IEEE Transactions on Circuits and Systems for Video Technology, Volume 28, Issue 8, August 2018, pp. 1850 - 1863
DOI: 10.1109/TCSVT.2017.2701860

Pair-Activity Analysis from Video Using Qualitative Trajectory Calculus

Alaa AlZoubi, Bashir Al-Diri, Tom Pike, Tanja Kleinhappel and Patrick Dickinson

***Abstract*—The automated analysis of interacting objects or people from video has many uses, including the recognition of activities, and identification of prototypical or unusual behaviors. Existing techniques generally use temporal sequences of quantifiable real-valued features, such as object position or orientation; however, more recently, qualitative representations have been proposed. In this paper we present a novel and robust qualitative method which can be used both for classification and clustering of pair-activities. We use Qualitative Trajectory Calculus (QTC) to represent the relative motion between two objects, and encodes their interactions as a trajectory of QTC states. A key element is a general and robust means of determining the sequence similarity, which we term *Normalized Weighted Sequence Alignment*; we show that this is an effective metric for both recognition and clustering problems. We have evaluated our method across three different datasets, and shown that it out-performs state of the art quantitative methods, achieving an error rate of no more than 4.1% for recognition, and cluster purities higher than 90%. Our motivation originates from an interest in automated analysis of animal behaviors, and we present a comprehensive video dataset of fish behaviors (*Gasterosteus aculeatus*), collected from lab-based experiments.**

***Index Terms*—Pair-activity analysis, qualitative trajectory calculus, sequence alignment.**

I. Introduction

THE use of computer vision as a method for analyzing behavior has attracted significant research interest, with typical applications in human, vehicle and animal-related contexts. Most work is motivated by either an interest in identifying the occurrence of specific pre-determined events, the unsupervised modeling of typical behaviors, or the detection of unusual behaviors. Both supervised and unsupervised learning techniques have been used, for example in the analysis of single objects [1]–[3], and group activities [4]–[12]. A smaller body of work has been directed specifically at pair-activity analysis [13]–[15], which may be seen as a special case of group activities. However, since human and animal interactions are typically pair-wise, these may also be considered as the building blocks of large group behaviors.

Previous research concerned with pair-activity analysis has been mainly focused on quantitative methods which use sequences of real-valued features (trajectories). However, increasing attention has been given to the use of qualitative methods, which use symbolic rather than real-value features, with applications such as human-robot interaction [16] and geographic information science [17]. There are a number of motivations for this interest in qualitative methods:

- Humans naturally conceptualize, reason, and communicate in qualitative ways rather than by using quantitative measurements [17], particularly when describing interactions and behaviors.
- Qualitative reasoning (in particular Qualitative Trajectory Calculus) has been shown to be effective in a number of existing works (e.g. [16]–[19]).
- Qualitative representations are typically more compact and computationally efficient than quantitative methods [2], [20]–[22].

Van der Weghe [17] describes Qualitative Trajectory Calculus (QTC) as a calculus for representing and reasoning about movements of objects in a qualitative framework. The essential features of these interactions are encoded symbolically, and describe the system state at any given time-step. Sequences of QTC states may be used to describe interactions between two objects over a given period of time. In this paper, we present our method for pair-activity clustering and classification (recognition), based on QTC. Our work is primarily motivated by our interest in the automated analysis of interactions between fish (the three-spined stickleback, *Gasterosteus aculeatus*), and we have developed a large video dataset of fish interactions, which we use in our evaluations. However, our motivations are wider than this single application. In order to gain traction as a main-stream analysis technique, QTC based methods require a generic and robust method of trajectory comparison, and also proper evaluation against existing quantitative techniques. We accordingly present our new metric for trajectory similarity, and detailed comparative work against state-of-the-art quantitative methods, using multiple datasets (in addition to our own). Our work is the first to make these comparisons. Furthermore, our results show that our proposed qualitative method outperforms current quantitative methods for both pair-wise trajectory classification and clustering, in different challenging applications. Fig. 1 shows the main components of our method.

A. Contributions

The key contributions of our work are as follows:

- We present a new adaptive metric for comparing the similarity of QTC sequences of different lengths, extracted from video data, which we term *Normalized Weighted Sequence Alignment* (*NWSA*). Our metric is developed from

Alaa AlZoubi is with the Centre for Electronic Warfare Information and Cyber, Cranfield University, Defence Academy of the United Kingdom, Shrivenham, SN6 8LA (e-mail: a.s.alzoubi@cranfield.ac.uk).

B. Al-Diri and P. Dickinson are with School of Computer Science, University of Lincoln, Lincoln, UK (e-mail: {baldiri,pdickinson}@lincoln.ac.uk).

T. Pike and T. Kleinhappel are with the School of Life Science, University of Lincoln, Lincoln, UK (e-mail: {tpike,tkleinhappel}@lincoln.ac.uk)

Manuscript received xxx xx, xxxx; revised xxx xx, xxxx.

© 2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

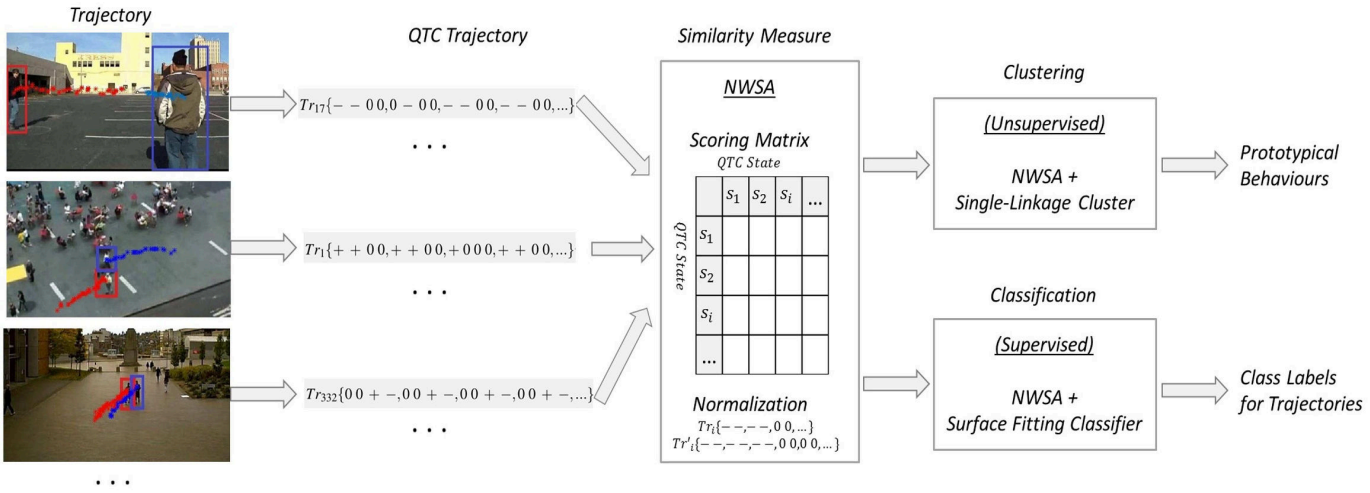


Fig. 1. Overview of the Proposed Method.

previously proposed QTC sequence alignment methods [18], but learns feature weightings from training data: we show experimentally that it can robustly compare complex sequences in different contexts (datasets), for both classification and clustering purposes. Our method is the first to show this level of generalization for QTC, enabling its wider use in video analysis.

- We propose new methods for pair-activity classification and clustering based on QTC and *NWSA*. Our classification method uses Surface Fitting (*SF*) [4] to match input QTC sequences against class exemplars, using *NWSA* as a similarity measure. Our clustering method uses Single-Linkage Clustering (SLC), again using *NWSA* as a similarity measure. We show experimentally that our proposed methods out-perform existing *state-of-the-art* quantitative methods (such as [4]). Our experiments are the first to directly and robustly compare methods using quantitative and qualitative representations for video analysis.
- We evidence the overall generality of our methods with evaluations against three datasets: two of these are pre-existing and publicly available video datasets (human and vehicle). We also introduce our own, new, video dataset of fish behavior, *FISHBEHAVE*, which is ground-truthed and consists of 322 video segments of pair-wise interactions. FISHBEHAVE is publicly available for other researchers studying animal behavior, and pair-activity analysis in general.
- Our experimental work is the first which directly uses trajectory clustering (quantitative and qualitative) to study interactions between fish in a lab-based environment.

B. Structure of this Paper

The remainder of this paper is organised as follows:

- In Section II we present background work, focussing on existing uses of qualitative reasoning and QTC, with an overview of comparable quantitative methods and trajectory similarity metrics.
- Our proposed methods for supervised classification, and unsupervised clustering, of pair-activities is presented in full in Section III.
- Section IV provides details of our evaluation and experimental results, using challenging, publicly available datasets. We also introduce our dataset, FISHBEHAVE.
- Our concluding remarks are presented in Section V.

II. Background and Related Work

A. Qualitative Spatial and Temporal Reasoning

Qualitative spatio-temporal reasoning is a method for representing and analyzing interactions between objects using symbolic rather than real-valued features, and is grounded in concepts of human cognition [17]. A number of qualitative representations have been previously proposed, such as Cardinal Direction Calculus [23] and Double-Cross Calculus [21]. However, these calculi capture only either spatial or temporal features, but not both. Qualitative Trajectory Calculus (QTC) [17] was developed specifically to overcome such limitations, and capture the full spatiotemporal features of moving objects. We also note that Double-Cross Calculus is intended for cases where one of a pair of objects is moving; QTC can represent cases where both are moving. As a result, QTC in particular has attracted interest from computer vision researchers studying interactions between humans, robots, and animals.

QTC, proposed by Van der Weghe in [17], encodes interactions between Moving Point Objects (MPOs) using qualitative symbols: {-, 0 and +}. Various features are represented in this way; for example, a decrease in the distance between one MPO and another during a time period t is represented by the symbol "-". The symbol "+" indicates an increase, and "0" represents no change (to within some tolerance). Given two objects (Obj_1 and Obj_2), the full QTC representation of the change in their relative positions between times t_1 and t_2 is given by a set of symmetric features or *codes*:

- $Code_1$: distance of Obj_1 with respect to Obj_2: "-" indicates decrease, "+" indicates increase, "0" indicates no change.

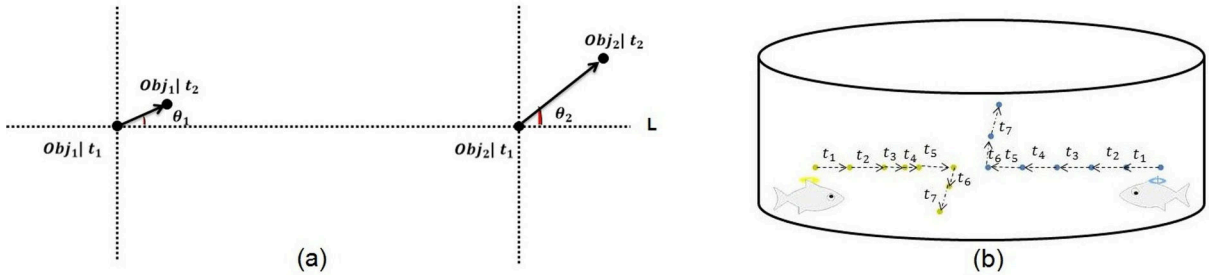


Fig. 2. (a) Example of QTC relations between Obj_1 and Obj_2 at time t_1 and t_2 (-,+,-,-,-,-). (b) Example of spatial interactions between two fish; where both fish are converging during the time interval t_1 to t_5, and they diverging at t_6 and t_7.

- $Code_2$: distance of Obj_2 with respect to Obj_1.
- $Code_3$: Relative speed of Obj_1 with respect to Obj_2 (which dually represents the relative speed of Obj_2 with respect to Obj_1)
- $Code_4$: Displacement of Obj_1 with respect to the reference line L connecting the objects: "-" if moves to the left, "+" if it moves to the right.
- $Code_5$: Displacement of Obj_2 with respect to L.
- $Code_6$: The respective angles between the velocity vectors of the objects and vector L (θ_1 and θ_2 in Figure 2): "-" if $\theta_1 < \theta_2$, and "+" if $\theta_1 > \theta_2$.

Fig. 2(a) illustrates the concept of qualitative relations in QTC for the two disjoint objects; Obj_1 and Obj_2. Three QTC variants have been defined, namely:

- Basic QTC (QTC_B) [17] using only $Code_1$ and $Code_2$.
- Double Cross QTC (QTC_C) [17] which adds the direction features: $Code_1$, $Code_2$, $Code_4$, $Code_5$.
- Full QTC (QTC_{Full}) which uses the full set of codes: $Code_1$, $Code_2$, $Code_3$, $Code_4$, $Code_5$, $Code_6$.

Some code combinations are excluded, such that QTC_{Full} can actually represent 305 distinct and valid states [17]. Pairs of analogous codes are sometimes referred to collectively as a QTC feature; for example, $Code_1$, $Code_2$ may be referred to as the *distance feature*, and $Code_4$, $Code_5$ as the *side feature*.

Some notable existing works have used QTC representations. Hanheide et al. [16] presented a case study for analyzing human-robot behavior using QTC_C, and a probabilistic behavior model. They used a Markov chain of joint spatial behavior, and the edit distance [24], to compare traces of joint spatial behavior. However, this method has limitations: state repetitions are collapsed into a single symbol, which may lead to the loss of important information. The edit distance similarity measure has its own limitations (see Section II-C). The method was applied only on trajectories of the same length, and only for a small number of interactions. More recently, a method for analyzing dance movement using QTC has been presented in [18]. The problem domain of this work corresponds more closely to ours and the method comprises three main components: transformation of movements into QTC sequences; sequence alignment; and comparison of sequences using the conceptual distance.

TABLE I
SUMMARY TRAJECTORY ANALYSIS METHODS.

Method	Type	Feature	Application
[13] [14]	Recognition	causality ratio, feedback ratio, velocity, and relative distance	human pair-activity
[15]	Recognition	relative distance and relative velocity	human pair-activity
[5]	Recognition	group center, motion histogram, closeness histogram, and centrality histogram	human small-group
[7]	Recognition	self causality, pair causality, and group causality	human group
[9]	Recognition	energy feature and attraction and repulsion features	human group
[6]	Analysis	speed and motion azimuth	football players
[11]	Recognition	distance, speed, direction, change of width and height, speed difference	human group
[4]	Recognition	heatmap	pair and group
[16]	Clustering and Classification	QTC_C	human-robot
[25]	Analysis	speed, motion azimuth, and vertical angle	dance movement
[18]	Clustering	QTC_B and QTC_C	dance movement

B. Trajectory Analysis

Previous works studying trajectory-based activity analysis mainly use quantitative features, and a review is provided by [26]. Trajectory-based approaches can be divided into three categories: Single-role activities ([3], [27], [28]); Pair-activities ([13]–[15]); and Group-activities ([4]–[12]). Table I summarizes for the most relevant existing work. Pair activities are most closely related to our own work: Zhou et al. [13] classified human pair-activities using motion information and Granger Causality Features (*GCF*). The method was applied to a video dataset of human interactions, but performed badly on complex pair-activities. They further extended their work in [14]. Sethi et al. [15] introduced an atomic pair actions dataset which includes 90 video samples for three different human pair actions.

A larger body of work addresses the general case of multiple interactions. Yin et al. [5] proposed a method for

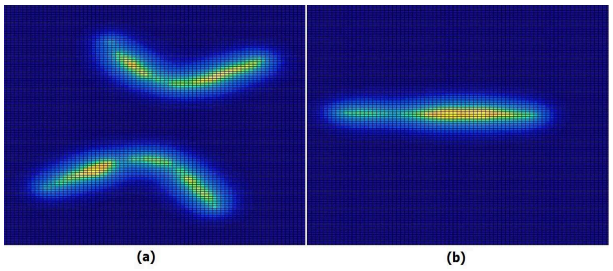


Fig. 3. Heat Map representations for the pair-activities (a) *BothTurn* and (b) *Follow*.

human activity classification using social network analysis and a conditional Gaussian-process dynamic model. Ni et al. [7] used localized causalities: interactions are represented within, between, and among motion trajectories, encoding the group-activities with three types of causalities (self, pair, and group). This can be seen a development of [13], using a 20-dimensional feature vector including causality ratio, feedback ratio, frequency magnitudes, relative distance, and speed. Both [7] and [13] have been experimentally evaluated in [4], which we use to benchmark our own work. Kim et al. [9] recognized human group activities using two types of features: group interaction energy, and attraction/repulsion features. A method for discovering relative motion patterns (REMO) in groups of moving point objects was presented in [6]. REMO describes motion patterns as changes in attributes, such as the speed and motion azimuth, and was used to football players and the movements of deer. However, it depends on expert knowledge, is sensitive to small time distortions, and changes to the entity order in the REMO matrices make it difficult to interpret.

Lin et al. in [11] proposed a method for human group event detection which accounted for varying group size, and used a hierarchical activity structure. More recently, Lin et al. in [4] proposed an effective heatmap-based algorithm for human group and pair activity recognition. The method first represents human trajectories as a series of heat sources; then, a thermal diffusion process creates an activity map. Finally, a surface-fitting (SF) method was developed to recognize the activities, described by:

$$a^* = arg_a min(min_{Ta}||T_a.S_{HM} - S_{SD,a}||) \qquad (1)$$

where S_{HM} is the heatmap, $S_{SD,a}$ is a predefined standard activity, T_a is the alignment operator, and a^* is the output class. The method was applied to vehicle interactions, and human pair-activities, and was shown to be superior to several existing approaches ([7], [11], [13]). Fig. 3 shows an example heatmap for two activities. The method does rely on accurate alignment, but has been shown experimentally as the most effective method for pair-wise activity recognition. We thus adopt it as a benchmark quantitative method, against which we evaluate our own work.

C. Similarity Measures

Several approaches have been used for measuring the similarities of sequences of symbols (or strings). These include edit distance [24], bag-of-words models [29], sequence alignment [1], string kernels [30], [31], Hamming distance [32], and cosine similarity [33].

The edit distance quantifies the similarity between two strings as the minimum cost to transform one into the other as a series of discrete changes. Using this method, each edit incurs the same cost-penalty, and the cost is independent of the substituted value. *Sequence alignment* is closely related, but uses a substitution matrix to assign scores to matches and mismatches, and a gap penalty for insertion and deletion. It has been used in many contexts including DNA sequencing [34], human activity analysis [1], and others ([35]–[37]). Dynamic programming algorithms (e.g. [38]) are used to minimize a distance measure, or maximize a similarity measure, using conventional operations (substitution, insertion and deletion). An extensive discussion is provided by Rosenberg [39]; however, it is important to note the limitations of these methods. The assumption that all edits are of the same cost is simplistic (e.g. where features correspond to physical processes). Differences in sequence lengths also results in additional edits (costs) which may not be meaningful, do not generalize, or which may lose important information (e.g. [16]). Our work directly addresses these limitations. Chavoshi et al. [18] used a scoring system based on the *conceptual distance*, using both QTC_B and QTC_C representations. This is defined as the number of changes required to incrementally change one value to another; for example the distance between "-" and "+" is two, because the transition must pass through the intermediate state [40]. Thus, a similarity score between two QTC_B states may then be calculated as $(4 - CD)$. The insertion/deletion penalties used for gap opening and extensions are -5 and -3, respectively. However, the scoring matrix proposed in this method is designed specifically for the dance movement dataset, is context specific, and is designed to deal only with sequences of very similar lengths.

Dynamic Time Warping (DTW) [41] has also been widely used to align signals of different durations. It uses non-linear signal warping through dynamic programming, and has some similar characteristics to symbolic sequence alignment methods (but is more often used with real-valued data). Some experimental comparisons have been made between DTW and other sequence alignment methods: for example, Grachten et al. [42] showed that a warped version of the Needleman-Wunsch algorithm (which is closely related to our own method), outperforms DTW for aligning musical sequences. This was because DTW was less able to handle structural differences, where sequences are composed of smaller, repeating, compound sequences.

Existing works for QTC analysis have used sequence alignment methods or edit distance (e.g [16]) rather than DTW. In their work on dance sequence analysis, Chavoshi et al. used both DTW with real-valued features [25], and QTC with sequence alignment [18], as analysis tools. They present a discursive comparison of DTW and sequence alignment methods in [18], but favour sequence alignment for their QTC approach because it is easy to interpret visually, and can be used to align multiple sequences. We have also followed a sequence alignment approach in our own work, which we use

with normalised QTC sequences. This is primarily because we develop from existing QTC methods, which are based on sequence alignment. However, there are also other motivations: our approach allows us to easily incorporate transition weightings for QTC code-pairs (learned from training data), into the substitution scoring matrix; and we also wish to deal with compound behaviours in our fish dataset, which are potentially more suited to a sequence alignment approach [42].

III. Our Method

Our proposed method comprises three main components:

- We represent interactions in video sequences using QTC_B, QTC_C and QTC_{Full}.
- We introduce Normalized Weighted Sequence Alignment (*NWSA*) as a general and adaptive metric for comparing sequence similarity.
- We either cluster similar trajectories using Single-Linkage Clustering (SLC), or classify trajectories using Surface Fitting (*SF*) [4].

Our *NWSA* metric for comparing QTC trajectories embeds context-dependent feature weighting, learned from the data, and accounts for significant differences in sequence length without incurring inappropriate costs, or losing important information. As we will show, our method generalizes across different data contexts, and enables us to consistently outperform state-of-the-art quantified methods.

A. Extraction of QTC Trajectories

We extract QTC features from video data using low-level image processing techniques (detection and tracking). These are applied frame-by-frame to produce standard quantitative features (e.g. position) from which the corresponding QTC code sequences are derived. We report results from evaluation on three datasets. Two of these are publicly available (vehicles [4] and human [15]), and the third is our own fish dataset (FISHBEHAVE) which is also publicly available. For the first two datasets, positions of the objects were provided. For our fish dataset we automatically extracted the required features: the fish were tagged using small circular discs ([43]) attached to the dorsal spine (see Fig. 8). We developed an automated visual tracking method which estimates the tag pose, which is fully described in [44], and comprises the following components:

- *Camera Calibration* to estimate the intrinsic parameters.
- *Image Enhancement* to eliminate the image noise and detect the region of interest.
- *Edge Detection* to detect the elliptical projection of the circular tag, as a set of pixel edge points.
- *Lens Distortion Compensation* to eliminate the noise caused by lens distortion.
- *Elliptical Feature Extraction* which include the center point, semi-major and semi-minor axes.
- *Estimation the 3D orientation* of the circular tag.

For all data sets, the 2D object positions in the image frame were used to construct corresponding QTC codes. For our experiments we have used all three variants: QTC_B, QTC_C and QTC_{Full}. For example, in the interaction between the two fish in Fig. 2(b): both fish are converging during the time interval t_1 to t_5, and then diverge between t_6 and t_7. This interaction is described using QTC_B: $(-\ -, -\ -, -\ -, -\ -, -\ -, +\ +, +\ +)_{t_1-t_7}$. The QTC_C representations for same sequence is given by: $(-\ -\ 0\ 0, -\ -\ 0\ 0, -\ -\ 0\ 0, -\ -\ 0\ 0, -\ -\ 0\ 0, +\ +\ +\ +, +\ +\ +\ +)_{t_1-t_7}$. The interaction is also be described using QTC_{Full} by including the speed and angle features (the expansion is omitted here).

Definition: Given sets of x, y coordinates (centroid positions) of two interacting objects, we define: $P_i = \{(x_1, y_1), ..., (x_t, y_t), ..., (x_{L_i}, y_{L_i})\}$, and $P'_i = \{(x'_1, y'_1), ..., (x'_t, y'_t), ..., (x'_{L_i}, y'_{L_i})\}$, where (x_t, y_t) is the centroid of the first interacting object at time t and (x'_t, y'_t) is the centroid of the second. The pair-wise trajectory is then defined as a sequence of corresponding QTC states: $Tr_i = \{Q_{i,1}, ..., Q_{i,t}, ..., Q_{i,L_i}\}$, where $Q_{i,t}$ is the QTC state representation the relative position/movement of the two objects (x_t, y_t) and (x'_t, y'_t) at time t in trajectory Tr_i; and L_i is the number of frames in Tr_i.

B. Normalized Weighted Sequence Alignment

We contribute a generic metric for comparing the similarity of QTC trajectories, which we refer to as Normalized Weighted Sequence Alignment (*NWSA*). Our method develops from existing approaches to QTC sequence alignment (e.g. [18]), but uses learned feature transition weights, and generalizes robustly to sequences of different lengths (our motivation for this approach to measuring QTC sequence similarity was outlined in section II-C). The challenge is to optimally align and compare any two QTC sequences, in order to determine their similarity (whether they represent similar activities); this facilitates both classification and clustering of object-pair behaviors. The robustness of this metric is dependent on the form of the scoring matrix, which defines the similarity between two individual QTC states, and also on the sequence alignment process which determines how sequences of different lengths are compared.

1) Feature Weighting: Existing alignment methods use different costs for substitution, insertion and deletion. However, none differentiates between the substitution costs for different QTC codes. As mentioned in Section II-C, Chavoshi et al. [18] used a substitution matrices based on the *conceptual distance* [45]. This expresses the physical meaning of code transitions, but still assumes that different QTC features and codes have the same significance (weighting).

Our approach develops from [18], on the premises that the assumption of equal substitution costs for different codes/features is simplistic; and that any method of differential weighting should capture the application context. We therefore introduce a general and data-driven method for assigning different substitution costs to different QTC features. In time series analysis, long and short-term variations are treated separately [46] as some feature transitions are less likely, and therefore represent a more significant dissimilarity. This is both intuitive and statistically motivated: for example, an approach

Algorithm 1 Substitution (Score) Matrix for QTC States

Input set of trajectories $\zeta = \{Tr_1, ..., Tr_i, ...Tr_N\}$ where N is the number of trajectories in ζ.
Input QSt; QSt is $n_s \times 1$ matrix, where n_s is the number of QTC states (9, 81 or 305 for QTC_B, QTC_C and QTC_{Full}, respectively)
Output $n_s \times n_s$ substitution matrix ($Score_Matrix$)
Count Fc_k for each feature k, across all the trajectories ζ
Find $Fc_{min} = Min_k(Fc_k)$
Calculate the weights for each feature by computing $W_k = Fc_{min}/Fc_k$
Calculate the $Score_Matrix$
for $i = 1$ to n_s **do**
  for $j = 1$ to n_s **do**
    $score = \sum_1^K (CD(QF(i,k), QF(j,k)) \times W(k))$
    $Score_Matrix(i,j) = score$
  end for
end for
return $Score_Matrix$

similar to feature weighting has been previously motivated by Dodge et al. [2] for Geographical Information Systems. In the case of behavior analysis, particular QTC codes map onto physical processes, and so certain transitions may represent higher costs to an individual. For example, in the case of fish, a change of direction may require more effort, and so the substitution cost should capture this as a higher dissimilarity. Accordingly, we define the four QTC features which map onto the six codes defined in Section II as follows:

- *feature1* (Distance): $\{code_1, code_2\}$
- *feature2* (Side): $\{code_4, code_5\}$
- *feature3* (Speed): $\{code_3\}$
- *feature4* (Angle): $\{code_6\}$

We compute weights for each feature from the dataset: assuming each to be independent, transitions between feature values are summed across consecutively occurring QTC states. This results in a transition count per feature Fc_k, where $k \in \{1, ..., 4\}$. The lowest count across all features is found as Fc_{min}. A corresponding weight W_k is then computed for each feature as: $W_k = \frac{Fc_{min}}{Fc_k}$. The substitution (score) matrix (SM) is then defined as:

$$SM_{i,j} = \sum_k CD(QF(i,k), QF(j,k)) \times W_k \qquad (2)$$

Where $QF(i,k)$ is feature k of QTC state i, $QF(j,k)$ is feature k of QTC state j, and CD is the conceptual distance [45] between features $QF(i,k)$ and $QF(j,k)$. Note that the states here refer to the possible QTC state values. This is described fully in Algorithm 1. The weightings are then used to weight the code transitions in the corresponding scoring matrix. An example scoring matrix for QTC_B is shown in Table II: this shows the conceptual distance between different codes, as used by Chavoshi et al. [18]. In our method we use an analogous matrix, but weight each transition using the computation shown in Algorithm 1.

TABLE II
SUBSTITUTION (SCORING) MATRIX FOR QTC_B STATES, WHERE $W_1 = 1$.

	(- +)	(+ -)	(0 -)	(- -)	(- 0)	(0 +)	(+ +)	(+ 0)	(0 0)
(- +)	0	4	3	2	1	1	2	3	2
(+ -)	4	0	1	2	3	3	2	1	2
(0 -)	3	1	0	1	2	2	3	2	1
(- -)	2	2	1	0	1	3	4	3	2
(- 0)	1	3	2	1	0	2	3	2	1
(0 +)	1	3	2	3	2	0	1	2	1
(+ +)	2	2	3	4	3	1	0	1	2
(+ 0)	3	1	2	3	2	2	1	0	1
(0 0)	2	2	1	2	1	1	2	1	0

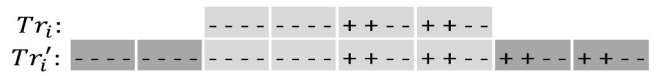


Fig. 4. Example of normalizing QTC_C trajectory using Algorithm 2, where Tr_i is the given trajectory and Tr_i' is the normalized trajectory with $S_i = 2$.

2) Trajectory Normalization: The similarity of activities or behaviors needs to account for varying sequence length in a generic way. Chavoshi et al. [18] only compared sequences of similar lengths, and Hanheide at al. [16] removed code repetitions. However, inspection of our datasets suggests that such repetitions are significant characteristics of particular activities (such as "following" behavior). We consider that there should be a high similarity score between the same activity when repeated more quickly or slowly; however, the relative duration of subsequences of identical QTC codes is a significant characteristic which we wish to preserve. This leads us to the concept of sequence normalization, and in section IV we demonstrate in our evaluation that this conceptual approach generalizes well across our data sets.

We normalize the length of all trajectories $Tr_i \in \zeta$. We first determine the length ($L_{max} = Max_i(L_i)$) of the longest trajectory in the input dataset; then, given a trajectory Tr_i with length L_i, we compute a scaling factor $S_i = L_{max}/L_i$. This scaling factor is used to resample the trajectory Tr_i to create a scaled trajectory Tr_i' where $L_i' = L_{max}$, using the procedure described in Algorithm 2. Fig. 4 shows an example of normalizing a QTC trajectory using this algorithm. Algorithm 3 describes the complete *NWSA* method, achieved by combining Algorithms 1 and 2. *NWSA* may also be seen as a development of the NeedlemanWunsch Algorithm [38] for sequence alignment, which uses data-driven weightings for code substitutions. Although we have normalized our trajectories, we have included insertion or deletion operations (gap penalties) in the algorithm, as in principal it is possible that their cost may be less than the cost of substitution (though not in our implementation).

C. Pair-Activity Classification

In our evaluation we compare our qualitative method with the heatmap-based method presented in [4], which has itself been shown to out-perform other quantitative methods ([7], [11], [13]). We wish to determine the ability of QTC to capture the characteristics of each behavior class, and so our proposed classification method combines QTC trajectories with the Surface Fitting scheme proposed by Lin et al. [4]. This compares an input trajectory with exemplars (standard

Algorithm 2 Normalized Sequence Alignment

Input set of trajectories $\zeta = \{Tr_1, ..., Tr_i, ..., Tr_N\}$, where N is the number of trajectories in the dataset
Find the length $L_{max} = Max_i(L_i)$, length of the longest trajectory in ζ
for $i = 1$ to N **do**
  Calculate the scale factor $S_i = \frac{L_{max}}{L_i}$
  Define The normalized sequence Tr'_i of length L_{max}
  for $j = 1$ to L_{max} **do**
    $Tr'_i(j) = Tr_i(floor(\frac{j \times L_i}{L_{max}}))$; where the $floor$ is a round down to the next integer function, and $Tr'_i(j)$ is the j^{th} state of sequence Tr'_i
  end for
end for
Output: $\zeta' = \{Tr'_1, ..., Tr'_i, ..., Tr'_N\}$

trajectories) of each class: the best match is taken as the class label.

In our case, we use *NWSA*, as a similarity metric, and define a standard trajectory for each activity from the corresponding training set, by selecting an exemplar for each class. We choose the training example with the lowest total intra-class matching score. That is, given a subset of trajectories $\zeta^f = \{Tr^f_1, ..., Tr^f_i, ..., Tr^f_{Nf}\}$ as the training set for class F and Nf is the number of elements in class F, we choose the exemplar F_{min} such that:

$$F_{min} = argmin_j \left(\sum_{i=1}^{Nf} NWSA(Tr_i, Tr_j), \quad i \neq j \right) \quad (3)$$

Each input trajectory is then compared with the standard trajectory for each class, and is assigned the class label with the lowest matching score.

D. Pair-Activity Clustering

Our ultimate objective is to identify prototypical behaviors, particularly in our FISHBEHAVE dataset, using QTC trajectories. That is, we wish to partition the trajectory set $\zeta' = \{Tr'_1, ..., Tr'_i, ..., Tr'_N\}$ such that within-group object similarity is maximized and the between-group object similarity is minimized [47]. We have used Single-Linkage Clustering (*SLC*), which is a hierarchical, bottom-up method [47] that defines the distance between two clusters as the closest cross-cluster pair. It has been shown to be sensitive to noise and outliers, but can also represent non-elliptical clusters. *SLC* does not require manually tuning, and has been previously used with symbolic sequences, including QTC trajectories ([18] and [16]). For comparative purposes, we have used the same clustering method in conjunction with the heatmap features generated by [4] as a benchmark quantitative method.

The most commonly used method for identifying clusters using *SLC* is a *fixed height branch cut*. However, this method is not ideal for more complicated dendrograms. We therefore used the *Dynamic Tree Cut* method proposed by Langfelder et al. [48], which uses a set of criteria to define the cut point, based on cluster shape. It has several advantages, can identify nested clusters, and parameters can be tuned based on the application.

Algorithm 3 *NWSA* Algorithm

Input: trajectories Tr_1 and Tr_2, with lengths of L_1 and L_2, respectively
Input: *Score_Matrix* from Algorithm 1
Input: $Gap_{Penalty}$; insert and delete penalties
Normalize the trajectories using Algorithm 2 and define a new length n for both Tr'_1 and Tr'_2.
Output: the cost of transforming Tr'_1 into Tr'_2 (Distance)
for $i = 0$ to n **do**
  $Cost_Matrix(i, 0) \leftarrow i * Gap_{Penalty}$
end for
for $j = 0$ to n **do**
  $Cost_Matrix(0, j) \leftarrow j * Gap_{Penalty}$
end for
for $i = 1$ to n **do**
  for $j = 1$ to n **do**
    $Match \leftarrow Cost_Matrix(i-1, j-1) +$
    $Score_Matrix(Tr'_1(i), Tr'_2(j))$
    $Delete \leftarrow Cost_Matrix(i-1, j) + Gap_{Penalty}$
    $Insert \leftarrow Cost_Matrix(i, j-1) + Gap_{Penalty}$
    $Cost_Matrix(i, j) =$
    $min(Match, Insert, Delete)$
  end for
end for
$Distance = Cost_Matrix(n, n)$
return $Distance$

Definition: Given a set of trajectories $\zeta' = \{Tr'_1, Tr'_2, ..., Tr'_N\}$, where i is the trajectory id, ($i =$ 1 to N), and N is the the number of trajectories. The *SLC* method is used together with *NWSA* to generate a set of distinct clusters $C = \{C_1, C_2, C_j, ..., C_M\}$, where j is the cluster id, ($j = 1$ to M), and M is the resulting number of clusters.

IV. Experiments

We have performed comparative experiments in order to evaluate the effectiveness of our methods, using three publicly available video datasets. These datasets represent different application domains, namely, vehicle traffic movement [4], human activities [15], and fish behaviors. We have evaluated both our classification and clustering methods:

- We first directly compare the performance of our classification method (described in Section III-C) against *state-of-the-art* quantitative methods (including [4]) using the traffic dataset. We then compare with the five methods presented in [15], using the dataset of human activities.
- We evaluate the performance of our unsupervised qualitative clustering method on all three datasets, and compare it with results obtained by clustering using the heatmap features presented by Lin et al. [4].

In this section we also introduce our fish activity dataset, FISHBEHAVE. We describe our bespoke labeling tool which we have created and used to facilitate expert annotation of this dataset. All experiments were run on an Intel Core i7 desktop, CPU@3.40GHz with 16.0GB RAM.



Fig. 5. Examples of the defined vehicle pair-activities [4].

Tr1	0 0 - +	0 0 - +	0 0 - +	0 0 - +	0 0 - +	0 0 - +	0 0 - +	- + - +	+ + - +	+ + - +	+ + - +	+ + - +	+ + - +	+ + - +	+ + - +	+ + - +	+ + - +	+ + - +	+ + - +	+ + - +	- - + -	Turn
Tr2	+ - + -	+ - 0 +	+ - + -	+ - 0 +	+ - 0 -	+ - 0 +	+ - + +	+ - - +	+ - + -	+ - - +	+ - 0 +	+ - + +	+ - - 0	+ - + +	+ - - -	+ - 0 0	+ - + +	+ - - 0	+ - + 0	+ - + -	Follow
Tr3	- 0 - 0	- 0 - 0	- 0 - 0	- 0 - 0	- 0 - 0	- 0 - 0	- 0 - 0	- 0 - 0	- 0 - 0	- 0 - 0	- 0 - 0	0 0 - 0	0 0 - 0	+ 0 - 0	+ 0 - 0	+ 0 - 0	+ 0 - 0	+ 0 - 0	+ 0 - 0	+ 0 - 0	+ 0 - 0	Pass
Tr4	- - 0 0	- - - -	- - - -	- - - -	- - - -	- - - -	- - - -	- - - -	- - - -	- - - -	+ + - -	+ + - -	+ + - -	+ + + 0	+ + - -	+ + - -	+ + - -	+ + - -	+ + - -	+ + - -	+ + - -	Both-Turn

Fig. 6. Examples of QTC_C trajectories for vehicle pair-activities Turn, Follow, Pass and BothTurn.

TABLE III
DEFINITION OF VEHICLES PAIR-ACTIVITIES.

Activity	Description
Turn	One car moves straight and another car in another lane turns right.
Follow	One car followed by another car on the same lane.
Pass	One car passes the crossroad and another car in the other direction waits for green light.
Bothturn	Two cars move in opposite directions and turn right at same time.

A. *Experiment* I*: Classification of Vehicle Activities*

The state of the art pair-activity classification method presented by Lin et al. [4] was shown to outperform a number of other methods ([7], [11], [13]), using the traffic dataset presented by the authors. We therefore use this algorithm, dataset, and ground truth, as a benchmark for evaluating our own classification method (Section III-C).

1) The Traffic Dataset: The traffic dataset was extracted from 20 surveillance videos. Four different pair-activities, {*Turn*, *Follow*, *Pass*, and *BothTurn*} are represented and corresponding annotations are provided. In total there are 35 clips for each activity. Each clip comprises exactly 20 frames. The dataset also includes x, y coordinates for the centroid of each vehicle in each frame, and a time-stamp t. Table III shows the definitions of the four vehicle pair-activities. Fig. 5 shows example frames from the dataset.

2) Results for Vehicle Activity Classification: We used the provided x, y coordinate pairs for each vehicle as inputs, and constructed corresponding QTC trajectories for each video clip. We repeated this for each QTC variant (QTC_B, QTC_C and QTC_{Full}). Fig. 6 shows sample trajectories of QTC_C, for four different interactions. Using each representation, we constructed corresponding score matrices for each pair of clips in the dataset, using Algorithm 1.

To determine the classification rates using our method (described in Section III-C), we used an iterative procedure. On each iteration, we randomly split the clips into training and test sets at a ratio of 75% to 25%, for each class. The training sets were used to parameterize the classifier for each class (that is, to determine the standard trajectories). The test data was then classified by comparison with the standard trajectories. This was repeated 5 times, using a randomly selected training set on each iteration, and the results were accumulated. Our results are shown in Table IV, which includes comparative results obtained by Lin et al. [4], and for the other algorithms [7], [11], [13] reported by Lin et al. The average error (AVG Error) is calculated as the total number of incorrect classifications (compared with the ground truth labeling) divided by the total number of activity sequences in the test set. Table IV further shows the results of our method using each of the three variants of QTC. Table IV shows that our method outperforms the other four, and is able to classify the dataset with no errors.

B. *Experiment* II*: Classification of Human Activities*

We have conducted similar classification experiments using the human activity dataset presented by Sethi et al. [15]. This comprises 90 video clips (20573 frames) of human pair-activities from various public sources. The authors have annotated each clip as either: *converging*, *diverging*, or *moving together*. Whilst there are only three classes, each captures significant diversity of behavior. Fig. 7 shows examples of the defined human pair-activities, and Table V shows their definitions. Additional data has been provided in the form of x, y coordinates of the centroid of each person in each frame, and corresponding time-stamps. We have used this dataset, algorithms, and ground truth, as a further benchmark for evaluating our own classification method, and have compared our results with the RDL, RVL, PS, K-Means and FUSION methods presented by the authors [15].

1) Results for the Human Activity Classification: We used the x, y coordinate pairs as inputs, and constructed corresponding QTC trajectories for each video clip. We repeated this for QTC_B, QTC_C and QTC_{Full}. To determine the correct classification rates using our method (Section III-C) we used 5-fold cross validation: results are shown in Table VI, which also includes comparative results for the other algorithms (RDL, RVL, PS, K-Means and FUSION) reported by Sethi et al. [15]. The average error (AVG Error) is calculated as the total number of incorrect classifications divided by the total

TABLE IV
MISS CLASSIFICATIONS ERROR FOR DIFFERENT ALGORITHMS ON THE TRAFFIC DATASET.

Type	QTC_B	QTC_C	QTC_{Full}	*Heat-Map*	*WF-SVM*	*LC-SVM*	*GRAD*
Turn	0.0%	0.0%	0.0%	2.9%	2.0%	16.9%	10.7%
Follow	0.0%	0.0%	0.0%	11.4 %	22.9%	38.1%	15.4%
Pass	0.0%	0.0%	0.0%	0.0%	11.7%	17.6%	15.5%
Bothturn	0.0%	0.0%	0.0%	2.9%	1.2%	2.9%	4.2%
AVG Error	**0.0%**	**0.0%**	**0.0%**	**4.3%**	**9.5%**	**18.9%**	**11.45**

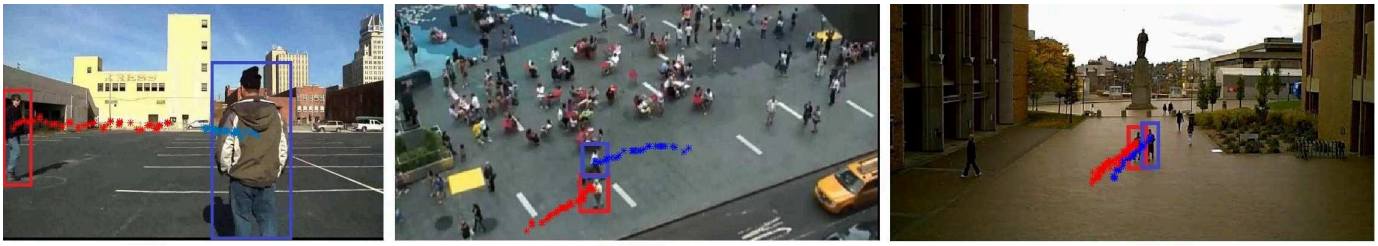


Fig. 7. Examples of the defined human pair-activities.

TABLE V
DEFINITION OF HUMAN PAIR-ACTIVITIES.

Activity	**Description**
Converge	One or both human exhibiting motion towards each other.
Diverge	One human moves away from the other human, or both human simultaneously move away from each other.
Parallel	Both human moving in the same direction.

number of activity sequences in the test set. Table VI further shows the results of our method using each of the three variants of QTC. It again shows that our method outperforms the five other methods, and is able to classify the data with errors rate varying between 3.0% and 4.1%. Whilst we are primarily interested in unsupervised clustering, the results reported in this section, and in Section IV-A clearly show the effectiveness of our classification method, when compared with other state-of-the-art methods, and validates the QTC representation and *NWSA* trajectory matching approach.

C. Experiment III: Clustering Vehicle Activities

We are primarily interested in the unsupervised clustering of pair-activities, and our second experimental evaluation applies our clustering method, described in Section III-D, to the traffic dataset. We also compare it with the analogous quantitative method (based on [4]) described in the same section. We again use the x, y coordinate pairs provided for each vehicle, and generate QTC trajectories for each pair-activity in each clip. We use all three QTC variants, and construct corresponding score matrices from the whole dataset using Algorithm 1. We then apply the *SLC* clustering method, to form activity clusters, in an unsupervised fashion.

We evaluate the quality of our clusters, and so compare them with corresponding clusters produced using the heatmap representation used by Lin et al. [4]. As described in Section III-D, we use the heatmap features in conjunction with the same *SLC* clustering method. Fig. 6 shows four sample QTC_C trajectories for four different vehicle interactions.

1) Results for Clustering Vehicle Activities: We use cluster *purity* as a quality metric, where the purity is determined from the ground truth annotations of the clustered activities. Cluster purity has previously been used by a number of authors as a measure of clustering quality and performance (such as [49], [50]). The class of each cluster is taken to be that of the activity class most commonly occurring within it: thus, the purity (for a set of clusters) is determined by dividing the number of activities whose ground-truth class corresponds to that of their associated clusters, by the total number of activities. Formally:

$$purity = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M}(Tr_i \in C_j) \wedge (G(Tr_i) = G(C_j)) \quad (4)$$

where $C = \{C_1, \ldots, C_i, \ldots, C_M\}$ is the set of clusters, $Tr = \{Tr_1, \ldots, Tr_i, \ldots, Tr_N\}$ is the set of trajectories, and G is a function which maps a class or trajectories to a ground truth category. Using cluster purity as a metric, the comparative results for clustering the traffic dataset are shown in Tables VII, VIII and IX for QTC_C, QTC_B and QTC_{Full} respectively. In each case there were exactly four clusters, which corresponded to the four classes of activity in the dataset. Table X shows the clustering results based heatmap method. The results show that the heatmap method produced 10 clusters, and achieved 92% purity, compared with 100% purity achieved using our method.

D. Experiment IV: Clustering Human Activities

We have repeated our experimental evaluation of unsupervised clustering, described in Section IV-C, using the human dataset introduced in Section IV-B [15]. Again, we compared the results with that obtained using the heatmap features [4], using cluster purity as a measure of cluster quality.

1) Results for Clustering Human Activities: The results of our method for clustering the human dataset are shown in Tables XI, XII and XIII. Using our method, we achieved cluster purities of 95.5%, 90.0% and 95.6% using the three QTC variants QTC_C, QTC_B and QTC_{Full}, respectively.

TABLE VI
MISS CLASSIFICATIONS ERROR FOR DIFFERENT ALGORITHMS ON THE HUMAN DATASET.

Type	QTC_B	QTC_C	QTC_{Full}	*RDL*	*RVL*	*PS*	*K-Means*	*FUSION*
Converge	1.3%	0.0%	0.0%	13.3%	86.7%	26.7%	0.0%	87.5%
Diverge	11.0%	11.0%	9.0%	0.0%	42.6%	0.0%	12.5%	0.0%
Parallel	0.0%	0.0%	0.0%	21.4%	14.3%	35.7%	0.0%	44.4%
AVG Error	**4.1%**	**3.7%**	**3.0%**	**11.6%**	**47.9%**	**20.8%**	**4.2%**	**44.0%**

TABLE VII
THE PURITY EVALUATION MEASURES OF OUR METHOD BASED QTC_C FEATURES OF TRAFFIC DATASET.

	Turn	Follow	Pass	BothT	Purity
C1	35	0	0	0	1
C2	0	35	0	0	1
C3	0	0	35	0	1
C4	0	0	0	35	1

TABLE VIII
THE PURITY EVALUATION MEASURES OF OUR METHOD BASED QTC_B FEATURES OF TRAFFIC DATASET.

	Turn	Follow	Pass	BothT	Purity
C1	0	0	0	35	1
C2	35	0	0	0	1
C3	0	0	35	0	1
C4	0	35	0	0	1

TABLE IX
THE PURITY EVALUATION MEASURES OF OUR METHOD BASED QTC_{Full} FEATURES OF TRAFFIC DATASET.

	Turn	Follow	Pass	BothT	Purity
C1	0	0	0	35	1
C2	35	0	0	0	1
C3	0	0	35	0	1
C4	0	35	0	0	1

TABLE X
THE PURITY EVALUATION MEASURES OF CLUSTERING BASED HEATMAP FEATURES OF TRAFFIC DATASET.

	Turn	Follow	Pass	BothT	Purity
C2	4	0	0	1	0.80
C3	0	0	0	6	1
C4	1	0	0	12	0.92
C5	0	0	0	16	1
C6	16	0	0	0	1
C7	2	11	0	0	0.85
C8	6	5	0	0	0.83
C9	5	2	0	0	0.71
C10	0	17	0	0	1

TABLE XI
THE PURITY EVALUATION MEASURES OF OUR METHOD BASED QTC_C FEATURES OF HUMAN DATASET.

	Converge	Diverge	Together	Purity
C1	0	9	0	1
C2	14	0	0	1
C3	0	0	4	1
C4	0	8	0	1
C5	0	0	12	1
C6	15	0	0	1
C7	1	3	14	0.78
C8	0	10	0	1

TABLE XII
THE PURITY EVALUATION MEASURES OF OUR METHOD BASED QTC_B FEATURES OF HUMAN DATASET.

	Converge	Diverge	Together	Purity
C1	0	12	0	1
C2	14	0	0	1
C3	0	0	5	1
C4	0	4	3	0.57
C5	0	1	8	0.89
C6	15	0	0	1
C7	1	4	14	0.74
C8	0	9	0	1

TABLE XIII
THE PURITY EVALUATION MEASURES OF OUR METHOD BASED QTC_{Full} FEATURES OF HUMAN DATASET.

	Converge	Diverge	Together	Purity
C1	0	10	0	1
C2	15	0	0	1
C3	15	0	0	1
C4	0	6	0	1
C5	0	1	17	0.94
C6	0	3	13	0.81
C7	0	10	0	1

TABLE XIV
THE PURITY EVALUATION MEASURES OF CLUSTERING BASED HEATMAP FEATURES OF HUMAN DATASET.

	Converge	Diverge	Together	Purity
C1	1	7	2	0.70
C2	0	1	5	0.71
C3	8	2	0	0.80
C4	1	2	3	0.50
C5	5	0	0	1
C6	0	6	1	0.86
C7	0	0	8	1
C8	1	10	0	1
C9	13	1	2	0.81
C10	1	1	9	0.81

Referring to Table XI, our method generates two distinct clusters for the *converge* class ($C2$ and $C6$). Inspection of these clusters reveals symmetric behaviors corresponding to the assigned indices of the tracked persons: one cluster corresponds to person 1 approaching person 2, and the other to person 2 approaching person 1. Clusters $C1$, $C4$ and $C8$ all map to the *diverge* class, and show similar symmetry. In clusters $C3$, $C5$ and $C7$ the two persons move together; in $C5$ person 1 predominantly moves in front of person 2, while in $C7$ the opposite is the case. Inspection of the $C7$ cluster is warranted, as it shows the lowest purity of all the emergent clusters. The converge and diverge clips in this cluster show some level of hybrid behavior: the two people move together, but also converge/diverge during the clips. We achieved similar

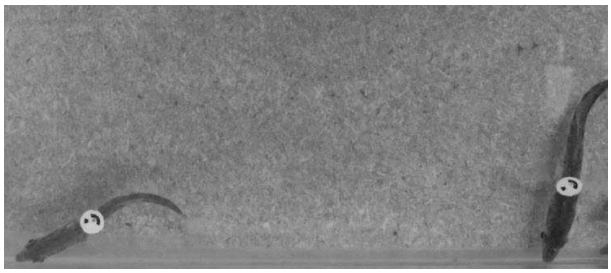


Fig. 8. A pair of three-spined sticklebacks with a unique circular tag attached to each individual.

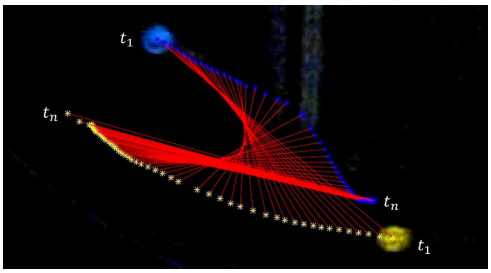


Fig. 9. Example of fish trajectory extraction between two fish.

TABLE XV
DEFINITION OF FISH PAIR-ACTIVITIES.

Activity	Description
Approach (A)	One fish moves towards the other fish, while the other fish is stationary.
Converge (C)	Both fish simultaneously move towards each other.
Follow (F)	One fish follows the other fish, maintaining an approximately equal distance and speed.
Diverge (D)	One fish moves away from the other fish, or both fish simultaneously move away from each other.
Stationary (S)	Neither fish exhibits any purposeful directional movement.

TABLE XVI
FISH PAIR-ACTIVITIES.

Behaviors	# of trajectories
$A \rightarrow D$	97
F	50
$A \rightarrow F$	34
$C \rightarrow D$	32
$F \rightarrow D$	22
$A \rightarrow S \rightarrow D$	14
$A \rightarrow D \rightarrow F$	10

results with QTC_B and QTC_{Full}, as shown in Tables XII and XIII.

Table XIV shows the clustering results based on the heatmap features. Using this method we derived 10 clusters, and achieved 82.2% purity, which is significantly lower. This is an important result as although there are only three classes in the annotations, the overall diversity in behavior is large, and so presents a greater challenge (as evidenced by the overall lower purity).

E. Experiment V*: Clustering Fish Behaviours*

As mentioned previously, our main motivation is the unsupervised analysis of fish behavior. We have constructed an extensive and expert-annotated dataset (FISHBEHAVE) of pair-behaviors for three-spined stickleback fish (*Gasterosteus aculeatus*), filmed in a laboratory environment. This species is often used by biological scientists for the study of social behavior. The FISHBEHAVE dataset is challenging, as individual clips comprise compound behaviors (that is, sequences of atomic behaviors), which are also captured by the corresponding annotations. This presents more diverse sets of behaviors, and therefore a more difficult clustering problem. We have repeated our evaluation on this dataset, and we present the results in this section. We first describe the FISHBEHAVE dataset, including details of the video collection, processing, and annotation method. We then describe our experimental procedure, which is similar to that used for the traffic and human datasets, followed by our results.

1) The FISHBEHAVE Dataset: The experimental setup comprised a black circular tank (30 cm diameter, 15 cm depth of water), in which pairs of fish were placed for filming. The fish were handled by appropriately trained staff and treated in accordance with the University of Lincoln ethical guidelines for work with live animals. For filming we used a Canon PowerShot SX200, recording in 1280 x 720p resolution, at 30 frames/second. A 30cm x 20cm calibration board containing 6 x 6 equally sized squares was used for the camera calibration process. Finally, all processing was performed off-line using an Intel Core i5-2450M laptop PC.

The fish were fitted with a circular marker disc for identification and tracking [43], and the camera was mounted directly above the tank: Fig. 8 shows an example image. We used seven pairs of fish, and recorded a total of 10 hours of interactions (1080000 frames), over a period of several days. The data is downloadable as either .mp4 videos (approximately 80GB in total), as 1080000 individual frames, or as x, y coordinates pairs, from [51].

Our method, described in Section III-A, was used to identify and track the positions of the fish. The video was automatically segmented into clips which represent interactions, based on the concept of Elective Group Size [52], which is used by biologists to determine when interactions are occurring. In recent work [53], interactions were assumed where separation was less than two body lengths (the approximate body length of this species is 3.5cm). We therefore automatically segmented our dataset into clips on this basis. We extracted 322 such clips, and Figure 9 shows an example of a frame (enhanced with trajectories). The clips vary in duration from 0.5 to 6 seconds.

2) Behavior Annotations and Labeling Tool: Biologists studying this species identified five atomic pair-wise interactions, namely: {Follow, Converge, Diverge, Approach, and Stationary}. These are elucidated in Table XV. Each segmented clip may contain multiple atomic behaviors, and so we developed a tool to allow expert ground-truth annotation of each clip (we later use these annotations to measure cluster purity). Three biologists (experts in fish behaviors, O_1, O_2 and O_3) independently observed and annotated the 322 clips, using this tool. They also second-labeled a randomly chosen subset of 30 video clips, to check consistency. Observer O_1 (the

TABLE XVII
THE PURITY EVALUATION MEASURES OF OUR METHOD BASED QTC_C FEATURES OF FISHBEHAVE DATASET.

	A,D	F	A,F	C,D	F,D	A,S,D	A,D,F	Purity
C1	0	0	0	13	0	0	0	1
C2	10	0	0	0	0	0	0	1
C3	1	0	0	0	0	9	0	0.90
C4	0	6	0	0	0	0	0	1
C5	0	1	0	0	10	0	0	0.91
C6	0	0	1	0	0	0	10	0.91
C7	15	0	1	1	0	4	0	0.71
C8	0	27	2	0	0	0	0	0.93
C9	1	14	0	0	0	0	0	0.93
C10	0	0	14	0	0	0	0	1
C11	15	0	0	0	0	0	0	1
C12	0	0	0	0	12	0	0	1
C13	0	0	16	0	0	0	0	1
C14	55	2	0	0	0	0	0	0.96
C15	0	0	0	18	0	1	0	0.95

most experienced researcher) achieved 100% self-agreement while O_2 and O_3 each achieved 96%. All observers gave the same labeling in 95% of cases, which we took to be the *de facto* ground-truth labeling for those clips. In the remaining cases, we used a voting system: if two observers agreed, then we took that to be the correct labeling, and if all three gave different labelings, we took that given by O_1 to be correct. The result was a total of 50 unique labelings, with seven commonly occurring (comprising 80% of the dataset), and 43 infrequently occurring. Table XVI shows the seven commonly occurring behaviors.

3) Results for Clustering Fish Behaviours: We used our image processing method described in Section III-A to extract the positions of the fish, resulting in x, y coordinate pairs for each frame (image frame coordinates of the tag centroids). A detailed description of the processing method is also given in our previous work [44]. The coordinates were used to construct a QTC trajectory for each clip. We again used all three variations: QTC_B, QTC_C and QTC_{Full}, and the same processing and clustering methods used for the traffic and human datasets in Sections IV-C and IV-D. Again, we compared with the heatmap features used by Lin et al. [4], using cluster purity as a metric. As mentioned, 80% of the sequences correspond to seven commonly occurring ground truth labelings. Our method, for all three QTC variants, generated 15 primary clusters corresponding directly to this labeling subset. Tables XVII, XVIII and XIX give details of the cluster compositions, and we achieved purities of 93.8%, 90.1% and 93.9% using QTC_C, QTC_B and QTC_{Full}, respectively, across these primary clusters. Note that QTC_C and QTC_{Full} produced better results than QTC_B.

The results obtained using the heatmap features are shown in Table XX. Note that there were a larger number of primary clusters in this case, and that the purity of these clusters is significantly lower at 80.5%. The infrequently occurring trajectories were detected, and included in the purity estimations in the results for both the QTC and heatmap methods (some were located within the primary clusters presented in the Tables XVII, XVIII, XIX and XX).

TABLE XVIII
THE PURITY EVALUATION MEASURES OF OUR METHOD BASED QTC_B FEATURES OF FISHBEHAVE DATASET.

	A,D	F	A,F	C,D	F,D	A,S,D	A,D,F	Purity
C1	1	0	0	17	0	0	0	0.94
C2	15	0	0	2	0	0	0	0.88
C3	3	0	0	0	0	10	0	0.77
C4	9	0	1	0	0	0	0	0.90
C5	0	7	0	0	0	0	0	1
C6	0	1	0	0	10	0	0	0.91
C7	1	0	0	0	0	0	9	0.90
C8	0	0	0	0	0	4	0	1
C9	2	28	2	0	0	0	0	0.88
C10	1	12	0	0	0	0	0	0.92
C11	4	1	15	0	0	0	0	0.75
C12	4	0	0	0	12	0	0	0.75
C13	0	1	16	0	0	0	0	0.94
C14	57	0	0	0	0	0	1	0.98
C15	0	0	0	13	0	0	0	1

TABLE XIX
THE PURITY EVALUATION MEASURES OF OUR METHOD BASED QTC_{Full} FEATURES OF FISHBEHAVE DATASET.

	A,D	F	A,F	C,D	F,D	A,S,D	A,D,F	Purity
C1	1	0	0	13	0	0	1	0.87
C2	11	0	0	1	0	0	0	0.92
C3	1	0	0	0	0	13	0	0.93
C4	0	6	0	0	0	0	0	1
C5	0	1	0	17	0	0	0	0.94
C6	16	0	0	1	0	0	0	0.94
C7	12	0	0	0	0	0	0	1
C8	1	15	2	0	0	0	0	0.83
C9	0	0	11	0	0	0	0	1
C10	0	0	0	0	8	0	0	1
C11	0	26	0	0	1	0	0	0.96
C12	0	0	0	0	12	0	0	1
C13	55	0	0	0	1	1	0	0.96
C14	0	0	21	0	0	0	0	1
C15	0	2	0	0	0	0	9	0.81

TABLE XX
THE PURITY EVALUATION MEASURES OF CLUSTERING BASED HEATMAP FEATURES OF FISHBEHAVE DATASET.

	A,D	F	A,F	C,D	F,D	A,S,D	A,D,F	Purity
C1	0	10	0	0	1	0	0	0.91
C2	0	1	8	0	0	0	0	0.89
C3	7	1	1	0	1	0	0	0.70
C4	13	1	1	0	0	1	0	0.81
C5	11	0	0	1	0	0	0	0.92
C6	1	0	5	0	0	0	0	0.83
C7	1	0	0	4	0	0	0	0.80
C8	0	0	0	0	6	0	1	0.86
C9	0	0	0	0	1	7	0	0.88
C10	2	0	0	0	0	0	7	0.78
C11	9	0	1	2	0	1	1	0.64
C12	0	10	1	0	0	0	0	0.91
C13	1	0	0	14	0	1	0	0.88
C14	15	2	2	0	0	2	0	0.71
C15	27	2	2	0	2	1	0	0.79
C16	2	20	0	0	1	1	1	0.80
C17	8	1	13	0	0	0	0	0.59
C18	0	2	0	0	9	0	0	0.81
C19	0	0	0	11	1	0	0	0.92

V. Conclusion and Discussion

Qualitative methods of describing pair-wise interactions use symbolic representations of relative geometric and motion data, and have attracted interest from computer vision researchers due to their intuitive and naturalistic representation. So far, work has been limited to a small number of specific cases: no general method has been investigated, nor have any comparisons been made with more commonly used methods which use quantitative features.

In this paper we have presented a general qualitative method for analyzing pair-activities, applicable to both classification and clustering problems. Our method uses the QTC representation [17], and we have constructed a metric for sequence similarity (*NWSA*), which, as we have shown, can be successfully used with different types of data. The demonstrated generality of *NWSA*, and of our classification and clustering methods, are an important contribution: we believe that this will enable the application of qualitative analysis to a variety of applications in future.

We have conducted a number of direct comparisons, with different datasets, against the *state-of-the-art* quantitative method proposed by Lin et al. [4], which has itself been shown to outperform other recent methods ([7], [11], [14]). We have shown that our classification method outperforms that developed by Lin et al. [4]: for the classification of traffic data, we achieved a zero error rate, compared to a between 4.3% and 18.9% reported by Lin et al. Similarly, for the human activity dataset, our method outperformed the five methods applied by Sethi et al. [15], achieving errors rates of only 3.0% to 4.1%.

Using our clustering method, we achieved between 8% and 14% higher cluster purity than those obtained using the features proposed by Lin et al. [4], across all three datasets, and with all versions of QTC. This again demonstrates the effectiveness of our method. We have also presented our FISHBEHAVE dataset, which provides a detailed and useful resource for researchers studying animal behavior, and is publicly available for download.

Encouraged by our results, we plan to extend our work in a number of ways. Firstly, we wish to further support ongoing research into fish behavior by analyzing further datasets collected from biological experiments. To support this, we will develop this method to analyze differences in behavior between pairs of fish, over long-run datasets. For comparative purposes, we have used the SLC clustering method, and also the classification method proposed by Lin et al. [4], in our experiments. We wish to investigate other clustering and classification techniques in our future work. We also wish to further investigate other methods of measuring the similarity between sequences of QTC states, which will include comparison of *NWSA* with Dynamic Time Warping. Finally, our tracking method described in [44] is able to extract 3-dimensional orientation data. We therefore wish to develop a new QTC representation which includes 3-dimensional features, and use this to analyze fish behavior in more detail.

References

[1] N. Shoval and M. Isaacson, "Sequence alignment as a method for human activity analysis in space and time," *Annals of the Association of American geographers*, vol. 97, no. 2, pp. 282–297, 2007.

[2] S. Dodge, P. Laube, and R. Weibel, "Movement similarity assessment using symbolic representation of trajectories," *International Journal of Geographical Information Science*, vol. 26, no. 9, pp. 1563–1588, 2012.

[3] H. Xu, Y. Zhou, W. Lin, and H. Zha, "Unsupervised trajectory clustering via adaptive multi-kernel-based shrinkage," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4328–4336.

[4] W. Lin, H. Chu, J. Wu, B. Sheng, and Z. Chen, "A heat-map-based algorithm for recognizing group activities in videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 11, pp. 1980–1992, 2013.

[5] Y. Yin, G. Yang, J. Xu, and H. Man, "Small group human activity recognition," in *2012 19th IEEE International Conference on Image Processing*. IEEE, 2012, pp. 2709–2712.

[6] P. Laube, S. Imfeld, and R. Weibel, "Discovering relative motion patterns in groups of moving point objects," *International Journal of Geographical Information Science*, vol. 19, no. 6, pp. 639–668, 2005.

[7] B. Ni, S. Yan, and A. Kassim, "Recognizing human group activities with localized causalities," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1470–1477.

[8] Y. Yin, G. Yang, and H. Man, "Small human group detection and event representation based on cognitive semantics," in *Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on*. IEEE, 2013, pp. 64–69.

[9] Y.-J. Kim, N.-G. Cho, and S.-W. Lee, "Group activity recognition with group interaction zone," in *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE, 2014, pp. 3517–3521.

[10] W. Lin, Y. Chen, J. Wu, H. Wang, B. Sheng, and H. Li, "A new network-based algorithm for human activity recognition in videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 5, pp. 826–841, 2014.

[11] W. Lin, M.-T. Sun, R. Poovendran, and Z. Zhang, "Group event detection with a varying number of group members for video surveillance," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 8, pp. 1057–1067, 2010.

[12] R. Sethi, "Towards defining groups and crowds in video using the atomic group actions dataset," in *2015 IEEE International Conference on Image Processing (ICIP)*, Sept 2015, pp. 2925–2929.

[13] Y. Zhou, S. Yan, and T. S. Huang, "Pair-activity classification by bi-trajectories analysis," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[14] Y. Zhou, B. Ni, S. Yan, and T. S. Huang, "Recognizing pair-activities by causality analysis," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 1, p. 5, 2011.

[15] R. J. Sethi, H. Jo, and Y. Gil, "Structured analysis of the isi atomic pair actions dataset using workflows," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 2023–2032, 2013.

[16] M. Hanheide, A. Peters, and N. Bellotto, "Analysis of human-robot spatial behaviour applying a qualitative trajectory calculus," in *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2012, pp. 689–694.

[17] N. Van de Weghe, "Representing and reasoning about moving objects: A qualitative approach," Ph.D. dissertation, Ghent University, 2004.

[18] S. H. Chavoshi, B. De Baets, T. Neutens, G. De Tré, and N. Van de Weghe, "Exploring dance movement data using sequence alignment methods," *PloS one*, vol. 10, no. 7, p. e0132452, 2015.

[19] M. Delafontaine, A. G. Cohn, and N. Van de Weghe, "Implementing a qualitative calculus to analyse moving point objects," *Expert Systems with Applications*, vol. 38, no. 5, pp. 5187–5196, 2011.

[20] P. Bogaert, N. Van de Weghe, A. G. Cohn, F. Witlox, and P. De Maeyer, "The qualitative trajectory calculus on networks," in *International Conference on Spatial Cognition*. Springer, 2006, pp. 20–38.

[21] C. Freksa, *Using orientation information for qualitative spatial reasoning*. Springer, 1992.

[22] J. Z. Li, M. T. Ozsu, and D. Szafron, "Modeling of moving objects in a video database," in *Multimedia Computing and Systems' 97. Proceedings., IEEE International Conference on*. IEEE, 1997, pp. 336–343.

[23] A. U. Frank, "Qualitative spatial reasoning: Cardinal directions as an example," *International Journal of Geographical Information Science*, vol. 10, no. 3, pp. 269–290, 1996.

[24] K. Rieck, "Similarity measures for sequential data," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 4, pp. 296–304, 2011.

[25] S. H. Chavoshi, B. De Baets, T. Neutens, H. Ban, O. Ahlqvist, G. De Tré, and N. Van de Weghe, "Knowledge discovery in choreographic data

using relative motion matrices and dynamic time warping," *Applied Geography*, vol. 47, pp. 111–124, 2014.
[26] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473–1488, Nov 2008.
[27] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *IEEE transactions on pattern analysis and machine intelligence*, vol. 30, no. 2, pp. 283–298, 2008.
[28] X. Wang, K. T. Ma, G.-W. Ng, and W. E. L. Grimson, "Trajectory analysis and semantic region modeling using a nonparametric bayesian model," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.
[29] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.
[30] J. Shawe-Taylor and N. Cristianini, *Kernel methods for pattern analysis*. Cambridge university press, 2004.
[31] S. Sonnenburg, G. Rätsch, and K. Rieck, "Large scale learning with string kernels," *Large Scale Kernel Machines*, pp. 73–103, 2007.
[32] R. W. Hamming, "Error detecting and error correcting codes," *Bell System technical journal*, vol. 29, no. 2, pp. 147–160, 1950.
[33] A. Singhal, "Modern information retrieval: A brief overview," *IEEE Data Eng. Bull.*, vol. 24, no. 4, pp. 35–43, 2001.
[34] W. M. David, "Bioinformatics: sequence and genome analysis," *Bioinformatics*, vol. 28, 2001.
[35] A. Abbott, "Sequence analysis: new methods for old ideas," *Annual review of sociology*, pp. 93–113, 1995.
[36] N. Shoval and M. Isaacson, "Tracking tourists in the digital age," *Annals of Tourism Research*, vol. 34, no. 1, pp. 141–159, 2007.
[37] M. Delafontaine, M. Versichele, T. Neutens, and N. Van de Weghe, "Analysing spatiotemporal sequences in bluetooth tracking data," *Applied Geography*, vol. 34, pp. 659–668, 2012.
[38] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of molecular biology*, vol. 48, no. 3, pp. 443–453, 1970.
[39] M. S. Rosenberg, *Sequence alignment: methods, models, concepts, and strategies*. Univ of California Press, 2009.
[40] A. Galton, "Dominance diagrams: A tool for qualitative reasoning about continuous systems," *Fundamenta Informaticae*, vol. 46, no. 1-2, pp. 55–70, 2001.
[41] C. A. Ratanamahatana and E. Keogh, "Everything you know about dynamic time warping is wrong," in *Third Workshop on Mining Temporal and Sequential Data*, 2004.
[42] M. Grachten, M. Gasser, A. Arzt, and G. Widmer, "Automatic alignment of music performances with structural differences," in *In Proceedings of 14th International Society for Music Information Retrieval Conference (ISMIR*, 2013.
[43] T. Kleinhappel, A. Al-Zoubi, B. Al-Diri, O. Burman, P. Dickinson, L. John, A. Wilkinson, and T. Pike, "A method for the automated long-term monitoring of three-spined stickleback gasterosteus aculeatus shoal dynamics," *Journal of fish biology*, vol. 84, no. 4, pp. 1228–1233, 2014.
[44] A. AlZoubi, T. K. Kleinhappel, T. W. Pike, B. Al-Diri, and P. Dickinson, "Solving orientation duality for 3d circular features using monocular vision," in *Proceedings of the 10th International Conference on Computer Vision Theory and Applications (VISIGRAPP 2015)*, 2015, pp. 213–219.
[45] N. Van de Weghe and P. De Maeyer, "Conceptual neighbourhood diagrams for representing moving objects," in *International Conference on Conceptual Modeling*. Springer, 2005, pp. 228–238.
[46] G. Foster, "Time series analysis by projection. i. statistical properties of fourier analysis," *The Astronomical Journal*, vol. 111, p. 541, 1996.
[47] T. W. Liao, "Clustering of time series data a survey," *Pattern recognition*, vol. 38, no. 11, pp. 1857–1874, 2005.
[48] P. Langfeldera, B. Zhangb, and S. Horvatha, "Dynamic tree cut: in-depth description, tests and applications," *November*, vol. 22, p. 2007, 2007.
[49] A. Cherian, V. Morellas, and N. Papanikolopoulos, "Bayesian nonparametric clustering for positive definite matrices," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 5, pp. 862–874, 2016.
[50] D. Phung, B. Adams, K. Tran, S. Venkatesh, and M. Kumar, "High accuracy context recovery using clustering mechanisms," in *Pervasive Computing and Communications, 2009. PerCom 2009. IEEE International Conference on*. IEEE, 2009, pp. 1–9.
[51] A. AlZoubi. (2017) FISHBEHAVE. [Online]. Available: http://3dfishtrack.com/wordpress/dataset
[52] T. Pitcher, A. Magurran, and J. Allan, "Shifts of behaviour with shoal size in cyprinids," in *Proceedings of the British Freshwater Fisheries Conference*, vol. 3, 1983, pp. 220–228.
[53] T. K. Kleinhappel, O. H. Burman, E. A. John, A. Wilkinson, and T. W. Pike, "Diet-mediated social networks in shoaling fish," *Behavioral Ecology*, vol. 25, no. 2, pp. 374–377, 2014.



Alaa AlZoubi is a Research Fellow in Control, Guidance and Navigation at Cranfield University. He received his BSc from the Al-albayt University, MSc from the University of Jordan, and his PhD from the University of Lincoln in 2017. His research interests are in computer vision, especially 3D object recognition and activity analysis, autonomous system, qualitative reasoning, and Geographic Information Systems (GIS). Alaa has more than 8 years of industrial experience, in computer vision R&D, and GIS as a senior software engineer.



Bashir Al-Diri is a Senior Lecturer at the University of Lincoln, and member of the Laboratory of Vision Engineering in the School of Computer Science. Dr. Al-Diri developed a robust fully automated system for retinal vascular segmentation and measurement. His research includes computer vision, medical image analysis, automated surveillance, artificial intelligence, speech recognition and language corpus and lexical analysis.



Tom Pike obtained his PhD from the University of Newcastle in 2004. He then went on to postdoctoral positions at the universities of Glasgow, St Andrews and Exeter, before taking up his current position as Senior Lecturer at the University of Lincoln in 2011. His research focuses on animal sensory and behavioral ecology, with an emphasis on the sensory underpinnings of social behaviors in fish.



Tanja Kleinhappel obtained her BSc from the University of Graz in 2009 and her MSc from the University of Vienna in 2012, both in animal behavior science. She completed her PhD at the University of Lincoln in 2015, which looked at the causes and consequences of social network structure within animal groups. Dr. Kleinhappel is a member of the Association for the Study of Animal Beahaviour (ASAB), Ethologische Gesellschaft e.V. and the International Society of Behavioral Ecology (ISBE).



Patrick Dickinson is a Principal Lecturer at the University of Lincoln, and member of the Interactions Laboratory in the School of Computer Science. He received his BSc from the University of Southampton, MSc from the University of Oxford, and his PhD from the University of Lincoln in 2008. His primary research interests are in computer vision, especially applied to animal behavior, computer games, and mobile platform services.



https://dspace.org/

Pair-Activity Analysis from Video Using Qualitative Trajectory Calculus

Alzoubi, A; Al-Diri, B; Pike, T; Kleinhappel, T; Dickinson, P

2017-05-08
IEEE
https://doi.org/10.1109/TCSVT.2017.2701860
AlZoubi A, Al-Diri B, Pike T, Kleinhappel T, et al., (2018) Pair-activity analysis from video using Qualitative Trajectory Calculus. IEEE Transactions on Circuits and Systems for Video Technology, Volume 28, Issue 8, August 2018, pp. 1850-1863

Downloaded from a DSpace repository. https://dspace.org/