-life underwater environment', Proc. Int. Conf. on Computer Vision Theory and Applications, 2012, pp. 409-414

[27] Huang, P., Boom, B., Fisher, R.: 'Underwater live fish recognition using a balance-guaranteed optimized tree', Proc. Asian Conference on Computer Vision, 2012, pp. 422-433

[28] Krizhevsky, A., Sutskever, I., Hinton, G.E.: 'ImageNet Classification with Deep Convolutional Neural Networks', Proc. Neural Information and Processing Systems, 2012, 25, pp. 1106-1114

[29] Shang, L., Yang, L., Wang, F., Chan, K.-P, Hua, X.-S.: 'Real-Time Large Scale Near-Duplicate Web Video Retrieval', Proc. ACM International Conference on Multimedia, 2010, pp. 531-540

[30] Tang, N.: 'Big Data Cleaning', Web Technologies and Applications, Lecture Notes in Computer Science. Springer International Publishing, 2014, 8709, pp. 13-24

[31] Fan, W., Geerts, F., Cao, Y., Deng, T., Lu, P.: 'Querying Big Data by Accessing Small Data', Proc. Association for Computing Machinery Symposium on Principles of Database Systems, 2015, pp. 173-184

[32] Fan, W., Geerts, F., Neven, F.: 'Making Queries Tractable on Big Data with Preprocessing: Through the Eyes of Complexity Theory', Proc. VLDB Endowment, 2013, 6, (9), pp. 685-696

[33] Fan, W., Geerts, F., Libkin, L.: 'On Scale Independence for Querying Big Data', Proc. Association of Computing Machinery Symposium on Principles of Database Systems, 2014, 6, (9), pp. 51-62

[34] Boom, B.J., Beauxis-Aussalet, E., Hardman, L., Fisher, R.B.: 'Uncertainty-Aware Estimation of Population Abundance using Machine Learning', Multimedia Systems, 2015, pp.1-13

[35] Krishnan, S., Wang, J., Wu, E., Franklin, M.J., Goldberg, K.: ActiveClean: Interactive Data Cleaning For Statistical Modeling, Proc. VLDB Endowment, 2016, pp. 1-12

[36] Krishnan, S., Franklin, J.M., Goldberg, K., Wang, J., Wu, E.: 'ActiveClean: An Interactive Data Cleaning Framework For Modern Machine Learning', Proc. SIGMOD, 2016, pp. 2117-2120

[37] Beyan, C., Fisher, R.B.: 'Detection of Abnormal Fish Trajectories Using a Clustering Based Hierarchical Classifier', Proc. British Machine Vision Conference, 2013, pp. 1-11

[38] Kollios, G., Gunopulos, D., Koudas, N., Berchtold, S.: 'Efficient biased sampling for approximate clustering and outlier detection in large data sets', IEEE Trans on Knowledge and Data Engineering, 2003, 15, (5), pp. 1170-1187

[39] Knorr, E.M., Ng, R.T.: 'Finding intensional knowledge of distance-based outliers', Proc. Int. Conf. on Very Large Data Bases, 1999, pp. 211-222

[40] Breunig, M.M., Kriegel, H.-P., Ng, R.T., Sander, J.: 'Lof: Identifying density-based local outliers', Proc. ACM SIGMOID International Conference on Management of Data, 2000, pp. 93-104

[41] Loureiro, A., Torgo, L., Soares, C.: 'Outlier Detection using Clustering Methods: A Data Cleaning Application', Proc. KDNet Symposium on Knowledge-Based Systems for the public Sector, 2004

[42] Kaur, P., Kaur, K.: 'A Review on Outlier Detection for Data Cleaning in Data Mining', International Journal of Innovative Research in Computer and Communication Engineering, 2016, 4, (7), 14373-14376

[43] Boom, B.J., He, J., Palazzo, S., Huang, P.X., Beyan, C., Chou, H., Lin, F., Spampinato, C., Fisher, R.B.: 'Research tool for the analysis of underwater camera surveillance footage', Ecological Informatics, 2013, 23, pp. 83-97

[44] Morris, B.T., Trivedi, M.M.: 'A survey of vision-based trajectory learning and analysis for surveillance', IEEE Trans. on Circuits and Systems for Video Technology, 2008, 18, (8), pp.1114-1127

[45] Naftel, A., Khalid, S.: 'Classifying spatiotemporal object trajectories using unsupervised learning in the coefficient feature space', Multimedia Systems, 2006, 12, pp. 227-238

[46] Sillito, R.R., Fisher, R.B.: 'Parametric trajectory representations for behaviour classification', Proc. British Machine Vision Conference, 2009, pp.1-11

[47] Bashir, F., Wu, Q., Khokhar, A., Schonfeld, D.: 'HMM- based motion recognition system using segmented PCA', Proc. IEEE International Conference on Image Processing, 2005, pp. 2286–2289

[48] Porikli, F.: 'Learning object trajectory patterns by spectral clustering', Proc. IEEE Conference Multimedia Expo, 2004, pp. 1171-1174

[49] Sillito, R.R., Fisher, R.B.: 'Semi-supervised learning for anomalous trajectory detection', Proc. British Machine Vision Conference, 2008, pp. 227-238

[50] Li, C., Han, Z., Ye, Q., Jiao. J.: 'Abnormal Behavior Detection via Sparse Reconstruction Analysis of Trajectory', Proc. International Conference on Image and Graphics, 2011, pp. 807-810

[51] Chen, G.: 'Deep learning with nonparametric clustering', arXiv preprint arXiv:1501.03084. 2015, pp. 1-14

[52] Bengio, Y., Courville, A., Vincent, P.: 'Representation learning: a review and new perspectives', IEEE Trans. Pattern Anal. Machine Intell., 2013, 35, pp. 1798-1828

[53] Ranzato, M., Hinton, G.E.: 'Modeling Pixel Means and Covariance Using Factorized Third-Order Boltzmann Machines', Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 2551-2558

[54] Katsageorgiou, V.M., Huang, H., Ferretti, V., Papaleo, F., Sona, D., Murino, V.: 'Unsupervised Mouse Behavior Analysis: A Data-Driven Study of Mice Interactions', Proc. International Conference on Pattern Recognition, 2016