To Ana

TABLE OF CONTENTS

AKNOWLEGMENTS

I wish to thank my supervisor, Michael Orger, for accepting me on his lab on a very difficult time. Besides, he always incited me to follow my path, in spite of being unclear, most the time, where I was going. Also, he solved all the really hard problems.

I wish to thank, Rita Felix, Sabine Renninger, Simone Lackner, and Antonia Groneberg, for having the patience to cope with my difficult personality.

I would like to acknowledge the privilege of being part of the 2008 INDP class. It was a great opportunity to learn about neuroscience with: Mafalda Vicente, Fernando Santos, Ana Amaral, Rita Fonceca, Ana Pereira, Clara Ferreira, Dennis Hermann, Scott Rennie, and Andre Mendonça. The really important insights always come from people who think differently and this was certainly the case of my PhD class.

I wish to thank everyone at the IGC and the CCU. Two amazing places to do science next to the ocean, with amazing surf.

I wish to thank my parents for letting me run lose all these years.

I wish to give special thanks to Ana Pereira for giving me the motivation to write this thesis and for being always present even when far away.

ABREVIATION LIST

AS	approach swim
BS	burst swim
CPG	central pattern generator
dpf	days post fertilization
HAT	high angle turn
LLC	long latency C-start
MST	minimal spanning tree
nMLF	medial longitudinal fasciculus
OMR	optomotor response
PC	principal component
RS	reticulo-spinal
RT	routine turn
ROC	receiver operating characteristic
SI	separability index
SAR	spot avoidance response
TBF	tail beat frequency
t-SNE	t-distributed stochastic neighbour embedding

RESUMO

A principal função do sistema nervoso é produzir comportamentos que contribuem para a sobrevivência dos animais. Deste modo, se queremos compreender como o sistema nervoso funciona é fundamental termos isto em conta. Foi proposto que os comportamentos são formados por sequências de unidades de movimento mais simples. A identificação e caracterização destes módulos de comportamento poderá ser útil para compreender o funcionamento dos circuitos neuronais que os produzem.

Começamos, no capítulo 1, por discutir as vantagens e desvantagens de utilizar métodos de classificação de comportamentos supervisionados por humanos ou completamente automáticos. Também sublinhamos que ambas os tipos de métodos podem produzir soluções drasticamente diferentes.

No capítulo 2 utilizamos um método de classificação que se baseia em parâmetros cinemáticos de movimentos e descrevemos os tipos de movimentos que larvas de peixe zebra utilizam para nadar a diferentes velocidades. Confirmamos que, como outros vertebrados, as larvas de peixe zebra possuem dois modos de locomoção.

O terceiro capítulo consiste em comunicar um novo algoritmo de agrupamento de dados baseado nos vales que existem entre picos de densidade. Este algoritmo é capaz de detector automaticamente o número de grupos que existem em uma grande variedade de dados sintéticos e reais.

No capítulo 4 descrevemos um sistema automático que mede os movimentos de peixes e como utilizámos este sistema para adquirir uma coleção de movimentos para múltiplos comportamentos. No capítulo 5 utilizamos o algoritmo de agrupamento de dados que descrevemos anteriormente à nossa coleção de movimentos e classificamos os tipos de movimentos que as larvas efectuam durante vários comportamentos. Finalmente, no capítulo 6, utilizamos esta classificação para descrever as sequências de movimentos que od peixes zebra produzem em resposta a diferentes estímulos sensoriais. Também, utilizamos as transições entre diferentes movimentos para criar

um espaço onde sequências de movimentos formam grupos que correspondem a distintos estados comportamentais.

ABSTRACT

The ultimate function of the nervous system is to produce behaviours that enable animals to survive and reproduce. Accordingly, if we want to understand how the brain works it is crucial to take into account its final output. Ethology postulates that complex behaviours are formed by sequences of simpler units of movements. The identification and characterization of these behavioural motifs could be useful to understand how the neuronal circuits that underlay them work.

We begin, in chapter 1, by discussing the advantages and disadvantages that exist in using supervised and unsupervised methods to classify behavioural motifs, and the radically different solutions that both approaches can produce. In chapter 2 we use a simple supervised method based on kinematic parameters of bouts to classify the movement types that zebrafish larvae execute while moving at different speeds, and confirm that, like other vertebrates, larvae use two gaits to control speed. The third chapter reports a new general purpose clustering algorithm based on the valley between density peaks that is able to automatically detect the number of clusters on a wide variety of synthetic and real live data sets. In chapter 4 we describe a novel automatic tracking system and how we used it to acquire a large collection of movements that zebrafish larva preformed while engaged in a wide range of behaviours. In chapter 5 we apply density valley clustering to this collection of larval movements and design an approach to compare similarity between clusters of movements. We found, free from human supervision, eleven bout types, seven of which had been described previously and four that are novel. Finally, in chapter 6, we use this classification to describe the sequences of bouts that larvae form while responding to continuous stimuli. By using the transitions between bout types we create a space where the sequences that fish execute in response to particular stimuli cluster into behavioral states.

CHAPTER 1 – General Introduction.

The Brain Can Only Be Understood In The Light of Behaviour

Understanding how the nervous system works is a daunting challenge. Its functions are numerous and complex: sensory perception, learning and memory, cognition, emotion and motor control. But in all its complexity the nervous system serves the single purpose of enabling animals to interact with the world by producing adaptive behaviours. To perform the appropriate behaviour at the appropriate time will often decide if an animal lives or dies, so it is likely that natural selection acts on behaviour and its components. Consequently, it may be useful to consider behaviour to understand how the brain works (1).

In recent years we have witnessed a technological revolution in neuroscience. It is based on targeting proteins that enable the monitoring, mapping, and manipulation of neuronal activity to genetically identified neuronal types (for an in-depth review see (2)). This toolbox of new technologies promises to unravel causal links between functional modules of neuronal circuits and its ultimate function, the production of behaviour. However, to understand the results of these experiments there is the need to use a clear framework that explains how behaviour is organized and is modulated with experimental manipulations.

One conceptual idea that may help resolve this issue arises from classical ethology, which defends that the brain produces meaningful behaviours by organizing simpler and stereotypical types of movements (also called, behavioural motifs (3), behavioural units, motor actions (1), behavioural modules (4) or movement primitives (5)) into sequences (6). It is possible that behavioural motifs are produced by specific neuronal circuits or activity states and that behaviour can be explained at the neuronal level by the transitions between those circuits (7). However, there is no consensus on what constitutes a behaviour motif or even a general agreement that they exist.

How to Detect Behavioural Motifs?

In classical ethology behavioural motifs have been identified by careful human observation and its descriptions were primarily qualitative in nature. Over the years their detection has become increasingly more quantitative, but for the most part performed by humans classifying animal movements according to some general accepted criteria. While initially such quantifications were performed manually in the field, using a pencil and a stopwatch, soon after computer programmes were developed that enabled the *in silico* annotation of videotaped behaviour. In these cases, humans classify the frames where the animal is performing the generally accepted types of movements (e.g. see (8)).

Although this frame by frame annotation enables correction and classification by multiple subjects, it is extremely time consuming and unreliable. To overcome these problems, machine learning algorithms have been developed that learn classifiers from human annotated videos and automatically classify the rest of the data (e.g. see (9) and (10)). The drawback of such supervised approaches is that, since they need to be trained by data classified by humans, they are always bound by human perception and intuition. Furthermore, they are unable to detect new movement types that were not present in the initial data set that was used to train the algorithm.

Another approach to classify behaviour consists in using unsupervised algorithms, that rely on statistical criteria and develop their own set of classifiers by which to decompose the animals' behaviour into units. Initial efforts have been pioneered for *C. elegans* (11–13), but now have been extended to other model organisms as *Drosophila melanogaster* (14, 15), zebrafish larvae (16) and mouse (4). Unsupervised methods have the advantage of being independent of human preconceptions and mistakes, and of being applicable to large data sets that are challenging to visualize directly.

However, the unsupervised classification approach has not clearly brought us closer to original behavioural motifs that early ethologists postulated to be at the core of all behaviours. Frequently, the solutions obtained by unsupervised methods have turned

out to be quite different from the original motifs found by ethologist's careful observations (6).

Let's take as an example the classic study by Berridge *et al* of rat's behavioural sequences (8). The authors "manually" classified rat's movements and come up with twelve behavioural motifs. These movement types are straightforward and one immediately understands what the rat is doing (e.g. "face washing", "paw licking" and "head shakes"). In other words, classical ethology methods rely heavily in human intuition and culture. Using these behavioural motifs, the authors use sequence analysis and find specific rules that govern movement sequences for several behaviours.

If we look at another study with rodents, but where an unsupervised approach was used, the solution is radically different (4). In the first place, the behavioural motifs are much shorter (between 10ms and 1s). This means that they are closer to behavioural postures or movemes (the simplest meaningful movement pattern that one can identify, e.g. "step" (1)) than to actions or movements, making it difficult to attach a function to any of them. Secondly, there are many. In the case of Wiltschko et al., there are 65 behavioural modules for a mouse in a box that is doing very little. Thirdly, the method relies on choosing from a library of models the one that explains better the data, thus it is possible that changes in the conditions under which the data is acquired, and how it is processed will have a profound influence on the final set of behavioural motifs that are found. Nevertheless, the authors were able to show that behavioural, genetic and optogenetic manipulations change the frequency and transition probabilities between modules, implying that the categories that were found provide useful descriptions of behaviour. Berman and colleagues utilized a distinct, unsupervised method to categorize the movements that drosophila use when walking. Their method found over one hundred behavioural motifs that are also difficult to relate to previously reported ethological motifs of the fly (14).

The question remains of which, if either, of these two contrasting approaches captures the biological phenomenon of behaviour in a meaningful way. Is it possible that the origin of the classical ethological "motifs" is rooted in the cultural perspectives of the human scientists and not on underlying biological mechanisms? On the other hand, do the solutions of unsupervised methods more accurately reflect the underlying biology, or could they also be shaped by the particular methods used to discover them?

The Temporal Scale Problem of Behaviour

Behavioural phenomena take place in multiple time scales ranging from simple movements occurring in fractions of seconds (a saccade, a blink), to more elaborate patterns lasting for minutes (a person jogging, a fly flying) or even hours and days (dominance) (1). Most animals, from *C. elegans*, to the rat, behave in a continuous way, making it difficult to isolate single movements and bridge multiple temporal scales. If the classical ethology idea of behaviour being composed of stereotypical movement units is true these behavioural motifs are fused together in most animals and are displayed in a continuous way. The trained eye of the ethologist, backed with the knowledge given by the field, is able to move between temporal scales and separate these behavioural units, a feat that is hard for any unsupervised algorithm to do. These algorithms focus on particular time scales and will not be sensitive to patterns of behaviour that happen on other time scales.

Ideally, one would like to have an approach to find behavioural motifs that could identify previously unknown categories of movements and be free from human intervention. But, for it to be useful, the ideal approach should also generate results that are robust to changes in the algorithm or the data set used, and identify movements or actions which can be meaningfully connected to the biology or ethology of the animal.

The zebrafish larva is close to an ideal model system to investigate these questions about the structure of behaviour. First the motor output of the larvae is very simple to track, consisting largely of curvature of the tail in the horizontal plane, rotations of the eyes and movements of the two pectoral fins. Although they are able to move in three dimensions, the larvae can be confined to areas of shallow water, effectively reducing the bulk of their behaviour to a two dimensional plane which can be recorded with a single camera view. Also, it is a genetically tractable organism that is transparent, being possible to record the activity of all the neurons of the brain in seconds (17) and to manipulate genetically defined populations of neurons using light (18). Thus, behavioural motifs can be correlated with recordings and changes in neural circuits. Finally, contrary to other commonly used genetic model organisms, the movements are organized in a highly discrete fashion which permits the identification of three non-overlapping time scales of organization in behaviour.

Larva's Behaviour is Discrete

The zebrafish larvae use an intermittent 'burst and glide' style of locomotion where tail movements (called swim bouts) are always followed by a period of tail stasis (interbouts). In Figure 1.1 we are plotting, as a marker of movement, the angle of eight points of the larva's tail versus time. Marked with numbers are the swim bouts that together form a sequence of five bouts. The swim bouts last hundreds of milliseconds and are composed of shorter movements (called half beats) that have a duration in the order of the tens of milliseconds (left panel of Figure 1.1). The larvae organize the swim bouts in arbitrary long sequences (seconds to hours), that may enable them to interact with the environment in complex and diverse ways.

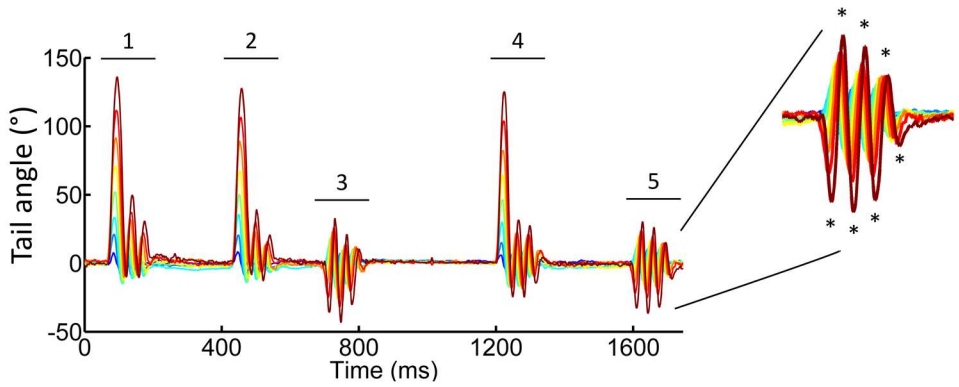

Figure 1.1. Zebrafish larvae behaviour is organized in discrete units of movement. Tail angle (°) versus time (ms). Colours represent eight points tracked on the tail; blue rostral and red is caudal. The numbers mark the order of bouts that fish performed in the bout sequence. Right panel is the expansion of the last bout. The bouts are composed of sequences of half beats (asterisks).

Due to the discreetness of the larvae's behaviour there is a rich tradition of ethological studies that is based on identifying kinematically distinct categories of swim bouts (called bout types). This work started on touch elicited escapes (C-starts) (19) and was extended to other behaviours by Donald O'Malley (20, 21). Looking at many swim bouts it becomes clear that many of them are very similar and also that they appear to form natural groups. For example, in the bout sequence of Figure 1.1 the fish performed two types of movements that are extremely stereotypical within each group. The first type of movement is biased and has a first half beat of large amplitude (bouts 1,2, and 4). The second type of bout is characterised by being symmetrical and having half beats that are similar between each other and have small amplitudes (bouts 3 and 5).

Using a combination of careful human observation and calculation of key kinematic parameters at least ten bout types have been proposed across different studies that fish use to perform distinct behaviours. For spontaneous swimming it was reported that fish use a combination of slow swim maneuvers and routine turns (21), during hunting of paramecia the fish orient towards the prey by performing J-turns (22) and use capture swims to strike down the prey (23), when touched, larvae escape laterally using C-starts (19) and S-starts (24) and escape forward using burst swims (21). When

responding to pure tones the fish display two movement types; a typical short latency response (C-Start), and long latency C-starts (LLC) (25). In response to the dimming of light, fish perform turns that begin with a high amplitude, but slower bend of the tail, termed an O-Bend (26). Finally, fish perform struggles in response to chemicals and noxious heat and cold (27).

In spite of the large literature on zebrafish larvae behaviour, there is still controversy on the number of bout types that these animals use, the behavioural situations when they are used or even if fish use distinct swim types. For example, Borla and colleagues have described that larvae use three distinct movements while hunting paramecia; J-turns and tracking slow swims to orient and approach the prey, and capture swims to consume the paramecia (22, 23). In more recent reports that used state of the art video tracking systems and analysed large collections of prey capture swims it is defended that larvae change smoothly their movements to track the prey, not possessing any movement types while doing it (28, 29). Other authors that also collected large prey capture data sets have reported that larvae use J-turns and converge their eyes while tracking paramecia (30). Recently, Semmelhack *et al* have used a machine learning supervised classification system that learned to categorize J-turns and found neural correlates of the latter (31).

The ability to segment the swimming movements of zebrafish larvae into temporally discrete units offers a unique opportunity to determine if these discrete movements are clustered into distinct types or are better described as a continuum where no boundaries exist (29, 32, 33). Furthermore, due to the existence of a rich ethological literature, the zebrafish model promises to resolve the discrepancies between the classical ethology approach and unsupervised methods, by using the same time scale (the bout) in both approaches. If common ground is found in both approaches the unsupervised algorithms may allow the discovery of unforeseen types of movements, and confirm or correct the classifications that were made by ethologists, while preserving the same core idea that behaviour is created by sequences of stereotypical movements (27). If animals organize their behaviour into distinct types of movements, it is likely that they are produced by dedicated neural circuits, therefore

the knowledge of the repertoire of movements of an animal could be extremely useful to understand how the brain controls movements.

REFERENCES

1. Anderson DJ, Perona P (2014) Toward a Science of Computational Ethology. *Neuron* 84:18–31.

2. Luo L, Callaway EM, Svoboda K (2008) Genetic dissection of neural circuits. *Neuron* 57:634–60.

3. Brown AEE, Yemini EI, Grundy LJ, Jucikas T, Schafer WR (2013) A dictionary of behavioral motifs reveals clusters of genes affecting Caenorhabditis elegans locomotion. *Proc Natl Acad Sci USA* 110:791–6.

4. Wiltschko AB et al. (2015) Mapping Sub-Second Structure in Mouse Behavior. *Neuron* 88:1121–35.

5. Flash T, Hochner B (2005) Motor primitives in vertebrates and invertebrates. *Curr Opin Neurobiol* 15:660–6.

6. Tinbergen N (1951) The study of instinct. Available at: http://psycnet.apa.org/psycinfo/2004-16480-000.

7. Kato S et al. (2015) Global Brain Dynamics Embed the Motor Command Sequence of Caenorhabditis elegans. *Cell* 163:656–669.

8. Berridge K, Fentress J, Parr H (1987) Natural syntax rules control action sequence of rats. *Behav Brain Res* 23:59–68.

9. Kabra M, Robie AA, Rivera-Alba M, Branson S, Branson K (2013) JAABA: interactive machine learning for automatic annotation of animal behavior. *Nat Methods* 10:64–7.

10. Mirat O, Sternberg JR, Severi KE, Wyart C (2013) ZebraZoom: an automated program for high-throughput behavioral analysis and categorization. *Front Neural Circuits* 7:107.

11. Stephens GJ, Johnson-Kerner B, Bialek W, Ryu WS (2008) Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput Biol* 4:e1000028.

12. Stephens GJ, Johnson-Kerner B, Bialek W, Ryu WS (2010) From modes to movement in the behavior of Caenorhabditis elegans. *PLoS ONE* 5:e13914.

13. Stephens GJ, Bueno de Mesquita M, Ryu WS, Bialek W (2011) Emergence of long timescales and stereotyped behaviors in Caenorhabditis elegans. *Proc Natl Acad Sci USA* 108:7286–9.

14. Berman GJ, Choi DM, Bialek W, Shaevitz JW (2014) Mapping the stereotyped behaviour of freely moving fruit flies. *J R Soc Interface* 11.

15. Vogelstein JT et al. (2014) Discovery of brainwide neural-behavioral maps via multiscale unsupervised structure learning. *Science* 344:386–92.

16. Girdhar K, Gruebele M, Chemla YR (2015) The Behavioral Space of Zebrafish Locomotion and Its Neural Network Analog. *PLoS ONE* 10:e0128668.

17. Ahrens MB, Orger MB, Robson DN, Li JM, Keller PJ (2013) Whole-brain functional imaging at cellular resolution using light-sheet microscopy. *Nat Methods* 10:413–20.

18. Douglass AD, Kraves S, Deisseroth K, Schier AF, Engert F (2008) Escape behavior elicited by single, channelrhodopsin-2-evoked spikes in zebrafish somatosensory neurons. *Curr Biol* 18:1133–7.

19. Eaton R, Farley R, Kimmel C, Schabtach E (1977) Functional development in the mauthner cell system of embryos and larvae of the zebra fish. *J Neurobiol* 8:151–172.

20. O'Malley D, Kao Y-H, Fetcho J (1996) Imaging the Functional Organization of Zebrafish Hindbrain Segments during Escape Behaviors. *Neuron* 17:1145–1155.

21. Budick SA, O'Malley DM (2000) Locomotor repertoire of the larval zebrafish: swimming, turning and prey capture. *J Exp Biol* 203:2565–79.

22. McElligott MB, O'Malley DM (2005) Prey Tracking by Larval Zebrafish: Axial Kinematics and Visual Control. *Brain Behav Evol* 66:177–196.

23. Borla MA, Palecek B, Budick S, O'Malley DM (2002) Prey capture by larval zebrafish: evidence for fine axial motor control. *Brain Behav Evol* 60:207–29.

24. Liu Y-CC, Bailey I, Hale ME (2012) Alternative startle motor patterns and behaviors in the larval zebrafish (Danio rerio). *J Comp Physiol A Neuroethol Sens Neural Behav Physiol* 198:11–24.

25. Burgess HA, Granato M (2007) Sensorimotor gating in larval zebrafish. *J Neurosci* 27:4984–94.

26. Burgess HA, Granato M (2007) Modulation of locomotor activity in larval zebrafish during light adaptation. *J Exp Biol* 210:2526–39.

27. Prober D et al. (2008) Zebrafish TRPA1 Channels Are Required for Chemosensation But Not for Thermosensation or Mechanosensory Hair Cell Function. *J Neurosci* 28:10102–10110.

28. Trivedi C, Bollmann J (2013) Visually driven chaining of elementary swim patterns into a goal-directed motor sequence: a virtual reality study of zebrafish prey capture. *Front Neural Circuits* 7:86.

29. Patterson BW, Abraham AO, MacIver MA, McLean DL (2013) Visually guided gradation of prey capture movements in larval zebrafish. *J Exp Biol* 216:3071–83.

30. Bianco I, Kampff A, Engert F (2011) Prey capture behavior evoked by simple visual stimuli in larval zebrafish. *Front Syst Neurosci* 5:101.

31. Semmelhack J et al. (2014) A dedicated visual pathway for prey detection in larval zebrafish. *Elife* 3.

32. McLean D, Fan J, Higashijima S, Hale M, Fetcho J (2007) A topographic map of recruitment in spinal cord. *Nature* 446:71–75.

33. McLean DL, Masino MA, Koh IYY, Lindquist WB, Fetcho JR (2008) Continuous shifts in the active set of spinal interneurons during changes in locomotor speed. *Nat Neurosci* 11:1419–29.

CHAPTER 2 – Modulation of Swimming Speed in the Larval Zebrafish.

ABSTRACT

Animals often use distinct gaits to move at different speeds, and this requires the engagement of distinct neural circuits. Zebrafish larvae recruit different pools of spinal interneurons during slow and fast swimming, but the behavioural strategies that larvae use to control speed are not completely understood.

We have developed a system to perform high-speed online analysis of tail kinematics in freely swimming fish, while presenting visual stimuli. We found that zebrafish larvae will adjust their swimming speed when presented with whole-field visual motion stimuli.

Larvae match the stimulus speed by performing more movement events, by changing certain kinematic parameters, such as the bout duration and tail-beat-frequency, and by switching between two discrete modes of locomotion.

By characterizing the different behavioural components that zebrafish larvae use to modulate speed of locomotion we have established a powerful behavioural paradigm that may be used to understand general rules of how the vertebrate brain controls movement.

INTRODUCTION

Like all moving animals, zebrafish need to control their speed of locomotion to successfully navigate through the environment.

In the larval zebrafish spinal cord, there is a systematic relationship between the ventral-dorsal location of neurons and the swimming frequency that they are active (1). Furthermore, excitatory premotor interneurons form discriminable genetic and morphological classes are recruited according to the frequency of swimming (2). This implies that at least in the spinal cord there are distinct neuronal circuits that are responsible for the production of swims at different speeds.

In the juvenile/adult zebrafish the differentiation of distinct microcircuits for control of speed is even more striking. In adults, besides the fast muscle filers that exist in the larvae, there are also slow and intermediate muscle fibers (3, 4). The three types of muscle fibres are specifically innervated by distinct pools of motor neurons (5, 6)), that are in turn are activated by particular pools of excitatory interneurons (7). Muscle fibers, motor neurons and excitatory interneurons form functional units that may control distinct movements in the adult zebrafish.

Larval zebrafish, when swimming spontaneously, propel themselves forward with slow swims that consist of caudally localized tail oscillations and low head yaw amplitude and which are accompanied by alternating movements of the fins (8, 9). When touched on the tail larvae can produce much faster forward movements (burst swimming), that are characterized by large head yaw angles and the pectoral fins being tucked against the body (8, 9). While Budick and O'Malley divided forward swims occurring spontaneously or during escape behaviour into 'slow' and 'burst' categories, it is unclear, given the small numbers of swims analysed, whether they formed distinct categories. Analysing swimming at the level of individual tail beats, McLean et al. found a smoothly graded continuum of beat frequencies, accompanied by a continuous shift in the set of active spinal excitatory interneurons, implying that the zebrafish larvae do not switch between distinct gaits to move at different speeds

(1). However, when fast and slow pools of excitatory spinal interneurons were ablated slow and burst swims were differentially impaired (1). These results create a conundrum: in the one hand there is evidence for distinct neural circuits in the spinal cord for different speeds of locomotion, but on the other hand the forward movements produced by zebrafish larvae, as well as the recruitment of excitatory interneurons, appears to form more of a smooth continuum.

Here, we create a novel behavioural paradigm to study forward locomotion in the zebrafish larvae. We take advantage of the optomotor response (OMR), a visuomotor reflex present in many visual animals (10, 11), that consists in orienting and moving in the direction of the perceived motion of a whole-field moving stimulus (12) and change one parameter of the stimuli: the speed. Larvae match the speed of moving gratings for a wide range of velocities, and do so by: changing the latency to start moving and the frequency of locomotor events; modulating specific kinematic parameters, such as duration and tail-beat frequency; and by switching gait. Also, we use this behavioural paradigm to collect a large collection of forward movements at a wide range of velocities and use discontinuities in kinematic parameter distributions to classify bouts into turns, slow swims and fast swims.

RESULTS

Zebrafish Larvae Control Speed in Response to Whole-field Visual Motion

We aim to characterize the different behavioural components that zebrafish larvae modulate during swimming at different speeds. To record swim events at different speeds of locomotion we made use of the OMR, a reflex that allows larvae to maintain their position relative to a moving grating (12).

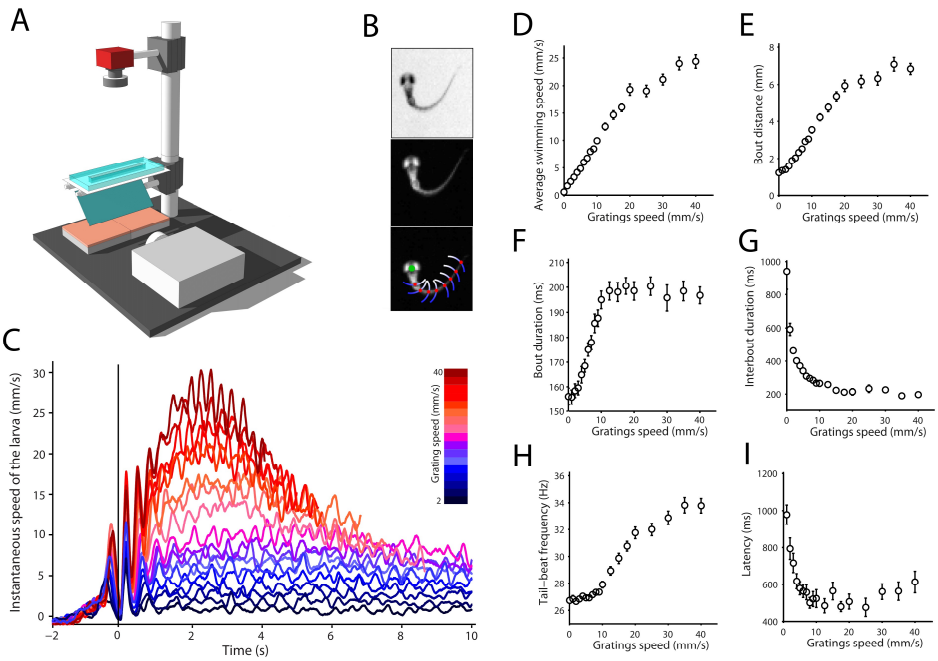

Figure 2.1 - Larvae Swimming Speed Changes with Grating Speed
Relevant kinematic variables, which describe the larval zebrafish swimming, are plotted against grating speed over a range of 0-40 mm/s.
(A) Schematic of the behavioural set up for freely moving larvae. High-speed video was acquired from above with drifting gratings projected on a screen below the arena. The arena was illuminated from below with infrared light.
(B) Image processing involves background subtraction, finding global maximum as centre of the fish (green point) and tail segmentation (red points) (Experimental Procedures).
(C) Instantaneous swimming speed versus grating speed. Traces are aligned so that 0 in the x axis is the start of the first bout in the stimulus direction. Each trace colour corresponds to the grating speed. (D)-(I) represent data from 52,938 bouts from 45 larvae.
(D) Average swimming speed during trial (mm/s) versus grating speed.
(E) Average bout distance (mm) versus grating speed.
(F) Average bout duration (ms) versus grating speed.
(G) Average tail-beat frequency (Hz) versus grating speed.
(H) Average interbout duration (ms) versus grating speed.
(I) Average latency (ms) versus grating speed. Error bars indicate SEM between fish.

Freely moving zebrafish larvae were presented individually with sinusoidal gratings moving at different speeds while high-speed video was acquired and their position

and tail movements were tracked (Figure 2.1 A-B). We were able to analyse the raw tracking data and divide it into epochs of movement (swim bouts) and stasis (interbout) (Experimental Procedures). For the movement events, we calculated relevant kinematic variables, and investigated how they changed with the speed of the stimulus (Figure 2.1C-I).

We started by confirming that larvae control their velocity with the speed of the gratings (Figure 2.1C-D). This relationship is linear for the average speed of the trial until grating speed of 20 mm/s (Figure 2.1D). For faster gratings the average swimming speed plateaus which probably reflects the upper limit of the speed that larvae are able to swim.

If we align the instantaneous speed of the larvae to the first bout performed oriented with the grating, we can observe that the speed of the larvae changes dynamically within a trial (Figure 2.1C). In Figure 2.1C the data on the left of the black line correspond to orienting manoeuvres the larva used to align its body with the axis of motion of the gratings. After zero on the x axis it can be observed for each trace that the larva's velocity increase is correlated with the gratings speed. Also the cyclic nature of this behaviour becomes apparent revealing the intermittent beat-and-glide swimming style that larvae use to move.

We next tried to find kinematic parameters that larvae modulate with the stimulus speed. We observed that the bout distance would increase linearly until 20 mm/s plateauing at higher speeds in a similar fashion to the average swimming speed (Figure 2.1D-E). So we went to look for bout kinematic parameters that could contribute to increasing bout distance and found that for slow gratings (0 to 10 mm/s) the duration of bouts increased (Figure 2.1F) while for fast gratings (12.5 to 40 mm/s) there was an increase of the tail-beat frequency (Figure 2.1G). Also, we found that for fast gratings the number of swim events increased (Figure 2.2H) and that the latency to start motor responses decreased (Figure 2.2I).

From these data we learned that zebrafish larvae are able to modulate their speed to maintain a tight correlation with moving gratings up to 20 mm/s. For slower speeds there is a complex interplay between bout duration, interbout duration and latency, while for higher speeds the tail-beat frequency is the most important contributor to the increase of speed of locomotion.

Larval Swim Bouts Cluster into Fast and Slow types

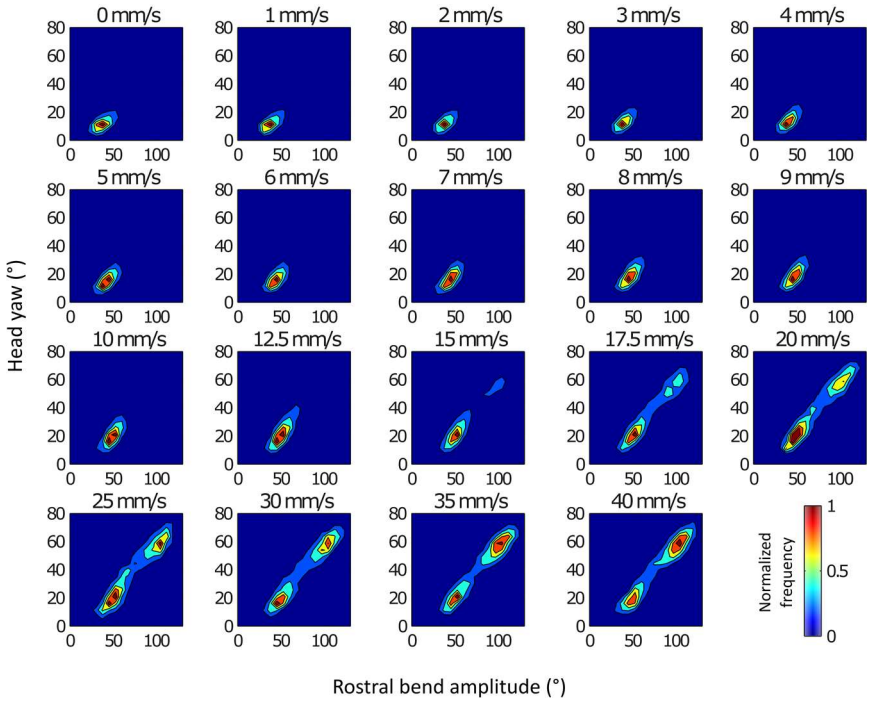

Figure 2.2 – Larvae Swim by Eliciting Bouts that Cluster into Two types
Bout probability distributions for the head yaw against the rostral bend amplitude.
(A – L) Bouts elicited at slow gratings speed stimuli. Gratings speed from 0 mm/s to 12.5 mm/s.
(M - S) Bouts elicited at fast gratings speed stimuli. Gratings speed from 15 mm/s to 40 mm/s.

There is controversy surrounding the behavioural strategy that vertebrates use to control speed. It may be that vertebrates switch between types of movements (called

gaits) to move at different velocities. Another possibility would be that they control speed by modulating a single class of movements (see the example of the mouse (13, 14)). So we wondered for the zebrafish larvae which is the case; do they control their velocity by having a single bout type that can be modulated by stimulus speed (2) or several bout types that are recruited for particular velocities? For the single bout type case we expect bouts to be distributed continuously through parameter space and to move in this space with the gratings speed. On the other hand, if the larvae utilize distinct gaits, we expect to find two or more distinct clusters in kinematic parameter distributions.

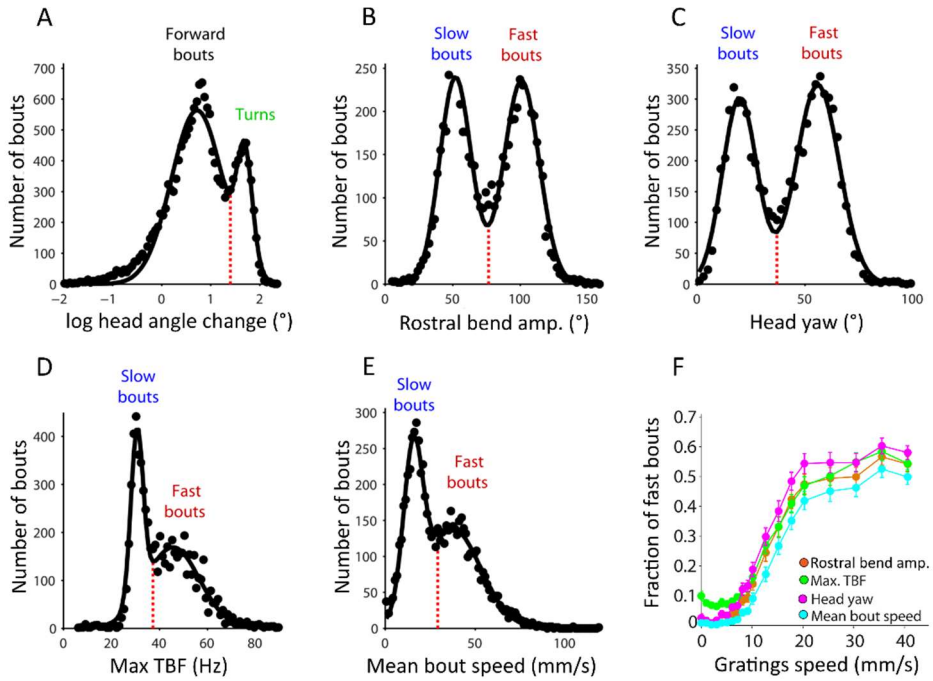

Figure 2.3 – Method to Categorize Turns, Fast and Slow Bouts
Black lines represent binormal distributions fitted to experimental data (black dots). Dashed red lines mark the categorization threshold.
(A) log head angle distribution (°) for all the trials.
(B) Rostral bend amplitude (°) distribution for fast trials (12.5mm/s to 40 mm/s).
(C) Head yaw (°) distribution for fast trials (12.5mm/s to 40 mm/s).
(D) Maximum TBF for fast trials (12.5mm/s to 40 mm/s).
(E) Rostral bend amplitude distribution for fast trials (12.5mm/s to 40 mm/s).
(F) Fraction of bouts in the faster category, as determined by different kinematic parameters: mean bout speed, head yaw, maximum TBF and rostral bend amplitude.

For slow grating trials, the bouts formed a single cluster in a kinematic space defined by the head yaw and rostral bend amplitude (Figure 2.2A-L). This cluster moves in this kinematic space, with head yaw becoming steadily larger as the grating speed increases (compare Figure 2.2A with figure 2.2L). For the fast-moving grating trials, two clusters were present with minimal overlap between them (Figure 2.2M-S). These data suggest that larval zebrafish utilize two gaits to move at different speeds.

Bout Types can be Distinguished Using Single Kinematic Parameters

After having observed that bouts naturally form distinct clusters for the head yaw and rostral bend amplitude we wondered if individual kinematic parameters could be used to categorize bout types. Our approach consists in finding discontinuities in kinematic parameter distributions. Briefly, we fit binormal equations to kinematic parameter distributions and define a cut-off threshold by finding the minimum value between the peaks (Figure 2.3A-E).

When larvae are performing the OMR, besides executing forward bouts at different speeds, they also carry out orienting maneuvers to align their bodies with the axis of motion of the gratings. The log head angle change (see experimental procedures) is one of the kinematic parameters that shows high values for turns and we used it to separate forward movements from turning manoeuvres (Figure 2.3A).

Using a similar approach, we used the distribution (12.5 mm/s to 40 mm/s) of the rostral bend amplitude, pooled across the faster trials, to categorize the forward movements into slow and fast bouts (Figure 2.3B). To assess the consistency of the categorization in slow and fast bouts, we used the previous approach using other kinematic parameters and found agreement in all cases (Figure 2.3B-F and Figure 2.4A-B).

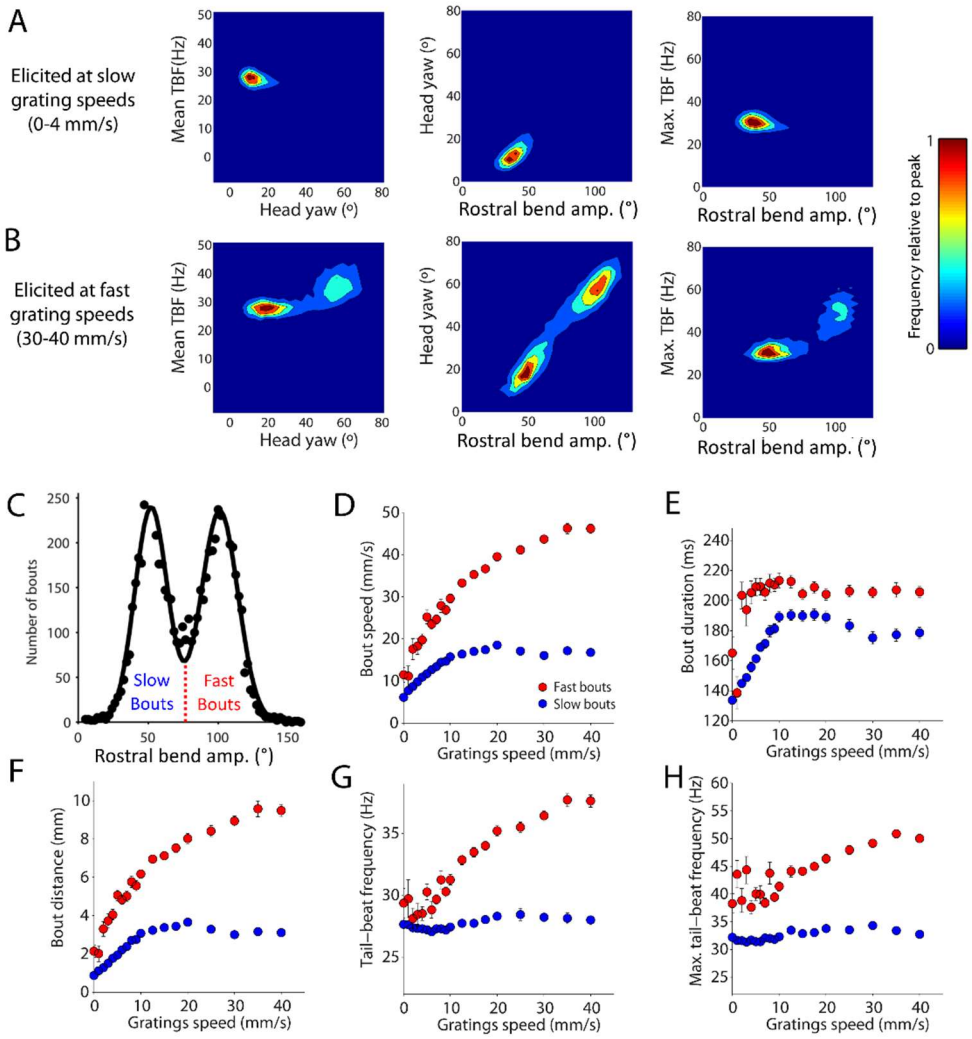

Figure 2.4 – Distinct Kinematic Variables Are Varied for Slow and Fast Bouts
(A) Bouts elicited at slow gratings speeds (0-3 mm/s). Joint distributions of several relevant kinematic parameters including mean and maximum TBF, head yaw, and rostral bend amplitude.
(B) Distributions of the same parameters as (A) for fast gratings speed (30-40 mm/s).
(C) Categorization of fast and slow swims by fitting rostral bend amplitude with a binormal distribution. Red dotted line: bout categorization threshold.
(D) Bout speed (mm/s) versus grating speed for fast (red) and slow (blue) bouts.
(E) Bout duration (mm/s) versus grating speed.
(F) Bout distance (mm) versus grating speed.
(G) Mean TBF (Hz) versus grating speed.
(H) Maximum TBF (Hz) versus grating speed. Error bars indicate SEM between fish (n = 52,938 of 45 larvae).

This categorization method enables us to divide the swimming movements that zebrafish larvae execute while performing the OMR into three distinct types: turns, slow swims and fast swims.

Distinct Bout Types Modulate Specific Kinematic Parameters

To test whether fast and slow bouts modulate kinematic parameters with the stimuli speed we repeated the same analysis as in Figure 2.1, but by sorting forward bouts in fast and slow (Figure 2.4C). Both bout types show a progressive modulation of speed and distance, although the slow swim modulation plateaus for gratings with speed higher than 10 mm/s while the fast bout modulation occurs for all the range of grating speeds (Figure 2.4D-2.4F). Fast bouts show a strong modulation of the maximum and mean TBF (Figure 2.4G-H), but are invariant for the bout duration; while the opposite is true for the slow swims, that are constant for maximum and mean TBF but show a robust increase of bout duration with the speed of gratings (Figure 2.4E).

In sum, not only do fast and slow swims form clusters for diverse kinematic variables, but they also show specific modulations of kinematic parameters with the stimuli speed.

Fast and Slow bout types are Organized in Stereotypical Sequences

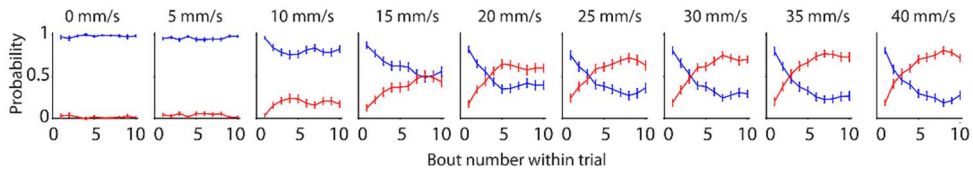

Figure 2.5 – Fast Bouts Are Recruited After Slow Bouts.
Probability that a bout will be slow or fast for different grating speeds plotted by the order of bouts elicited (bout number) independent of time. Error bars indicate SEM between fish.

It is quite puzzling that for high speed gratings the larvae use both slow and fast bouts (Figure 2.2M-S). One possibility is that zebrafish larvae organize their locomotion in stereotypical sequences in which the probability of a given gait is not random through a trial. Another possibility would be that for some animals or trials the fish would utilize only one of the gaits possible due to individual differences or fatigue.

In order to distinguish between these hypotheses, we computed the probability of a bout being fast or slow, according to its position in the sequence of bouts for a trial. For the stimuli where fast bouts are present in high percentage (grating speed larger than 10 mm/s, Figure 2.3F) larvae typically commence with slow swims and switch to using fast bouts over the first few swims (Figure 2.5). Thus, the bout sequence organization is not random and reflects most often a unidirectional switch of gait from slow swims to fast swims.

Larvae Need Appropriate Visual Feedback to Control Speed

Head restrained zebrafish larvae adjust their motor output if the feedback of the visual stimulus does not match their expectations (15, 16). We wondered if this is also true in the freely moving case and in what way is the gait switch and the modulation of kinematic parameters dependent on the visual feedback that occurs after movements are performed.

To tackle this question, we introduced to our previous experiments trials where the gratings are displaced in space the same amount as the instantaneous forward movement of the fish (virtual open loop). In these trials, the fish experience that the speed of the gratings is constant and independent of their own movements. In normal trials (called normal visual feedback trials) the fish perceive a decrease of the gratings speed that is proportional to the swim velocity produced by the bouts that are being executed.

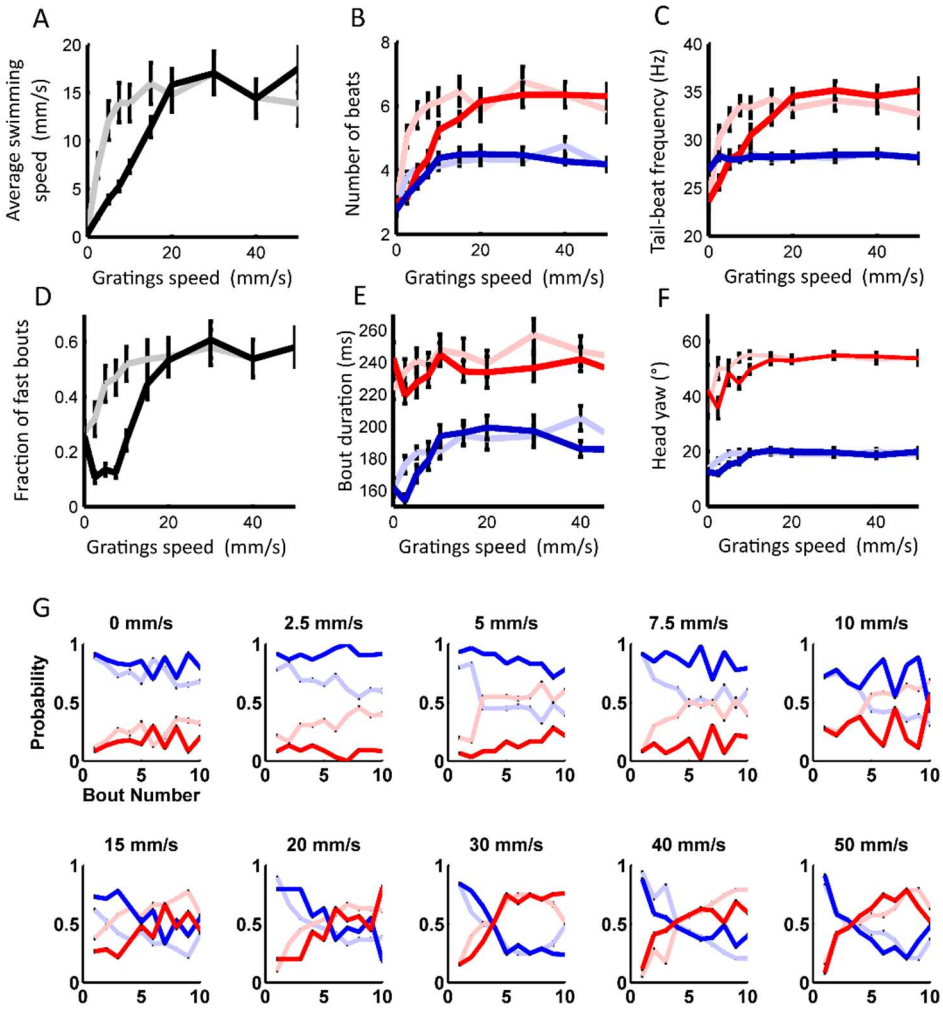

Figure 2.6 – Freely Moving Larvae Swim faster in Closed Loop
Light lines correspond to virtual open loop and darker lines correspond to normal feedback trials. Blue lines are slow bouts and red lines are fast bouts.
(A) Average swimming speed during trial (mm/s) versus gratings speed.
(B) Average number of beats versus gratings speed.
(C) Average TBF versus gratings speed.
(D) Fraction of fast bouts versus gratings speed.
(E) Average bout duration (ms) versus gratings speed.
(F) Average head yaw (°) versus gratings speed.
(G) Probability of bout type in sequence of bouts from start of trail for different gratings speed. Error bars are the SEM of the average between fish.

We observed that in virtual open loop the fish move at faster speeds for slow moving stimuli than in trials where the visual feedback was normal (Figure 2.6A). In accordance, in the virtual open loop situation the fish still exhibited slow and fast bouts and modulated the bout duration for the slow swims and the TBF for the fast swims, but these changes were shifted towards slow moving gratings when compared with the trials with normal feedback (Figure 2.6B-F). Also, for both situations the switch from slow swims to fast swims is present, but in the case of the virtual open loop it also occurs for slow speed gratings.

In sum the lack of visual feedback results in the overestimation of the gratings speed by the larvae, but all the hallmarks of the OMR are conserved.

DISCUSSION

The OMR is a behaviour that has been widely used in physiological studies of vision (10, 17), forward genetic screens for visuomotor defects (18), to understand the role of reticulo-spinal (RS) neurons in orienting turns (19, 20) and to study motor adaptation (15, 16). Here, we focused on the forward component of the OMR and changed one variable of the visual stimulus, the velocity, with the purpose of understanding the behavioural strategies that zebrafish larvae use to control speed.

We have found that when controlling slow speeds (which happened with moving gratings between 0 to 10 mm/s) larvae decrease the latency to start moving, shorten the interval between bouts, and utilize one gait (slow swims), that increases in duration as the speed of the gratings becomes larger. For fast locomotion (gratings faster than 12.5 mm/s) another strategy is used; in short, besides slow bouts, another gait is recruited (fast bouts) that increases in tail-beat frequency with the velocity of the visual stimuli.

Zebrafish larvae produce spontaneously the “slow-swim manoeuvre” (8, 9) and, when touched on the tail, the “burst swims” (2, 8). Both the slow swims and fast swims that

we describe here have very similar kinematic parameters to the "slow-like manoeuvre" and "burst swims" respectively, so it is quite likely that they correspond to the same type of movement, being recruited to perform another behaviour: the OMR.

The fact that larvae utilize two bout types to control speed of locomotion raises the hypothesis that there exist distinct motor circuits for each one. In the zebrafish, there are excitatory spinal interneurons that are exclusively active during fast swimming or slow swimming (1, 2). The two forward gaits we found for the OMR are correlated with slow and fast speeds of swimming, so it would be very interesting to know if fast and slow spinal interneurons are part of a larger circuit that produces these gaits.

It is thought that spinal CPGs are excited by descending glutamatergic RS neurons that exist in the mid- and hindbrain. In a survey on the activity of RS activity to moving gratings it was found that the group of RS cells that would show more activity for forward gratings was the medial longitudinal fasciculus (nMLF) (19). This cluster of cells consists of thirty RS cells, in which four are identifiable morphologically from fish to fish (big nMLF cells) (21), that can produce selective motor responses by preferentially activating spinal motor pools (22). Severi and colleagues have ablated the nMLF large cells and shown that fast swimming is impaired (23). Also, it is shown in the same study, that electrostimulation of the nMLF is sufficient to illicit swims at different tail-beat-frequencies, that change with increasing frequency of stimulation, and resemble slow and fast swims. Furthermore, it was found that specific large cells of the nMLF are correlated to tail-beat frequency and the duration of swims (23), two kinematic parameters that we found that fish modulate when executing slow and fast swims. Therefore, it is possible that cells in the nMLF are involved in the production of one or both gaits. Another option is that the nMLF is not initiating any of the two gaits, but is involved in the modulation of specific kinematic parameters that vary for these bout types (duration and frequency).

We also determined that all the behavioural strategies involved in controlling speed are present in the absence of a normal visual feedback. In the case we tested (gain 1)

the fish experiences the stimulus speed as constant, not having an appropriate visual feedback. Thus, it is possible that the major features of this behaviour can be studied in the head restrained case in open loop, a preparation where two photon calcium imaging is possible and no movement feedback is given (15) .

Overall we identified, for the OMR, three bout types. To support our categorization, we relied on finding several kinematic parameters that would form distributions with minimally overlapping clusters. This raises the possibility that kinematic parameters could be used to create a general behavioural space where clusters of bouts from several behaviours could be compared and categorized. Such common space linked with cutting edge clustering algorithms could be used as a general approach to categorize movement types and define movement repertoires.

EXPERIMENTAL PROCEDURES

Animal care

Fish were reared on a 14/12 hr light/dark cycle at 28 °C. Animal handling and experimental procedures were approved by the Champalimaud Foundation Ethics Committee and the Portuguese Direcção Geral Veterenária and were performed according to the European Directive 2010/63/EU.

Freely Swimming Behavioural Assay

Wild-type Tübingen zebrafish larvae at 6 days post fertilization swam freely in a 150 mm x 10 mm rectangular acrylic arena with 8 mm depth containing E3 medium. Their behaviour was recorded from above at 700 frames per second using an infrared-sensitive, high-speed camera (MC1362, Mikrotron), fitted with a machine vision lens (Schneider apo-Xenoplan 2.0/24) optimized for large sensors and a 790 nm long pass

filter. Fish were illuminated from below by a 20 x 10cm LED-based diffuse backlight (850 nm, Nerlite).

A sine wave grating with spatial period of 10 mm drifting at different speeds was projected onto a 150 mm x 150 mm opal glass diffuser 5 mm below the fish using a DLP projector (BenQ). Each stimulus presentation was initiated when the fish stayed 5 s in one of the extremes of the arena and it was terminated when the fish reached the opposite end or 30 s had elapsed. If a fish failed to reach the opposite end during this period, a 10 mm per s grating was shown to it until the fish swam the remaining distance. For the close loop trials, the position of the gratings was updated online with the position of the fish.

Acquisition, stimulus presentation, fish tracking and tail segmentation were performed on-line by a custom written program (Visual C#, Microsoft). Fish location was determined by similar methods to previous studies (24). Briefly, following background subtraction the image was smoothed with a 2-dimensional spatial boxcar filter, and the global maximum determined. This point always lies approximately between the larva's eyes. This point was then used to seed a flood fill on a thresholded version of the image, and the center of mass of the resulting shape which defines a consistent location on the larva's head, was defined as the larva's location. The direction of the tail was found by finding the maximum pixel value on a 0.65 mm diameter circle around this point, which corresponded roughly to the position of the swim bladder.

To evaluate tail curvature, we successively computed the angles of seven tail segments 0.39 mm long, by finding the center of mass of the pixel values along an arc centered on the end of the previous segment. Tail curvature was measured by summing the absolute deviation from the body angle at the head at all points along the tail. This measure was used to find the start and end of individual bouts.

A custom-made script written in Matlab 2010 (Mathworks) was used to compute the kinematic parameters. The angle of the last segment is used to count the number of oscillations of each bout and the tail beat frequency was calculated as the inverse of

the time between successive extreme tail positions in the same direction. The rostral bend amplitude was measured as the maximum peak-to-peak bend amplitude 1.17 mm caudal to the swim bladder in our analysis. The head yaw is defined as the maximum peak-to-peak amplitude of the angle of a line between the swim bladder and a point between the two eyes of the larva during each forward bout. The log head angle change is defined as the base 10 logarithm of the head yaw for all bouts. Bouts with one cycle of oscillation where excluded.

REFERENCES

1. McLean D, Fan J, Higashijima S, Hale M, Fetcho J (2007) A topographic map of recruitment in spinal cord. *Nature* 446:71–75.

2. McLean DL, Masino MA, Koh IYY, Lindquist WB, Fetcho JR (2008) Continuous shifts in the active set of spinal interneurons during changes in locomotor speed. *Nat Neurosci* 11:1419–29.

3. Raamsdonk W, Mos W, Smit-Onel M, Laarse W, Fehres R (1983) The development of the spinal motor column in relation to the myotomal muscle fibers in the zebrafish (Brachydanio rerio). I. Posthatching development. *Anat Embryol* 167:125–39.

4. Devoto S, Melançon E, Eisen J, Westerfield M (1996) Identification of separate slow and fast muscle precursor cells in vivo, prior to somite formation. *Dev Camb Engl* 122:3371–80.

5. Gabriel J et al. (2010) Principles governing recruitment of motoneurons during swimming in zebrafish. *Nat Neurosci* 14:93–99.

6. Ampatzis K, Song J, Ausborn J, Manira A (2013) Pattern of innervation and recruitment of different classes of motoneurons in adult zebrafish. *J Neurosci Official J Soc Neurosci* 33:10875–86.

7. Ampatzis K, Song J, Ausborn J, Manira A (2014) Separate microcircuit modules of distinct v2a interneurons and motoneurons control the speed of locomotion. *Neuron* 83:934–43.

8. Budick SA, O'Malley DM (2000) Locomotor repertoire of the larval zebrafish: swimming, turning and prey capture. *J Exp Biol* 203:2565–79.

9. Green MH, Ho RK, Hale ME (2011) Movement and function of the pectoral fins of the larval zebrafish (Danio rerio) during slow swimming. *J Exp Biol* 214:3111–23.

10. Orger MB, Smear MC, Anstis SM, Baier H (2000) Perception of Fourier and non-Fourier motion by larval zebrafish. *Nat Neurosci* 3:1128–33.

11. Tuthill J, Chiappe M, Reiser M (2011) Neural correlates of illusory motion perception in Drosophila. *P Natl Acad Sci Usa* 108:9685–90.

12. Portugues R, Engert F (2009) The neural basis of visual behaviors in the larval zebrafish. *Curr Opin Neurobiol* 19:644–7.

13. Bellardita C, Kiehn O (2015) Phenotypic characterization of speed-associated gait changes in mice reveals modular organization of locomotor networks. *Curr Biology Cb* 25:1426–36.

14. Machado A, Darmohray D, Fayad J, Marques H, Carey M (2015) A quantitative framework for whole-body coordination reveals specific deficits in freely walking ataxic mice. *Elife* 4.

15. Portugues R, Engert F (2011) Adaptive locomotor behavior in larval zebrafish. *Front Syst Neurosci* 5:72.

16. Ahrens MB et al. (2012) Brain-wide neuronal dynamics during motor adaptation in zebrafish. *Nature* 485:471–7.

17. Maaswinkel H, Li L (2003) Spatio-temporal frequency characteristics of the optomotor response in zebrafish. *Vision Res* 43:21–30.

18. Neuhauss S et al. (1999) Genetic disorders of vision revealed by a behavioral screen of 400 essential loci in zebrafish. *J Neurosci Official J Soc Neurosci* 19:8603–15.

19. Orger MB, Kampff AR, Severi KE, Bollmann JH, Engert F (2008) Control of visually guided behavior by distinct populations of spinal projection neurons. *Nat Neurosci* 11:327–33.

20. Huang K-H, Ahrens M, Dunn T, Engert F (2013) Spinal projection neurons control turning behaviors in zebrafish. *Current biology* 23:1566–73.

21. Kimmel CB, Powell SL, Metcalfe WK (1982) Brain neurons which project to the spinal cord in young larvae of the zebrafish. *J Comp Neurol* 205:112–27.

22. Wang W-C, McLean D (2014) Selective responses to tonic descending commands by temporal summation in a spinal motor pool. *Neuron* 83:708–21.

23. Severi KE et al. (2014) Neural control and modulation of swimming speed in the larval zebrafish. *Neuron* 83:692–707.

24. Burgess HA, Granato M (2007) Modulation of locomotor activity in larval zebrafish during light adaptation. *J Exp Biol* 210:2526–39.

CHAPTER 3 – Clustering by Search of Density Valleys.

ABSTRACT

It is common that data is composed of distinct groups of more similar elements. The structure in data often reflects the natural processes underlying the data being collected. One major challenge of machine learning is to develop reliable ways to automatically determine the number and the boundaries of clusters in diverse data types. Such an unsupervised approach can be useful to identify novel natural phenomena that otherwise would be hidden from the experimenter.

There is no clustering method that can solve all problems in data, however, an algorithm based on search for density peaks, has been proposed recently to be fast, resilient to noise, capture clusters of many different shapes and allow unbiased selection of the number of clusters.

Here, we automate the cluster center selection step of the density peak algorithm and search its parameter space. We find that the algorithm's success is highly dependent on the parameter that is used for estimating the local density. We also identify a class of synthetic data sets where the density peak heuristic fails to detect the most prominent clusters. In this chapter we propose an alternative clustering procedure based on the density valleys between clusters and adaptive gaussians to estimate densities. We apply the density valley algorithm with success to several synthetic and real world data sets and discuss the use of unimodal control distributions to automatically determine the number of clusters in data.

1. INTRODUCTION

Data is often composed of distinct sets of similar points. These so called clusters may reflect underlying structure in the variables being measured, but can also originate as a result of the data collection process. Also, the assumptions that a particular algorithm has may impact on the clustering solution that is obtained.

In many situations, for example where the data is very complex, or high dimensional, the experimenter may not have prior knowledge of the existence of categories in the data or of what natural processes are at work during the experiment. Unsupervised computational methods that can determine the number of clusters in data and define their natural boundaries are therefore useful to identify unsuspected natural phenomena in such circumstances.

Although clustering analysis has been widely used for more than sixty years there is no universal consensus on the definition of a cluster or on which clustering algorithm is the most effective (1). In fact, it was formally proven in "an impossibility theorem for clustering" that there is no single clustering function that can satisfy three fundamental criteria (scale-invariance, richness and consistency) implying this is one reason why the search for a unified framework for the clustering problem has been extremely difficult (2). In spite of that, it is possible to construct clustering algorithms that produce useful results by relaxing the proposed criteria (2). One of the aims of machine learning has been to develop general purpose clustering heuristics that function for diverse types of data and hence many clustering strategies have been proposed (for a useful review see (3)). However, there is no single clustering algorithm that can find all types of structure in data.

Hierarchical clustering algorithms are among some of the most used clustering methods, in particular in taxonomy and gene expression studies. They are distance-based algorithms that yield a hierarchy of nested clusters that normally are represented as a dendrogram (4). The cluster number is defined by the level where

the dendrogram is cut. The common criticism for hierarchical clustering algorithms is that they lack robustness, being sensitive to noise and outliers (3).

Another widely used algorithm is k-means clustering (5). This algorithm aims to minimize the sum of squared distances of points to the cluster centers. As a result, data points are always assigned to their nearest center, and so the success of this algorithm depends greatly on the shape of clusters in the data (6). One drawback of k-means and its derivatives is that the number of clusters to be found must be defined by the experimenter, and therefore a method is required to select the optimal number. While various methods and metrics have been proposed there is no efficient way to determine the number of clusters that works reliably across a wide range of data sets where the true underlying structure is known (6). One attempt to expand the k-means approach to capture different cluster geometries is ensemble clustering: a clustering method where data is split into a large number of spherical clusters by multiple runs of k means and the frequency with which pairs of points are assigned to the same cluster is used to determine the assignment of the natural clusters (7).

In model-based algorithms it is assumed that data consists of a mixture of underlying distributions that correspond to each cluster. This class of algorithms depends on the verification of the assumption that data is well represented by the mixture of distributions that the algorithm uses, being only able to capture clusters that possess that shape (8). For most cases it is impossible to ascertain this assumption before the data is divided into its components.

Density-based clustering methods can capture clusters with arbitrary shape. DBSCAN uses a density based threshold to discard noise and assigns high density points to clusters that belong to separate regions of space (9). However, choosing the appropriate value of the noise threshold can be difficult. Another density-based method is the mean-shift clustering algorithm that shifts each data point to the average data points in its neighbourhood and a cluster is defined by the points that converge to the same mode in the density probability function (10). The computational cost of

this method grows exponentially with the number of data points and hence becomes prohibitive for large data sets and dimensions.

Rodriguez and Liao have proposed a clustering method based in the fast search of density peaks that combines many of the qualities an ideal clustering algorithm should have (11, 12). Like the k-means method, it relies only on the calculation of distances between points, thus making it very fast and applicable to large data sets. The clusters are defined as maxima in a density function, but contrary to mean shift clustering there is no need to optimize the density field for each data point nor does it rely on a difficult to access noise parameter (as DBSCAN). The number of cluster centers has to be defined by the user, but by plotting two quantities that capture the quality of a cluster center, this selection can be made quite straightforwardly based on a single interaction with the algorithm. A fully automatic version of the density peak clustering has been proposed by Wang and Xu where the densities are calculated by a multivariate kernel estimation and the number of clusters are determined by maximizing the average silhouette index (13).

The outline of this chapter is as follows. In Section 2 we describe briefly the density peak clustering algorithm (11), present an approach to find automatically the number of clusters and identify two shortcomings that are critical for the success of this clustering method. In section 3 we present the density valley algorithm and explain how it improves upon the density peak clustering. Section 4 shows examples how the density valley clustering algorithm can solve several artificial and real world data sets. In section 5 we discuss our results.

2. CLUSTERING BY FAST SEARCH OF DENSITY PEAK AND ITS SHORTCOMINGS

The Density Peak Clustering Algorithm

Rodriguez and Liao's clustering algorithm (11) forms the basis for the clustering approach we are going to propose in section 3. The density peak algorithm assumes that clusters have a center that is surrounded by points of lower density and that cluster centers are far enough apart from any points with a higher local density. In order to capture this definition of cluster the algorithm calculates two quantities for every point: the local density (ρ) and the δ value.

The local density (ρ) is calculated by counting for each data point the number of points that are closer than the cut-off distance parameter (d_c). The δ quantity is defined as the minimum distance between a particular data point and any other point with higher density. For the point of highest density δ is defined to be the maximum distance from that point to any other.

Using this approach, cluster centers are defined as points that exist at a large distance from any other point of highest density (large δ) and simultaneously possess a high local density (ρ). Outliers also have large δ, but exist in isolation, having low values of local density (ρ). If we plot ρ vs δ (decision plot) the cluster centers can be identified by existing in the top right corner of this plot, while the outliers reside in the top left corner. After, the user identifies the cluster centers in the decision plot by drawing a square that encapsulates them and the remaining points are assigned to the same cluster as the closest neighbour of higher density.

The Problem of Determining the Correct Number of Clusters and the Use of Reference Distributions

Although the density peak clustering has many advantages compared to classical clustering algorithms the number of clusters is decided by the user (11). This restricts the usage of the density peak clustering to problems with few clustering tasks, makes unpractical the exploration of the d_c parameter space, and incorporates a subjective step in the algorithm that will unavoidably create distinct results for different users. We were interested to develop a version of the algorithm that did not require selection of any parameters and provided a robust and automatic method to select the cluster centers. Others have implemented a silhouette index based method to select the number of clusters for the density peak algorithm (13), but such a method has been found to be unreliable at finding the number of clusters in a variety of simulated and real data (14).

Estimating the optimal number of clusters is a major challenge in clustering analysis. This problem is challenging due to the very large number of shapes, sizes and dimensions that data can possess (1). The correct number of clusters in a data set corresponds to the groupings that exist in the distribution of the process that originated the data, be it artificial or natural. The most common reason why clustering analysis is performed is to infer from data how many groupings exist in the underlying distribution that normally is hidden from the researcher.

Let's take the example of a five gaussian mixture distribution. One can draw from the original distribution an infinite number of collections of points (data sets). For all the data sets that are created the correct number of clusters is five because that is the number of groups that exist in the five gaussian mixture that originated the data. However, in particular data sets there can be more or less than five clusters due to the random nature of the formation of the individual data sets. The more data points each data set has the closest it will be to the original distribution, but it will always have a finite number of points and possibly have false clusters.

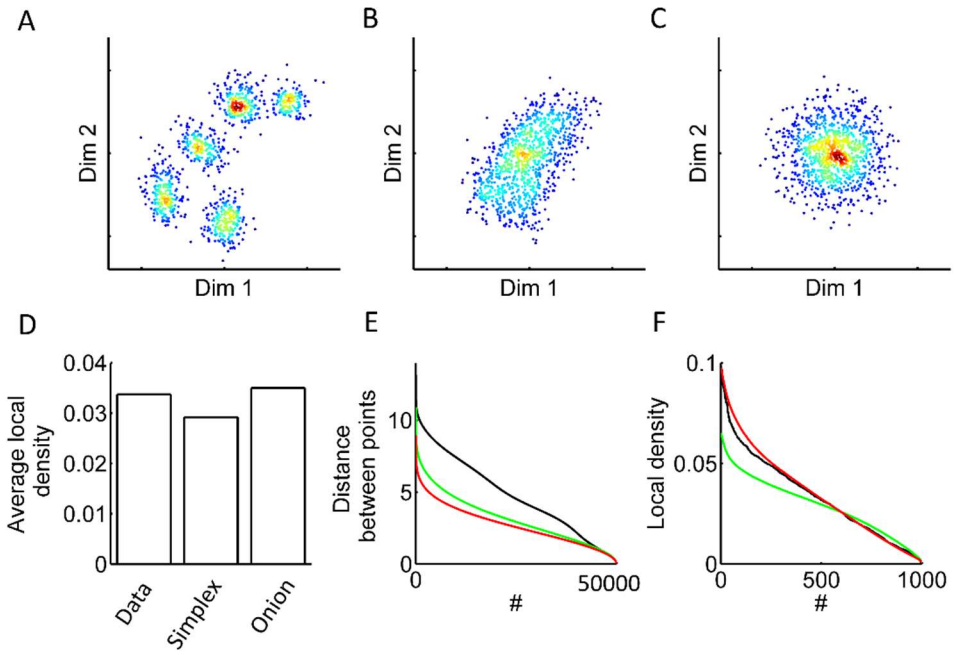

Figure 3.1 – Methods to calculate reference distributions. (A) Synthetic point distribution drawn from a mixture of five gaussians. (B) Reference distribution obtained by resampling the distribution in A using the simplex method. (C) Reference distribution obtained by resampling the distribution in A using the onion method. (A-B) Colours represent local densities calculated by using an adaptive mixture of gaussians density estimator. (D) Average local density between the three distribution in A-C. (E) Sorted distance profile between points. (F) Sorted local densities. (E-F) Black line, original distribution. Green line, simplex distribution. Red line, onion distribution.

Since points selected randomly from a unimodal distribution will show some random fluctuations of density by chance, it is necessary to set a threshold for cluster detection on a particular data set, based on the results of applying the same method to an unstructured reference distribution. Such an approach was used to develop the Gap statistic, a method that finds the number of clusters by calculating a measure of cluster consistency (the pooled within cluster dispersion) and subtracting the log of this value to the same value obtained from a reference distribution that contains one cluster (Gap). The Gap value is calculated in sequence for several number of clusters (K) and the correct number of clusters is found by finding the solution with the largest Gap value (15).

The Gap statistic reference distribution is created by resampling a uniform distribution contained in a box that is aligned with the principal components of the data (15) . This method creates a uniform distribution with a similar shape to the data but with a density structure profile that is close to uniform. Since the density peak algorithm relies on local densities we reasoned that a reference distribution with uniform density profile would not be a fair comparison to use.

Thus we have established two methods to create unimodal reference distributions that are straightforward to calculate and which recapitulate distinct features of the original data. We will refer to them here as the simplex method and the onion method.

The simplex method aims at creating a reference distribution within the same convex hull as the original data (Figure 3.1A-B), but without multiple density peaks (Figure 3.1D-F). Briefly, every set of ndim + 1 points defines a simplex, where ndim is the number of dimensions of the data. We sample points randomly from within this set of simplexes. Since points that are closer together define simplexes with smaller volumes, we sample the set of simplexes with a probability that is proportional to their volume, which smooths over any local variations in density, and gives a distribution which does not have multiple peaks.

For the onion method we sacrifice the shape of the distribution (Figure 3.1A-3.1C) in favour of having a similar local density profile to the original data (Figure 3.1D-3.1F). We make the onion reference distribution by first calculating the local density profile of the data, and arranging the points in decreasing order of density. We then choose the first point randomly from within an n-sphere, whose radius is chosen so that the local density is that of the densest point. Subsequent points are then chosen from n-spherical shells surrounding the previous level, whose outer radius is chosen to match the local density of each point in the original distribution. The onion method creates a unimodal reference distribution with an average density and local density profile similar to the original data, but which will always have a spherical shape.

The pairwise distance between points is dependent on the structure that exists on the data. Because both methods create unimodal reference distributions the distance profiles are very similar between each other, but distinct from the original data (Figure 3.1E).

Automatic Cluster Centers Identification for the Density peak Algorithm

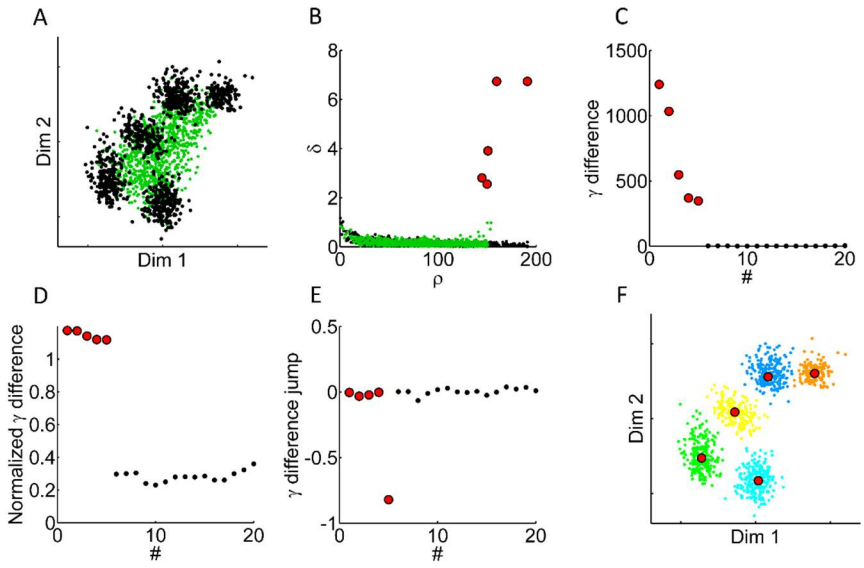

Figure 3.2 – Automatic method to find the number of clusters for the density peak method. (A) Synthetic point distribution drawn from a mixture of five gaussians (black). In green, a reference distribution with the same geometric as the black distribution obtained by the simplex method. (B - E) Mathematical transformations used to detect automatically the number of clusters, red points correspond to cluster centers found by the algorithm. (B) Decision plot (δ vs ρ) of the two distributions in A. (C) $\gamma = \rho^* \delta$. Difference between the γ values of the gaussian mixture distribution and the average values of 100 random distributions, the points are ordered for decreasing values of γ difference values. (D) Normalized γ difference values. (E) Difference of values in D (jump), the minimum point corresponds to the number of cluster centers. (F) Point assignment of black data set in A according to the number of clusters found in E. Notice that the five clusters found in F correspond to the five gaussian distributions that are combined to form the black distribution in A.

By definition $\gamma_i = \rho_i \delta_i$ will be large for cluster centers that belong to clusters that are well separated and have many points (11). So we rely on this quantity to find a "jump"

on the data that signals when data points stop being cluster centers and start being data points belonging to an already identified cluster, or outliers (Figure 3.2).

First we create a large number of unimodal reference data sets using the simplex resampling method (one hundred for Figure 3.2). We used the simplex method because the density peak clustering uses pairwise distances to calculate its two measures and the distances between points are highly influenced by the shape of the distribution.

Since the reference distribution only has one cluster the point with the largest value of γ corresponds to real structure in the data, while all the other γ values do not correspond to real clusters. Our purpose is to have a good estimation of the γ values for the reference distributions, in order to be able to infer for the experimental data which values of γ are due to real clusters. So if we subtract the sorted average γ values of the reference distributions from the real data we obtain the γ difference distribution that should reflect the presence of real structure in the data (Figure 3.3C). The number of clusters of the data set can be determined by finding which values of γ difference distribution have a value significantly larger than zero. One way to select the appropriate cut-off is to find the elbow in the sorted distribution, but this is not always very easy to define. We took a different approach to the problem by determining how much the γ difference values deviate from the reference distributions. This is done by dividing the γ difference values by the γ values of the original distribution and force the resulting quantity to be positive by subtracting the minimum value (normalized γ differences) (Figure 3.3D). Finally, we find the cluster centers by finding the largest jump on the normalized γ values (Figure 3.3E-F).

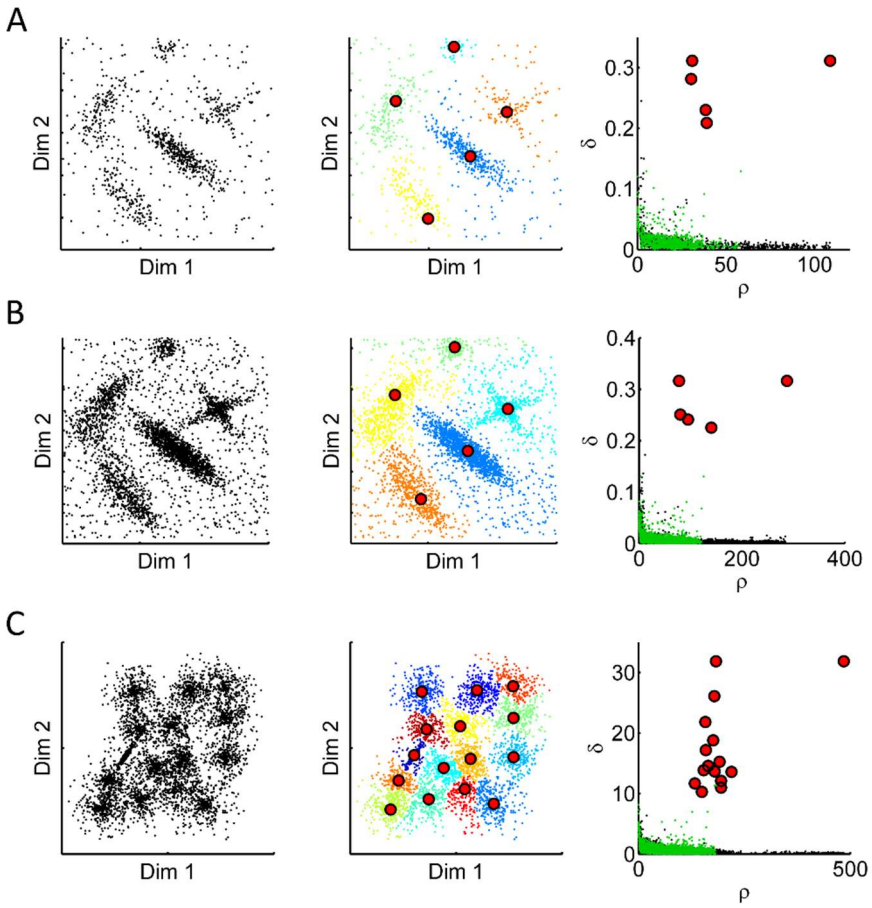

Figure 3.3 – Results for automatic density peak clustering for synthetic cases of data with noisy or fuzzy clusters. In left panels synthetic data sets from (11) (A - B) and (16) (C). Central panels are the same data sets solved by the automatic version of density peak clustering. Right panels are decision plots (ρ versus δ) where red points represent cluster centers and green points come from one example reference distribution obtained using the simplex method. For each data set the d_c value that was used came from (11).

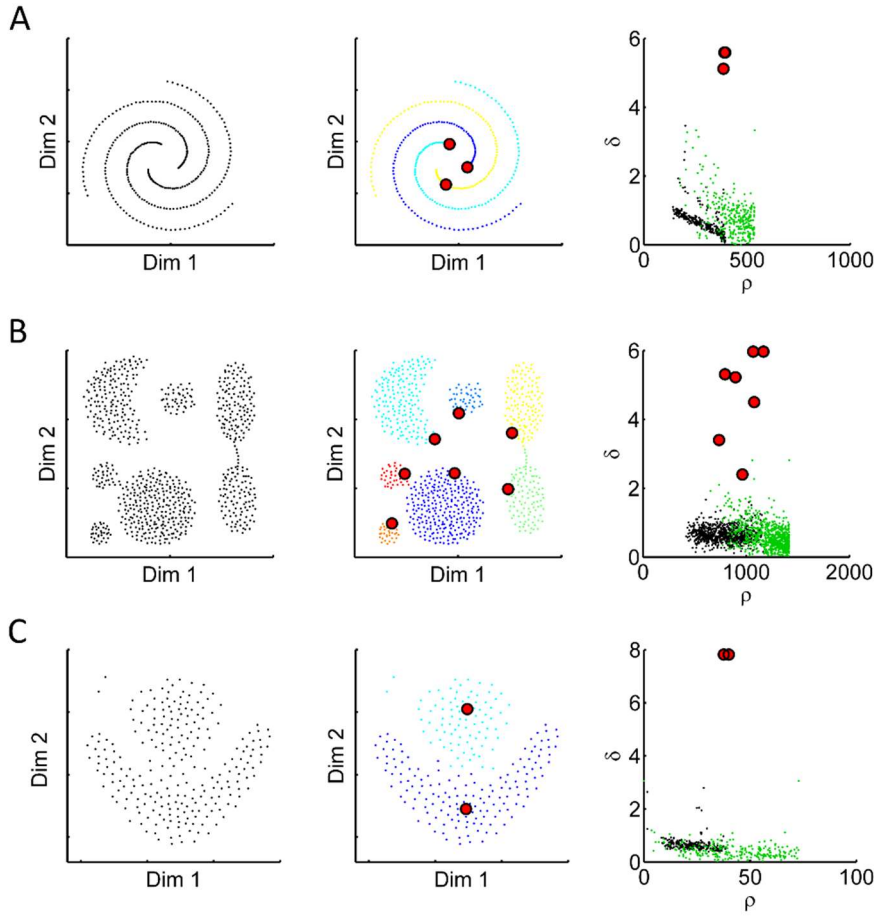

Figure 3.4 – Results for automatic density peak clustering for synthetic data sets with different shapes. In left panels synthetic data sets from (17) (A), (18) (B) and (19) (C). Central panels are the same data sets solved automatically by the density peak clustering algorithm. Right panels are decision plots (ρ versus δ) where red points represent cluster centers and green points come from one example reference distribution obtained using the simplex method. For each data set the d_c value that was used came from (11).

We have applied the automatic version of density peak clustering to the six synthetic data sets that Rodriguez and Liao used to validate the density peak algorithm (11). These six data sets were designed to recapitulate distinct features that exist in real data and that make clustering a hard problem, namely: clusters can have different densities (Figure 3.3A), data can be polluted with noise (Figure 3.3B), clusters can be fuzzy (Figure 3.3C), and clusters can have arbitrary shapes (Figure 3.4A-C). For

all tested data sets, we were able to find the correct number of clusters and achieve the correct assignment if we use similar d_c values to the original study (11).

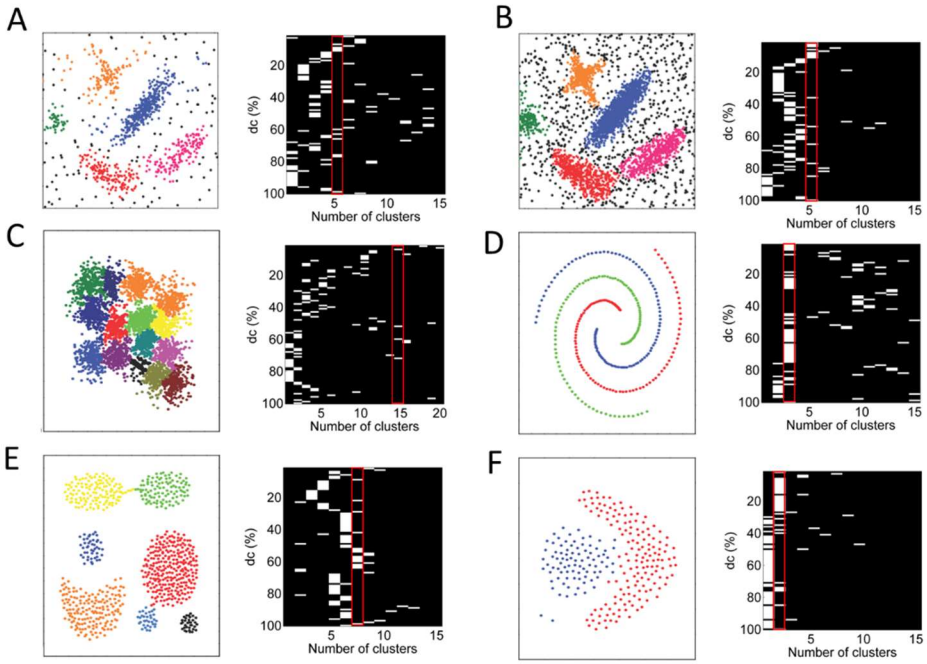

Figure 3.5 – Parameter d_c is critical for the success of the density peak algorithm. Left panels represent synthetic point distributions from (11) (A - B), (16) (C), (17) (D), (18) (E) and (19) (F). Colours represent distinct clusters assigned according to the ground truth. White squares in right panels correspond to number of clusters found using automatic density peak for a particular d_c (%) value. The red square marks the column with the correct number of clusters.

The Distance Cut-Off is Critical for the Success of the Density Peak Algorithm

In the Rodriguez and Liao's algorithm the local density is calculated using a truncated counting measure and is dependent on the d_c parameter (11). The authors suggest that the d_c should be such that the average number of neighbours is between 1% and 2 % of the total number of points in the data set (11). Our automatic method to detect cluster centers enabled us to systematically explore the d_c parameter on the synthetic data sets that were used in their paper. We found that for all the data sets we analysed

that there is at least one value of d_c that gives the correct number of clusters (Figure 3.5, right panels). However, the success of the algorithm is highly dependent on the value of d_c. With the exception of the spiral data set (Figure 3.5D), the most frequent solution of the algorithm was not the correct number of clusters (Figure 3.5A-C, Figure 3.5D-E). Furthermore, the range of d_c suggested by the authors (d_c = 1-2%) is not in the range where the correct solution occurs for most data sets (with the notable exception of Figure 3.5B).

In sum, the local density estimator of the density peak algorithm is highly dependent on the parameter d_c. The range of values of this parameter that give the correct solution is arbitrary and difficult to determine without knowing beforehand the solution and exploring the parameter space.

Distance to Higher Density Points Does Not Reflect Cluster Separability

The density peak algorithm calculates for every data point two quantities: the local density (ρ) and the minimal distance to points of higher density (δ) (11). In the original version of the algorithm the user chooses the cluster centers by drawing a rectangle that encloses them. Thus, the user's decision is not arbitrary (as it is in the k-means method for example), because the decision plot (ρ vs δ) not only informs the user about the "size" and "separability" of the clusters (by the ρ value and δ value respectively), but also restricts the user's choice. For instance, the user is unable to pick a cluster center that has lower local density and lower δ than a point that has not been picked. So the density peak algorithm implicitly creates a hierarchy of cluster centers and assumes that this hierarchy is correlated with cluster separability. This relationship must be true for the user (or any automatic algorithm) to have the possibility to choose the most separated clusters present on the data set without selecting clusters that are less separated and have arisen, for example, from noise in the density estimation.

In Figure 3.6 we have created a test case to show that δ does not fulfil such a role in all situations. The data points are drawn from a probability distribution of two non-overlapping uniform distributions of different densities (Figure 3.6A). Let's call the high density cluster A and the low density cluster B. For any particular instance of the data set (Figure 3.6B) the clusters will not be perfectly uniform, so the local density estimator will be plagued by some amount of noise and there will be some spurious local density variations that are produced by the random sampling in each data set.

One of the roles of the algorithm is to distinguish between cluster centers originated by noise and cluster centers that arise from structure in the original distribution; be it an automatic or user dependent method. We want to create a situation where noise in the density estimation will create sporadic density peaks that are not well separated, but are spatially far apart from each other. To achieve this, we choose to make cluster A larger than cluster B (Figure 3.6B).

We started by using our automatic method to sample the d_c parameter and found that there are many d_c values that give the correct solution of two clusters (Figure 3.6C). However, if we look at any of the two cluster solutions we can observe that the algorithm does not find the two clusters that exist in the original distribution (Figure 3.6A), but finds two cluster centers that are present on cluster A (Figure 3.6D-F). If we manually pick the three cluster centers with higher δ (Figure 3.6G-I) we observe that it is impossible to choose the center of B without first choosing the two cluster centers of A (Figure 3.6H). This is the case because the two centers of A, although they are not well separated, they are farther apart from each other than the cluster center on B is from any high density point of cluster A (Figure 3.6G).

This example shows that the δ quantity is not in all cases a good measure to capture the separability between clusters, in particular for data that is organized in distinct density plateaus. Furthermore, the δ quantity may induce the user or automatic algorithms to pick false cluster centers, making it impossible for any implementation of the density peak algorithm to solve data with these characteristics.

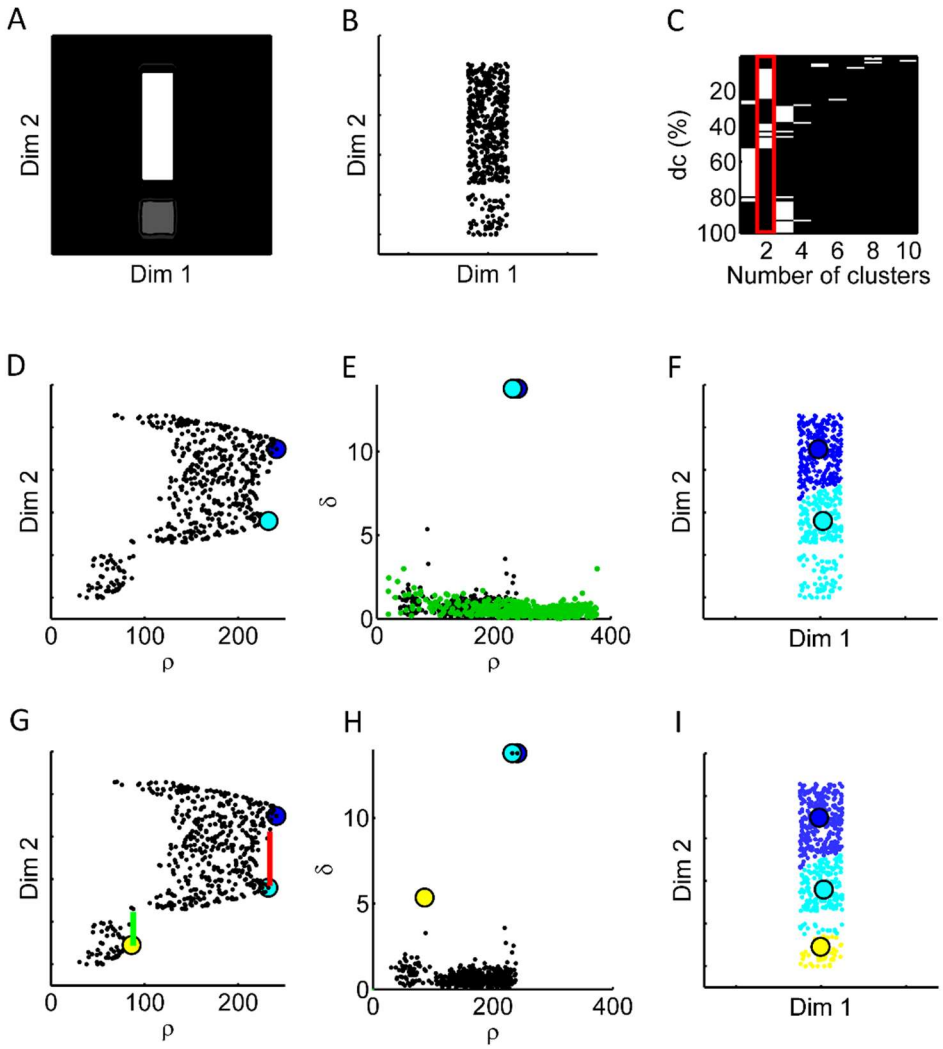

Figure 3.6 – δ does not represent the separability between clusters. (A) The probability distribution from which the point distribution in (B) was drawn. (C) White squares represent the number of clusters obtained by automatic density peak algorithm for a particular value of the d_c parameter. The red rectangle marks the column of the correct number of clusters. (D) Vertical projection of synthetic data set in B versus local density (d_c = 17%). (E) Decision plot for data set in B. Black points represent the synthetic data set while the green points represent a simplex reference distribution. (F) Cluster assignment for the automatic density peak solution (two cluster solution). (D-F) Light blue and dark blue circles correspond to cluster centers found in (E), neither correspond to the lower density cluster in (A-B). (G-I) Same distribution as in B, but clustering solution was obtained by choosing the three cluster centers with higher δ. In (G) the red and green lines represent the distances between the light blue cluster center and yellow cluster center to the respective point of higher density. Notice that red line is longer than green line.

3. DENSITY VALLEY CLUSTERING

The Density Valley Clustering Algorithm

The density peak algorithm possesses many features that would be common with the ideal clustering algorithm; it is fast, resilient to noise, captures clusters of many shapes and informs the user about the number of clusters that are present on the data. However, the fact that it relies on a difficult to access density parameter and that it is based on a cluster rule that is not general, makes it difficult to use it in real world situations where the cluster number is unknown. In order to develop a fast clustering approach that is fully automatic and more applicable to real world clustering problems we decided to change three key features of the density peak algorithm. First, use a more consistent density estimator; second find a more general rule to find the cluster centers and, finally use unimodal reference distributions to infer automatically the number of clusters.

As in Wang and Xu, we estimate the local densities using a multivariate gaussian density estimation (13, 20). We calculated this using the Matlab-based KDE (Kernel density Estimate) library (http://www.ics.uci.edu/~ihler/code/kde.html) (20), with the gaussian bandwidth for each data point chosen using a leave-one-out maximum likelihood estimate, and with all the bandwidths constrained to be proportional to the kth nearest neighbour distance where:

$$k = \sqrt{N} \qquad (1)$$

and N is the number of data points (20) (Figure 3.7A-B).

Next, we reasoned that a separability measure that can detect clusters that exist in different density plateaus (as the interrogation mark data set) could not be based on distances between points, because in data such as that the relationship between distances between cluster centers and separability breaks down. We argue that a better approach to quantify cluster separability is to measure the valleys in density profiles that exist between their centers. So we choose to create a heuristic that finds for every

point the shallowest density valley that connects it to each point in the rest of the data (Figure 3.7).

We first estimate the density profile along a set of straight lines connecting each pair of points (Figure 3.7A). To quickly calculate the minimum density on a line between points, we divide it into a fixed number of divisions (nSample parameter) and use our kernel density estimate to calculate the density at each point. We are interested in finding connections between pairs of points where the density lines form valleys. To detect valley connections, we find the minimum value for each density line. If a density line has the shape of a valley this value will be smaller than the densities of the original data points.

However, points may be connected via indirect paths that have shallower densities valleys than the direct connections (compare third and fourth panel of Figure 3.7A). It is vital for the algorithm to capture clusters of arbitrary shapes that it finds, for every pair of points, the lowest value in the highest density path (maximum density valley). We treat the search for the density paths as a clustering problem on its own accord and use the minimum of the density lines as a separability distance between pairs of points. We aim at finding a measure that is related to how pairs of points are separated between each other. To find such a measure we use the standard single link clustering algorithm, because at each step, this algorithm combines the two clusters that have the closest pair of elements that do not belong to the same group, creating a dendrogram where groups of points connected by density valleys create jumps in this plot (fourth panel of Figure 3.7A) (21). The maximum density valley is, for each data point, the minimum distance on the dendrogram needed to reach a point of higher density.

The highest density point does not have any other point of higher density, so by definition is a cluster center, and we calculate its maximum density valley as the lowest density valley of all the points.

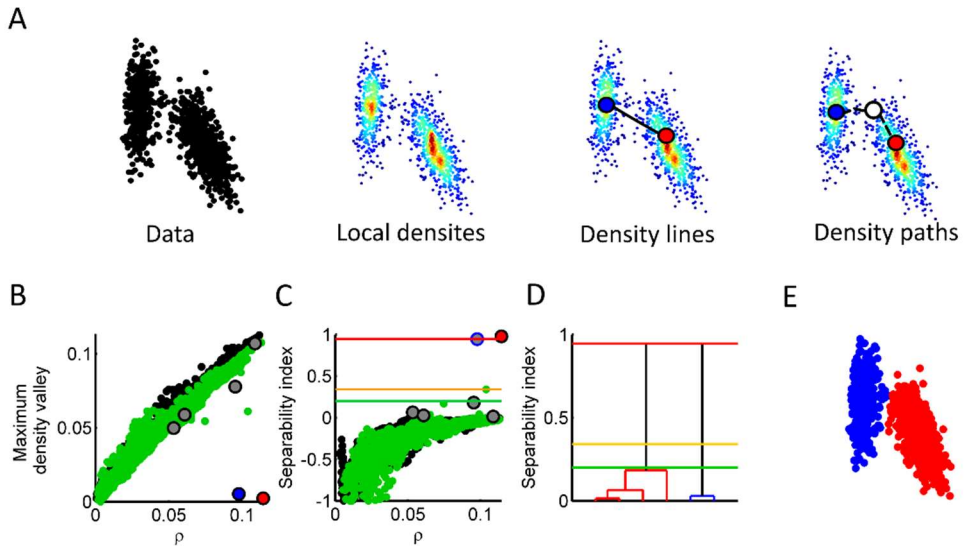

Figure 3.7 – The density valley algorithm. (A) Point distribution drawn from a two gaussian mixture (first panel). Local densities are calculated using an adaptive gaussian density estimator (second panel). Density profiles are calculated for the closest line between every pair of points (third panel). Single link algorithm is used to determine density paths between pairs of points and the minimum density of the path (white point) is considered the maximum density valley (fourth panel). (B) Maximum density valley versus local density (ρ). (C) Separability index versus local density (ρ). (B-C) Black points come from distribution in A; green points correspond to example resampled distribution obtained by onion method. (D) Dendrogram computed from separability index. (C-D) Lines represent cut-off measures used to exclude sporadic clusters: IC onion (green), IC simplex (orange), SI jump (red). Blue and red points are cluster centers that were not excluded by any of the cut-offs. (E) Cluster assignment of the point distribution in A.

For non-cluster points the maximum density valley quantity is highly correlated with the local density. Those points possess at least one non valley connection to another data point making the maximum density valley value be its own density or a higher density (Figure 3.7B). Cluster centers always have a value of maximum density valley smaller than its own density because all the density paths to points in denser regions must contain valleys (Figure 3.7B). The separability of a cluster center to the rest of the data will be given by the depth of its maximum density valley. Based on that we calculated a cluster center separability index (SI) by:

$$SI = \frac{-\,Maximum\ density\ valley}{\rho} + 1 \qquad (2)$$

where ρ is the local density. Every point that has a SI lower than zero cannot be a cluster center, while every point with a SI larger than zero is connected to the rest of the data by a valley in the density profile.

As can be observed in the example of Figure 3.7 not all points with positive separability indexes correspond to cluster centers that existed in the original distribution from which the data points were drawn (in this case a mixture of two gaussians) (grey points on Figure 3.7B-C). In this example these sporadic cluster centers come from errors in the density estimation due to the random procedure used to draw data from the original distribution. Thus the grey points should not be considered as real cluster centers. One option to exclude them is for a user to choose a SI threshold from which cluster centers are considered significant (a similar method of cluster center choice to density peak algorithm (11)).

Another option would be to use a completely automatic cut-off based on the SI quantity. We propose three distinct cut-off measures that have different meanings. Most synthetic data sets have clusters that are equally separated from each other. In these data sets the SI values of the real cluster centers will be similar to each other, while the SI values of the sporadic cluster centers will be smaller and randomly distributed. So a way to determine the meaningful level of separability would be to use the largest jump of the SI (SI jump). In data sets in which clusters are not equally separated (real world data or data sets with several levels of separation) there may not be a single jump on the separability dendrogram (red line in Figure 3.7C-D). For these types of data, one can use a cut-off criteria based on the confidence interval of the probability that unimodal reference distributions have of forming sporadic cluster centers. We can calculate such a confidence interval by creating one hundred reference distributions using the onion method and computing the 95^{th} confidence interval of the second point with higher SI value (green line in Figure 3.7C-D). A similar confidence interval can be created by using a collection of unimodal reference distributions produced by the simplex method (CI simplex) (orange line in Figure 3.7C-D). The CI onion will capture sporadic clusters formed due to the density profile of the data. The CI simplex will capture sporadic cluster centers that spawn from the

shape of the data. In the example of figure 3.7 all the three cut-offs give the correct answer of two clusters (Figure 3.7E). After the cluster centers have been found the remaining points are assigned to the same cluster of the nearest neighbour of higher density (Figure 3.7E).

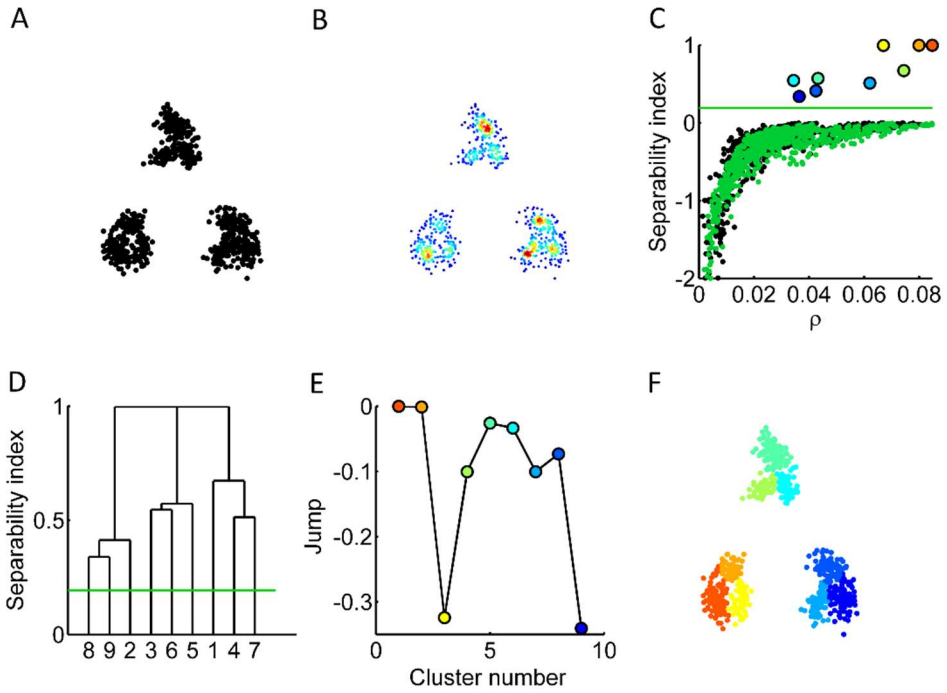

Figure 3.8 – Synthetic data set with clusters organized in two separability levels. (A) Point distribution drawn from mixture of nine gaussians. (B) Local densities are calculated using adaptive gaussian density estimator. (C) SI versus local density (ρ). Green points are example reference distribution calculate using the onion method. Green line is CI onion cut-off. Coloured points are cluster centers above the CI onion cut-off. (D) SI dendrogram. (E) SI jump versus cluster centers sorted by decreasing SI value. (F) Cluster assignment of the point distribution in A using the CI onion cut-off.

Hierarchy of Cluster Centers

The SI not only defines which points are cluster centers, but its value also indicates how separated a cluster center is from the other clusters. Points that have a SI value that is positive but close to zero will be connected to at least another high density

point through a very shallow density valley, not being well separated from that data point. Points that have a SI close to one will be connected to other higher density data points through valleys close to density zero. In this sense the SI can be used to create a separability hierarchy of the cluster centers. This hierarchy can be visualized by computing a dendrogram where every step of the dendrogram is the SI value where two cluster centers are joined together (Figure 3.7D).

The separability dendrogram can be useful to identify distinct levels of cluster center separability. Let's take for example a set of points drawn from a mixture of nine gaussians where the gaussians are close together in sets of three (Figure 3.8A) (22) . Although such a distribution has nine clusters (Figure 3.8B) it is obvious that there are three groups of clusters that are very similar to each other. If we look at the SI for this data set, we can observe that this measure has two groups of values: three cluster centers with a SI value close to one and six cluster center of a SI value close to 0.5 (Figure 3.8C). The existence of two groups of cluster centers reflect the fact that this data set has clusters organized in two levels of separability: three groups of clusters that do not overlap; and within each cluster group, three clusters that are very close together. Accordingly, the dendrogram also reflects the two level organization, by having two distinct jumps on the SI values (Figure 3.8D-E).

In practice this data set has two possible solutions that correspond to the two levels of structure present on the data. The density valley clustering can be used to detect distinct levels of cluster separability that in real world data sets may be created by distinct natural processes.

Big data problem

One of the most attractive features of the density peak clustering approach is that the algorithm is carried out in a single step without any iterations, being fast and potentially useful for big data analysis. Furthermore, the calculation of the two critical quantities (ρ and δ) will increase linearly with the amount of data points.

The density valley clustering algorithm relies on the calculation of density lines which number increase with the square of the number of points. This makes the running time of the algorithm explode with the size and dimensionality of the data and make our approach impractical for datasets larger than ten thousand data points. To overcome this problem, we implemented a "fast" and "medium" version of the density valley clustering, that calculate a subset of all the possible density lines that exist on the data.

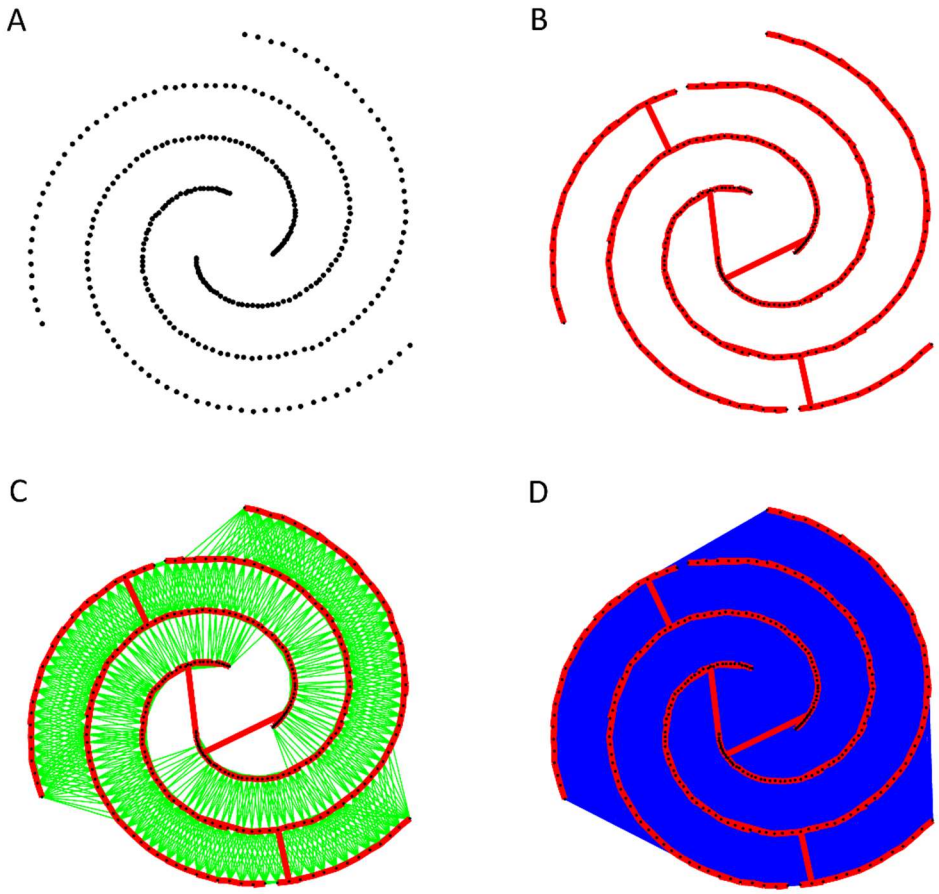

Figure 3.9 – Fast, medium and slow density lines calculators. (A) Spiral data set (17). (B) Minimum spanning tree of A (red lines). (C) K nearest neighbours of the minimum spanning tree of A, where k = $\sqrt{number\ of\ points}$ (green lines). (D) All connections between points in A (blue lines).

The fast version uses Kruskal's algorithm to calculate the minimal spanning tree (MST) of the data (23), and uses the pair wise connections of this tree to select the density lines to be calculated (Figure 3.9A-B). The computational time of the fast version of the algorithm increases linearly with the number of points, however it can yield false cluster centers because the MST only takes into account the nearest data point and there may be cases where the nearest point is located away from the cluster center. This will result in density paths connecting a point to its cluster centre that pass through an unnecessary dip in density. This problem is mitigated somewhat by also including connections between every point and the nearest point of higher density.

In order to mitigate the formation of incorrect cluster centers by the fast density valley clustering we implemented a version that runs at intermediate running speed, but that can improve significantly the accuracy of the algorithm. The medium version also relies in the calculation of a MST, but for each connection of the MST it searches the k nearest neighbour points and adds those connections to the density line list that will be calculated (Figure 3.9C). In this version of the density valley clustering algorithm the user must chooses how many k nearest points to add to the MST enabling him to decide how much he is willing to sacrifice speed for accuracy of the algorithm. In our hands the square root of the number of data points gives a good compromise, producing very few mistakes, while maintaining an acceptable running time of the algorithm for analysis of big data sets.

4. APPLICATIONS OF DENSTY VALLEY CLUSTERING

Artificial Data Sets

With the intent to benchmark the density valley clustering method we tested the success of the algorithm with a collection of fifteen synthetic datasets that recapitulate some of the diverse features that are present in real data, namely: diverse numbers of

clusters (Figure 3.10A-D), presence of noise (Figure 3.10E-F), fuzzy clusters (Figure 3.10G), existence of distinct separability levels (Figure 3.10H-I), clusters with arbitrary shapes (Figure 3.11A-D) and clusters that exist in different density plateaus (Figure 3.11E-F).

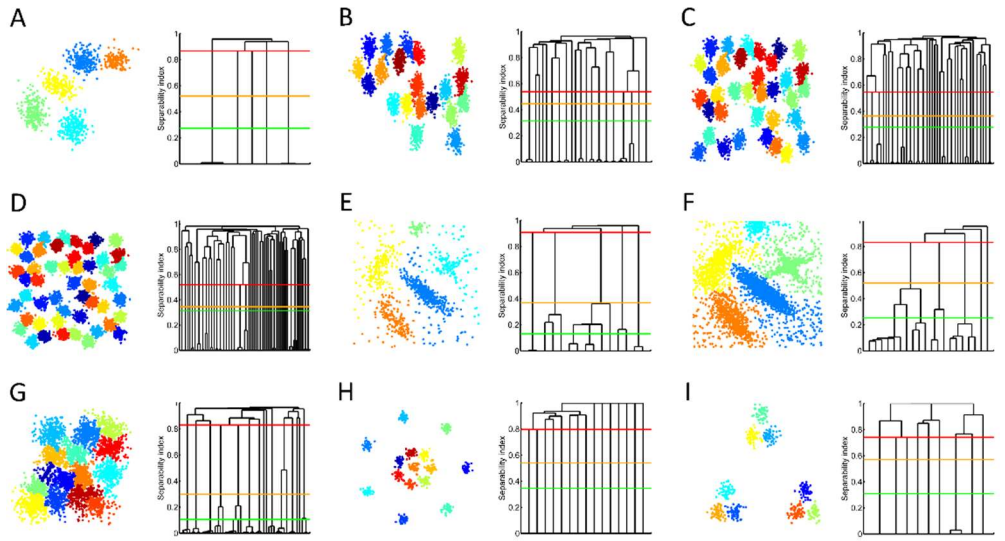

Figure 3.10 – Results for density valley clustering for synthetic cases of data with noisy or fuzzy clusters. Synthetic point data sets from this report (A and I), (24) (B-D), (11) (E - F), (16) (G) and (25) (H). In the left panels the assignment of the data set according to the SI jump criteria. The right panel has the respective SI dendrogram. Red line, SI jump. Orange line, CI simplex. Green line, CI onion.

To determine automatically the number of clusters we tested, for each data set, three distinct SI cut offs: SI Jump, CI onion and CI simplex. We found that the largest SI jump was able to find the correct number of clusters for all data sets (red lines in dendrograms), and was always above the noise threshold for both reference distributions. The CI onion cut-off overestimates the number of clusters for data sets that are plagued with noise (green line in Figure3.10E-F, Figure 3.10G, Figure 3.11D and Figure 3.11E). The CI simplex cut-off alone was enough to produce the correct solution in every case apart from the Jain data set (Figure 3.11D), probably because the clusters present in this data set possess internal gaps that would make it reasonable for them to be divided in more than two groups.

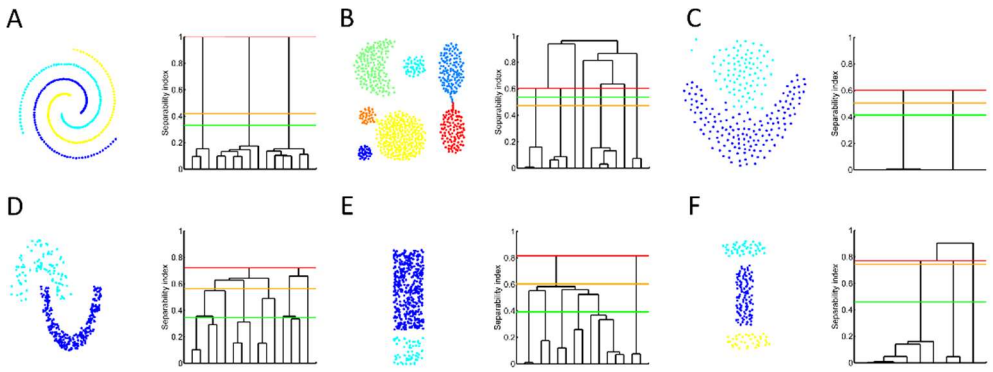

Figure 3.11 – Results for density valley clustering for synthetic cases of data with different shapes. Synthetic point data sets from, (17) (A), (18) (B), (19) (C), (26) (D) and this report (E-F). In the left panels the assignment of the data set according to the SI jump criteria. The right panel has the respective SI dendrogram. Red line, SI jump. Orange line, CI simplex. Green line, CI onion.

Real Data Sets

Next we applied our method to three real world cases that have known ground truths: Iris (27), Seeds (28) and Olivetti face data sets (29). A fundamental aspect of clustering real data is to embed the data in a space where the relevant structure is present. We used t-distributed stochastic neighbour embedding (t-SNE), a dimensionality reducing method that preserves the similarity between points and the structure on data (30) to create low dimensional spaces for these data sets (Figure 3.12).

Next we used the density valley clustering and applied the CI onion, CI simplex and SI jump cut-off measures to determine the number of clusters in these data sets. For the seeds data set (Figure 3.12C-D) all the cut off measures give the correct answer of three. For the iris data set the CI simplex and SI jump give the answer two while the CI onion is more permissive and gives the answer three. It should be noted that although the Iris dataset is composed of three species that two of the species are considered to form one single cluster (7, 27). In the case of the Olivetti data set no natural boundary exists in the dendrogram and thus no cut off measure gives the

correct answer (Figure 3.13 A-B). This is probably due to the inadequacy of the t-SNE space to capture all the forty face categories in the data (29, 30). In spite of that if we choose the forty cluster centers with the highest SI we are able to achieve a high rate of true association (78%) and a very low rate of false association (0.7%) (Figure 3.13 C-D). To our knowledge this is the best score for this data set achieved with an unsupervised method (11, 31).

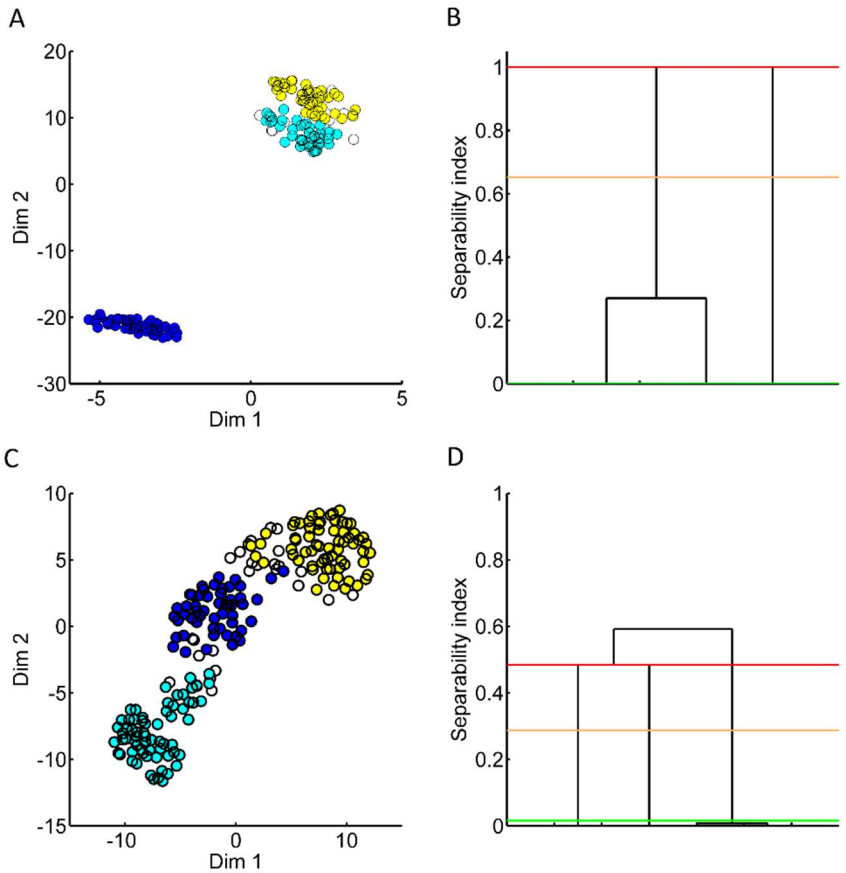

Figure 3.12 – Results for density valley clustering for the Iris and Seeds data sets. (A) The Iris data set embedded in a t-SNE space (27). (B) The separability dendrogram for the Iris data set. (C) The Seeds data set in a t-SNE space (28). (D) The separability dendrogram for the Seeds data set. Green line is CI onion, orange line CI simplex and red line is the largest jump of the SI.

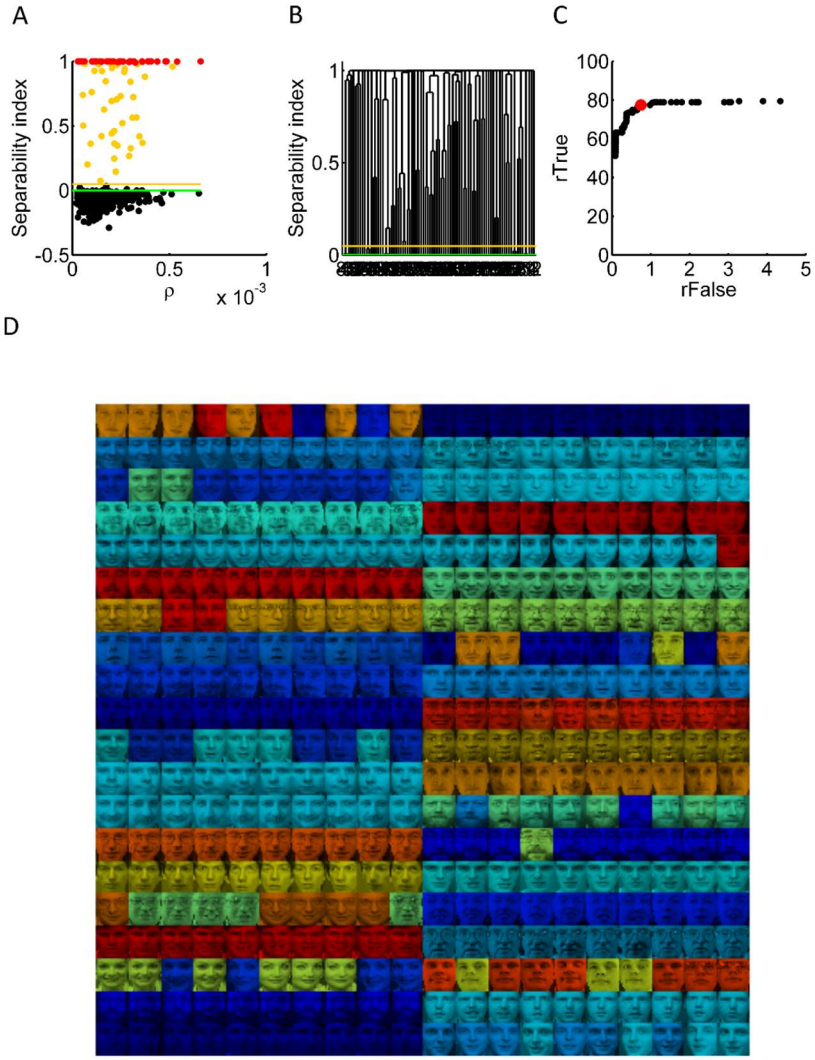

Figure 3.13 – Results for density valley clustering for the Olivetti data set. (A) SI versus local density (ρ). Red points are the forty cluster centers with higher SI values. Orange points are cluster centers above the CI simplex cut-off. (B) SI dendrogram of A. Green line is CI onion and orange line CI simplex. (C) Fraction of pair of images of the same subject correctly associated to the same cluster (r true) as a function of the fraction of pair of images of different subjects erroneously assigned to the same cluster (r false). Each point corresponds to a different number of putative centers. Red point corresponds to the solution of forty cluster centers; the number of subjects in the data set. (D) Assignment of the Olivetti data set for the forty cluster centers solution. Each picture is labelled with the colour of the cluster it was assigned to.

5. DISCUSSION

The density peak clustering algorithm introduced the idea of using a heuristic that is based on simple quantities calculated for each data point to find the centers of clusters. This avoids the use of iterations for the internal workings of the algorithm, which enables it to be performed in a single step, making it fast. However, the algorithm uses the distance between cluster centers as a measure of separability, which is a flawed criterion for data where clusters exist in distinct density plateaus. Furthermore, as noted by Wang and Xu, the algorithm's success is critically dependent on the parameter of the local density estimator (13). These two drawbacks make the density peak algorithm unsuited to be used for data of all types and to solve clustering problems where the cluster number is unknown.

With the aim of solving these two problems we propose a new approach that we call density valley clustering. First we use an adaptive gaussian density estimator to compute the local densities. We found that this density estimator is much more reliable and consistent than the truncated counting measure that Liao and Rodriguez use. It is so reliable that a simple rule of thumb was sufficient to estimate correctly the local densities for all the data sets we analysed (20). We must note that Wang and Xu arrived at a similar solution in their adaptive density peak method (13).

Second we tried to find a more general rule to find the cluster centers. We were inspired by the poor results that the density clustering algorithm showed for the exclamation mark data set and defined the separability between clusters as the maximum density valleys that separate cluster centers. Besides being more general, this approach has the advantage that it is possible to calculate a separability index (SI) that automatically excludes all points that are negative from being cluster centers. Furthermore, because the maximum valley heuristic is more general than the δ quantity the SI recapitulates accurately the hierarchy of separation of clusters. If we plot the dendrogram of the SI it is often obvious which points are the cluster centers and if they are ordered in distinct levels of separation, making it possible for a user to identify the cut-off SI that defines how many clusters exist on the data.

Although the picking of cluster centers by a user is often accurate, it is a method that has two major disadvantages: it is inherently subjective, since it is dependent on the user's criteria for defining a cluster center and is ill-suited for clustering problems with very many clustering steps. So we tried to find an automatic method to the best-number-of-clusters problem that would be compatible with the density valley clustering.

We were inspired by the Gap statistic, a statistical method that can be used with any type of clustering method and is based on comparing an internal measure of cluster consistency between the real data and a uniform reference distribution (15). Although conceptually this method is very appealing, it relies on the within-cluster homogeneity, so is not a measure that is well suited to all cluster shapes (22). Furthermore, it was shown for microarray data that the Gap method tends to overestimate the number of clusters (14). Yan and Ye have proposed to use a weighted version of the Gap statistic and proved in simulated and real data that this rule is more consistent (32), however their method is still based in uniform reference distributions.

For clustering purposes, the ideal reference is a distribution that has all the statistical features of the original data but is unimodal. We argue that the uniform distribution, although it does not have more than one peak in its distribution, and may have the shape of the data, does not share many of the properties that the real distribution has and thus is not an appropriate comparison to use. To our knowledge there is no method that is able to create an ideal reference distribution for all kinds of data, so we approached the problem by designing two methods that would create distributions that recapitulate different features of the original data that we considered relevant for density based clustering methods.

The simplex method creates a unimodal distribution that shares the same spatial distribution as the data with a single density maximum. Such a reference could be useful for clustering techniques that are sensitive to the shape of the data and thus we used it to on the automatic version of the density peak clustering.

The onion method creates references that are spherical but aim to recapitulate the density profile of the real data. Such reference distributions should be well suited to density based methods.

We applied confidence interval cut-offs based on both the types of reference distributions and applied them to the density valley algorithm. To benchmark the approach, we used twelve synthetic data sets and three real world data sets with known ground truths. First we found that we could easily solve all the artificial data sets using the largest jump of the SI value. We think this is the case because the synthetic data sets were built with a particular problem in mind and the clusters are for the most cases equally separated from each other, and thus the correct number of clusters is expressed in the SI dendrogram as the level with the largest jump.

Next, we found that the onion method would, for noisy data sets, overestimate the number of clusters, while the simplex method would almost always be with agreement with the ground truth. In spite of that we would like to point out that this may not mean that the simplex method is more accurate than the onion method. These results may vary on the shape of the data. Also, one could interpret that the onion method is finding finer structure in the data, because the reference distribution densities are similar to the real data. Probably a closer to ideal reference would be one that recapitulates the density profile of the real data and captures as well it's multidimensional shape, unfortunately we are not aware at this point of a simple method to generate a unimodal distribution with these properties.

One major interest in clustering analysis comes from the recent exponential growth and availability of data. One limitation of the density valley algorithm is that it depends on the calculation of density lines and their quantity increases to the square of the number of points. This makes this algorithm ill-suited to be used for the analysis of very big data sets. We have created two versions based on adding connections to the minimal spanning tree that perform much faster, and, although they sacrifice the guaranteed accuracy of the algorithm, produce similar results to the full calculation. The medium version possesses a parameter that defines how many k

nearest neighbours are added to the MST and thus enables the user to control the speed of the algorithm and use it in data sets with large number of data points.

Finally, the density valley clustering does possess one parameter, the nSample. It is the number of times the density lines are going to the divided. However, this parameter has very little impact into the final solution of the algorithm. The estimation of the local densities for each stamp relies on the density estimation and increasing the nSample more than a certain amount will have no impact because the gaussian bandwidths will be the same. In our hands using a nSample of ten is sufficient to give accurate results and increasing the parameter further had no practical impact into the solution.

Overall we can conclude that the density valley clustering is resilient to noise, captures clusters of arbitrary shapes, possess an easy to estimate parameter, that has little impact in the final result, is able to detect the number of clusters and levels of separability on the data, and with a small relaxation on its assumptions is well suited to big data analysis. Thus it may be a well suited tool to be used in real world situations where the number of clusters are unknown and data may have any form.

REFERENCES

1. Jain A, Murty A, Flynn P (1999) Data clustering: a review. *ACM Computing Surveys* 31:264–323.
2. Kleinberg J (2003) An impossibility theorem for clustering. *Advances in neural information processing systems*.
3. Xu R, Wunsch D (2005) Survey of clustering algorithms. *Neural Networks*.
4. Sneath P, Sokal R (1962) Numerical Taxonomy. *Nature* 193:855–860.
5. MacQueen J (1967) Some methods for classification and analysis of multivariate observations. *Berkeley Symp on Math Statist and Prob* 1:281–297.

6. Jain AA (2010) Data clustering: 50 years beyond K-means. *Pattern recognition letters* 31:651–666.

7. Fred A, Jain A (2002) Data clustering using evidence accumulation. *Pattern Recognition* 4:276–280.

8. Krishnan T, McLachlan GJ (2008) *The EM algorithm and extentions* (Wiley-Interscience).

9. Ester M, Kriegel H-P, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. 96:226–231.

10. Cheng Y (1995) Mean shift, mode seeking, and clustering. *Ieee T Pattern Anal* 17:790–799.

11. Rodriguez A, Laio A (2014) Clustering by fast search and find of density peaks. *Science* 344:1492–1496.

12. Wiwie C, Baumbach J, Röttger R (2015) Comparing the performance of biomedical clustering methods. *Nature Methods* 12:1033–1038.

13. Wang -F X, Xu Y (2015) Fast clustering using adaptive density peak detection. *Statistical Methods in Medical Research*.

14. Dudoit S, Fridlyand J (2002) A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome biology* 3:1–21.

15. Tibshirani R, Walther G, Hastie T (2001) Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society B* 63:411–423.

16. Fränti P, Virmajoki O (2006) Iterative shrinking method for clustering problems. *Pattern Recogn* 39:761–775.

17. Chang H, Yeung D-Y (2008) Robust path-based spectral clustering. *Pattern Recogn* 41:191–203.

18. Gionis A, Mannila H, Tsaparas P (2007) Clustering aggregation. *Acm Transactions Knowl Discov Data* 1:30.

19. Fu L, Medico E (2007) FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data. *BMC bioinformatics* 8:1–25.

20. Unknown (2007) Kernel Density Estimation Toolbox for MATLAB (R13) .

21. Sibson R (1973) SLINK: An optimally efficient algorithm for the single-link cluster method. *The Computer Journal* 16:30–34.

22. Sugar C, James G (2011) Finding the number of clusters in a dataset. *J Amer Statist Assoc* 98:750–763.

23. Kruskal J (1956) On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society* 7:48–48.

24. Fränti P (2000) Genetic algorithm with deterministic crossover for vector quantization. *Pattern Recognition Letters* 21:61–68.

25. Veenman CJ, Reinders MJT, Backer E (2002) A maximum variance cluster algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24:1273–1280.

26. Hutchison D et al. (2005) *Data Clustering: A User's Dilemma* (springer).

27. Fisher R (1936) The use of multiple measurements in taxonomic problems. *Annals of eugenics* 7:179–188.

28. Kacprzyk J et al. (2010) *Complete Gradient Clustering Algorithm for Features Analysis of X-Ray Images* (springer).

29. Sampat M, Wang Z, Gupta S, Bovik A, Markey M (2009) Complex wavelet structural similarity: a new image similarity index. *Ieee Transactions Image Process Publ Ieee Signal Process Soc* 18:2385–401.

30. Maaten L, Hinton G (2008) Visualizing data using t-SNE. *Journal of Machine Learning Research* 9:85.

31. Dueck D, Frey B (2007) Non-metric affinity propagation for unsupervised image categorization . *Computer Vision*:1–8.

32. Yan M, Ye K (2007) Determining the number of clusters using the weighted gap statistic. *Biometrics* 63:1031–7.

CHAPTER 4 – Automatic Behavioural Tracking Enables the Recording of Zebrafish Larvae Movements Over a Wide Range of Behaviours

ABSTRACT

Animal behavior is the final product of complex biological mechanisms that are present in multiple levels, spanning all the way from single molecules to large circuits of neurons. Therefore, the detection and understanding of behavioral phenotypes has been a key component used in a variety of experimental approaches that encompasses: forward genetic screens, psychophysical experiments, drug screens, optogenetic manipulations and ethological studies. However, in most cases, such studies are performed for a single behavioral context providing an incomplete description of the behavioral phenotype that is produced by a particular manipulation.

We developed a behavioral system that combines online high-speed tracking of several freely swimming zebrafish larvae, the segmentation of its body parts and the simultaneous presentation of visual and acoustic stimuli. We used this novel set-up to record the swimming responses that zebrafish larvae perform while engaged in nine innate behaviors. The use of this behavioral system enables the recording of swimming under the same experimental conditions for many different larval behaviors and may be useful to obtain a wider view on the impact of experimental manipulations on behavior.

INTRODUCTION

The biological processes that underlie behaviour are numerous and complex. Quantitative measurements of behaviour have proven to be a key tool to understand genetics, evolution, development, the nervous system, and human diseases.

Many behavioural experiments are based on large scale screens where the behaviour of thousands of animals needs to be measured. Recently, automatic high throughput behavioural systems, that enable the tracking of the position of multiple animals automatically, have been developed for the *C. elegans* (1, 2), fly (3, 4) and zebrafish larvae (5–7) (for a useful review see (8)). However, many different systems exist for each organism, with analysis in a particular study often restricted just to one or a few behaviours in response to a limited set of possible stimuli. Just within zebrafish studies, variation in experimental conditions, and behavioural analysis can make quantitative comparisons across different studies, and different behaviours challenging (9). Additionally, many methods require extensive offline image processing, which may make chronic high-speed recordings challenging due to data acquisition and storage issues.

Recently a wide array of tools has been developed for neuroscience based on optical approaches which are very applicable to the small, transparent zebrafish larva, and consequently this vertebrate model is emerging as an important system for the study of the genetic and neural basis of behaviour. For example, it is possible to record the activity of large populations of neurons in behaving animals (10, 11), single (12) and genetically defined populations of neurons can be targeted for ablations (13), and light can be used to manipulate neural activity through optogenetics (14, 15).

In parallel, several studies have reported that already at just five days of age zebrafish larvae have developed a varied array of innate behaviours, that enable them to explore the world (16, 17), stabilize their position in the environment (18, 19), respond to light changes (5, 20–22), pursue and capture prey (23, 24), avoid predation (25–27), and interact with conspecifics (6, 28) (for a useful review see (29)).

We developed an automatic behaviour tracking system, based on image processing algorithms that are applied to high-speed video in real time, which allows for the positions of multiple zebrafish larvae and their eye and tail movements to be tracked. We have applied this to two complementary set-ups in which 6-7 days post fertilization larvae swim in shallow acrylic arenas. Simultaneously, our system is able to present a wide range of visual and acoustic stimuli that can easily be changed to fit particular experimental needs. Using this tracking system, we were able to record millions of swim events of zebrafish larvae while they were engaged in nine distinct innate behaviours and to systematically sample the stimuli space that produced those behaviours. This approach may be useful to study thoroughly the behavioural phenotypes that are induced by genetic, pharmacological or optogenetic manipulations.

RESULTS

General Purpose Behavioural Tracking Systems

We aimed to create an automatic behaviour tracking system with sufficient time and spatial resolution to capture comprehensively the eye and tail movements of zebrafish while engaged in a wide range of behaviours. We focused on two sensory modalities, vision and audition; because most of the fundamental motor behaviours described in the zebrafish larvae can be elicited using these modalities.

We created two behavioural set-ups and tracking systems that fulfil distinct roles. The first, which has a high resolution, allows the tracking of one fish in a 2.5 x 2.5 cm area and measures the orientation of the fish' eyes as well as the tail. The second set-up, which has lower resolution, cannot track the larvae's eyes, but enables the simultaneous recording of behaviour of nine larvae in a wider area (5 x 5 cm). Both set-ups allow for tracking of tail curvature. We built these behavioural rigs in a similar way; they use an infrared sensitive high speed camera fitted with a visible light blocking filter to record the larvae's movements from above. The fish are illuminated

from below by a uniform, diffuse infrared backlight. A projector provides visual stimulation from underneath and two mini speakers deliver sounds.

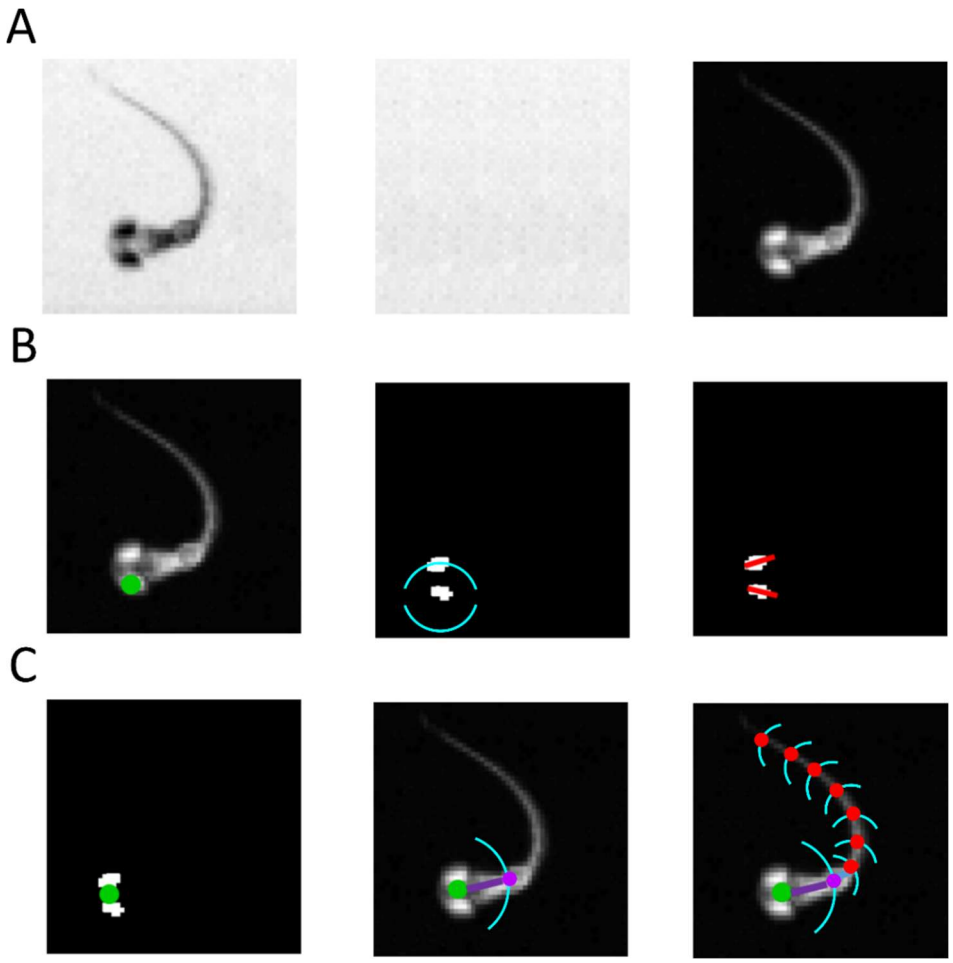

Figure 4.1 – Zebrafish larvae tracking for high resolution set-up. (A) Left panel, raw image. Central panel, background model. Right panel, inverted image with background subtracted. (B) Left panel, brightest pixel in the inverted image is located in one of the eyes (green point). Central panel, arc search for the second eye (blue lines). Right panel, the eye's centers are used to make a flood fill in a thresholded version of the image. Red lines indicate the principal axes of the found objects. (C) Left panel, fish location is defined as the middle point between the eyes (green point). Central panel, swim bladder is found by using an arc search of 0.9 mm length. Right panel, tail segmentation by successive arc searches. Purple point marks the swim bladder. Red points mark segments of the tail.

The high resolution set up uses a tracking system that first subtracts a background model from each frame, after which the darkest pixel in the image always corresponds

to one of the fish' eyes. Subsequently a sequence of searches along different arcs are used to first locate the other eye and then establish the fishes' orientation and track the curvature of the tail in 8-16 short segments (for details see Experimental Procedures and Figure 4.1).

The eye heuristic proved not to be reliable in lower resolution set-ups (190 pixels/cm), so, for these, we use a similar approach to Burgess and Granato (5). We smooth the subtracted image using a boxcar filter (Figure 4.1D), that transforms the larvae into "blobs", whose center of mass corresponds to a stereotyped location on the fishes' body (Figure 4.1E). After, we track the tail the same way as in the high resolution set-up (for details see Experimental Procedures).

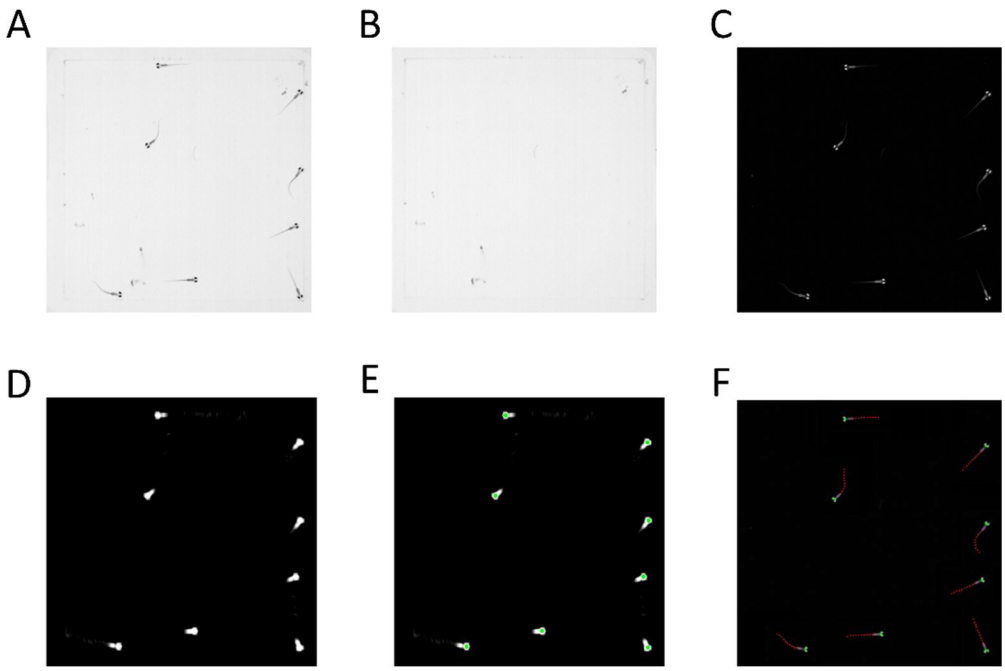

Figure 4.2 – Zebrafish larvae tracking for low resolution set-up. (A) Raw image with eight larvae. (B) Background model. (C) Subtracted image. (D) Smoothed image. (E) Centers of "blobs" (green points) correspond to larvae's positions. (F) Tail tracking (red points) of the eight larvae in A.

We used these set-ups to present 255 visual and acoustic stimuli to larval fish. Overall we recorded 1794076 bouts of 3204 fish that were actively engaged in one of nine innate behaviours: spontaneous swimming at different light intensities (5), responses

to light transitions (5), prey capture (23), directional OMR (12, 30), forward OMR (19), phototaxis (20), acoustic startle (26), avoidance of expanding spots (27), and social avoidance (6). We spend the next sections showing evidence that fish reliably perform these behaviours.

Spontaneous Swimming at Different Light Intensities

In the absence of salient sensory stimuli fish still perform "spontaneous" swimming manoeuvres with the assumed aim of exploring the environment (16, 17). It was shown that in continuous darkness larvae have a transient high level of locomotor activity that decreases with time or with the switch to light (5). We wondered how the level of spontaneous locomotor activity varies with changing levels of light intensity. Does locomotor activity decrease with light intensity in a continuous way?

To answer this question, we created a behavioural paradigm where, after thirty minutes of adaptation to darkness, fish experience transitions of three minutes between darkness (0 lux) and light with varying intensities (Table 4.1). We confirmed that, although there is great variability between individual animals (see grey traces in Figure 4.3A), the larvae's speed (a proxy for locomotor activity) in the dark periods is higher than in the light periods (Figure 4.3B). Also, the locomotor activity in the light periods decreases with the light intensity in a continuous way. Since, in our paradigm, the dark periods are always the same light intensity (0 lux) we were able to ascertain that the locomotor activity in the dark is not influenced by the light intensity of the previous light period (Figure 4.3B).

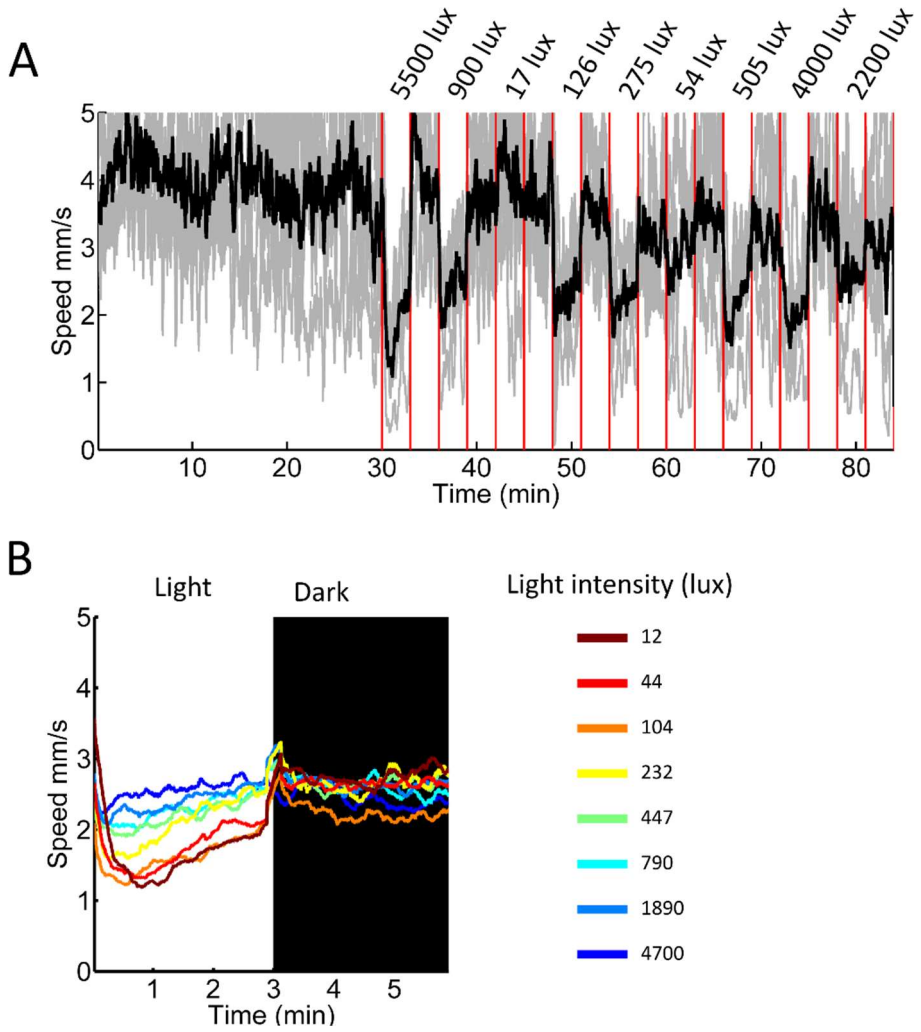

Figure 4.3 – Spontaneous swimming at different light intensities. (A) Example speed trace of three minute periods of darkness and light at a set of distinct light intensities. Red lines mark when stimuli change. Light stimuli at varying intensity are intercalated by dark periods (0 lux). Black line is average of five fish; grey lines are individual fish traces. (B) Average speed (mm/s) versus time (min) (n = 27). Data smoothed with a 15 s kernel.

Table 4.1. Spontaneous swimming at different light intensities.

Stimulus type	Light stimulus (lux)	Number of bouts	Number of animals
Homogenous square of light that changes every 3 minutes [a].	0	45374	27
	12	4971	
	44	5205	
	104	5657	
	232	6145	
	447	4996	
	790	5901	
	1890	5298	
	4700	5022	

[a] The light stimuli are intercalated by a 3 min dark stimulus (0 lux).

Responses to Light Transitions

Besides adapting the level of spontaneous locomotion to light intensity, larvae also execute specific movements in response to abrupt changes in illumination. We used a light transition protocol that consisted of adapting the larvae to three minutes of a particular background light intensity and recorded the responses of larvae to dimming or brightening changes of light intensity. Figure 4.4A shows a schematic of such a behavioural protocol. The diagonal line of this protocol corresponds to stimuli with continuous light intensity. The stimuli above the diagonal are brightening changes of light intensity. The stimuli below the diagonal are dimming stimuli. In order to thoroughly explore this stimulus space, we took advantage of the multitracking ability of our behavioural system and recorded the movements of hundreds of larvae for 72 different light transitions (Table 4.2-3).

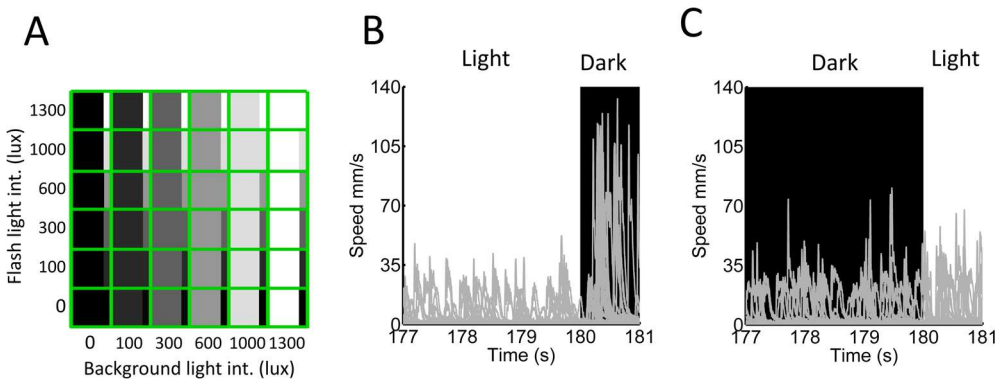

Figure 4.4 – Responses to changes of light intensity. (A) Protocol stimulus matrix with different light transitions. (B) Speed (mm/s) versus time (s) for dimming light transition (4229 lux to 0 lux). (C) Speed (mm/s) versus time (s) for brightening light transition (0 lux to 4229 lux). Each grey trace is a response from a distinct larva (n = 27), data was not smoothed.

As was described previously by Burgess and Granato, we observed that larvae respond very reliably to dimming stimuli by executing high velocity movements (Figure 4.4B) (5). For brightening stimuli, the responses are less reliable and vary according to stimuli and individuals, being often lower velocity than the responses produced by dimming stimuli (Figure 4.4B).

Table 4.2. Light transitions from 0 to 10000 lux.

Stimulus type	Light intensity before transition (Lux)	Light intensity after transition (Lux)	Number of bouts	Number of animals
Homogenous light square [a]	0	0	37	70
	0	769	68	
	0	2224	88	
	0	4229	102	
	0	6784	102	
	0	9886	107	
	769	0	85	
	769	769	73	
	769	2224	55	
	769	4229	81	
	769	6784	81	
	769	9886	75	
	2224	0	88	
	2224	769	22	
	2224	2224	49	
	2224	4229	53	
	2224	6784	48	
	2224	9886	49	
	4229	0	86	
	4229	769	44	
	4229	2224	27	
	4229	4229	53	
	4229	6784	55	
	4229	9886	52	
	6784	0	73	
	6784	769	44	
	6784	2224	27	
	6784	4229	38	
	6784	6784	42	
	6784	9886	34	
	9886	0	66	
	9886	769	42	
	9886	2224	20	
	9886	4229	29	
	9886	6784	38	
	9886	9886	39	

[a] Background light lasts 3 min. Flash of light lasts 1s.

Table 4.3. Light transitions from 0 to 1300 lux.

Stimulus type	Light intensity before transition (Lux)	Light intensity after transition (Lux)	Number of bouts	Number of animals
Homogenous light square[a]	0	0	131	259
	0	73	90	
	0	330	81	
	0	634	76	
	0	986	107	
	0	1386	135	
	73	0	102	
	73	73	111	
	73	330	126	
	73	634	109	
	73	986	122	
	73	1386	120	
	330	0	159	
	330	73	73	
	330	330	182	
	330	634	129	
	330	986	136	
	330	1386	121	
	634	0	204	
	634	73	104	
	634	330	79	
	634	634	173	
	634	986	132	
	634	1386	136	
	986	0	196	
	986	73	138	
	986	330	65	
	986	634	113	
	986	986	158	
	986	1386	151	
	1386	0	203	
	1386	73	174	
	1386	330	80	
	1386	634	81	
	1386	986	121	
	1386	1386	158	

[a] Background light lasts 3 min. Flash of light lasts 1s.

Prey Capture

Zebrafish larvae hunt live prey such as paramecia (23) and eye convergence has been shown to be a key indicator of the initiation of hunting behaviour (24). We used the high resolution set-up to be able to measure eye movements while fish are hunting. To ensure that the fish were hunting we counted the number of paramecia at the start and end of each experiment. As can be seen in Figure 4.5A, after 1 to 2 hours in the presence of prey, most fish had eaten the majority of the paramecia that were available to them.

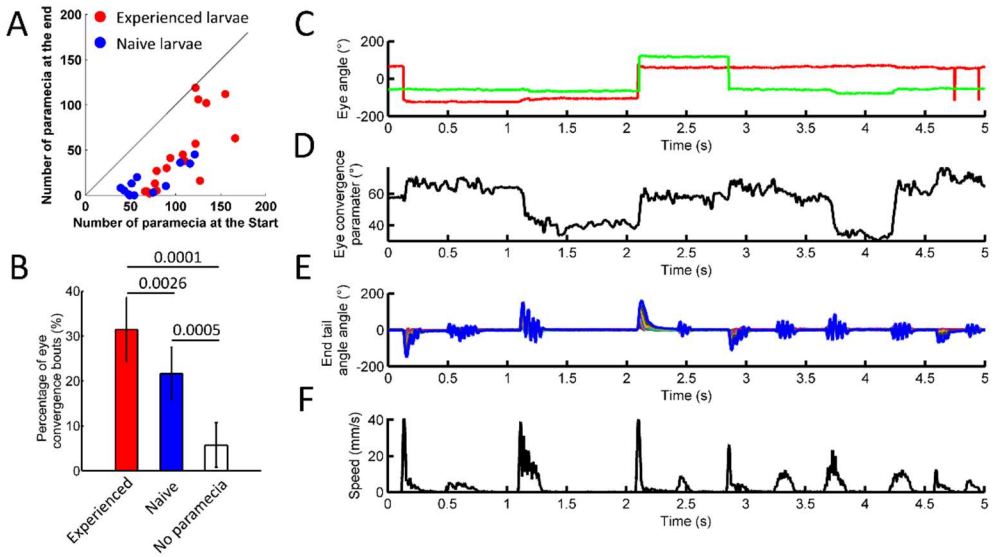

Figure 4.5 – Larvae hunt paramecia using eye convergence sequences. (A) Number of paramecia at the start of the experiment versus number of paramecia at the end. (B) Average percentage of eye convergence bouts for experienced (blue), naïve (red) and without paramecia (white) fish. Error bars are standard deviation and values represent p-values of Mann–Whitney test with Bonferroni-Holm multiple comparisons correction. (C) Angle (°) of left (red) and right (green) eye versus time (s). (D) Eye convergence (°) versus time (s). (E) End tail angle (°) versus time (s). (F) Speed (mm/s) versus time (s).

We analysed two experimental groups: larvae that had been raised with paramecia (experienced larvae) and naïve larvae. Both groups at the end of the experiment had eaten similar amounts of paramecia (Mann–Whitney test, $p = 0.801$, Figure 4.5A). Next we confirmed that larvae, when in the presence of paramecia, converge their

eyes for long periods of time (Figure 4.5C-D) and that these epochs of eye convergence are accompanied by rapid sequences of swim bouts (Figure 4.5E-F).

To confirm that the execution of bouts with the eyes converged is specific to hunting we performed the same experiment in the absence of paramecia and categorized bouts according to the convergence of the eyes (see Experimental Procedures). We observed that, in the absence of paramecia, fish rarely execute bouts with the eyes converged (~6 %), while in the presence of paramecia the number of eye converged bouts raises to 20-30 % (Figure 4.5B). Also we saw a significant increase in the percentage of eye convergence bouts for the experienced fish (Figure 4.5B)

Table 4.4. Data sets with eye tracking.

Behaviour	Eye kinematics	Stimulus type	Stimulus (lux)	Number of bouts	Number of animals
Spontaneous swimming	Eye conv.	Homogenous square of light	1000	1574	11
Spontaneous swimming	Non Eye conv.			25645	
Prey capture naïve	Eye conv.			19694	12
Prey capture naïve	Non Eye conv.			57133	
Prey capture trained	Eye conv.			41708	17
Prey capture trained	Non Eye conv.			73930	

Optomotor Response

Fish, like many animals, are able to stabilize their position relative to the visual world by moving in the direction of whole-field motion stimuli (optomotor response, OMR). For the OMR we probed two sets of stimuli. As described by Orger *et al*, we changed the orientation of moving gratings in relation to the fish and updated their position in closed loop with the lava's position (12) (Figure 4.6A and Table 4.5). Thus, the fish would perceive a moving grating at a fixed angle during a trial

independently of its own movements, effectively a virtual open loop condition, where the fishes' movements do not affect its visual feedback. Fish, when in the presence of laterally moving gratings, turn in the direction of the perceived motion of the stimulus. As can be seen in Figure 4.6B the larvae's swimming trajectories show a strong tuning in relation to the grating orientation.

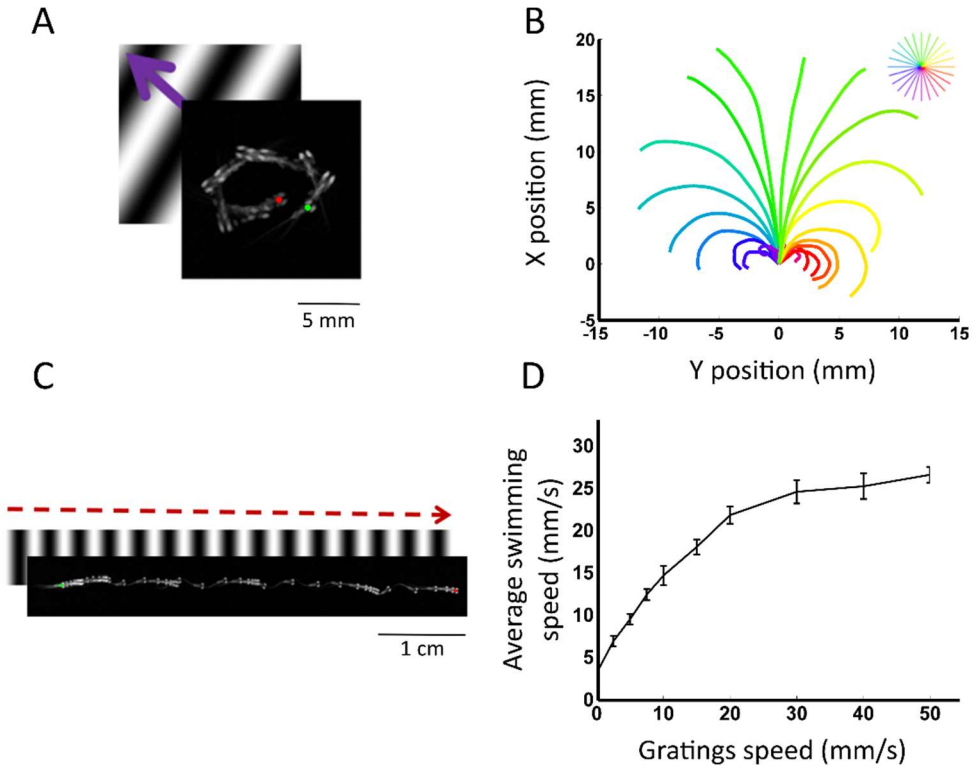

Figure 4.6 – Forward and directional OMR. (A) Overlaid frames of one directional OMR trial (90°). The stimulus in the background represents the grating movement and direction at the beginning of the trial. Points signal the positions of the fish at the start (green) of end (red) of the trial. (B) Average trajectories of 19 fish performing directional OMR. Trajectories are colour coded according to stimulus direction as shown in the key. (C) Overlaid images of one forward OMR trial (50 mm/s). (D) Average swimming speed during trial (mm/s) versus grating speed of 12 fish performing forward OMR. Error bars represent standard error of the mean.

In parallel, we presented forward gratings moving at different speeds to the larvae. This experimental paradigm is similar to the one described in chapter 2, with the exception that in the present experiment the arena is much shorter (5 cm long). Even

when moving in such a small arena larvae were able to match the speed of the gratings for a wide range of velocities (chapter 2 and (19)) (Figure 4.6C-D and Table 4.6).

Table 4.5. Directional OMR.

Stimulus type	Grating angle (°)	Number of bouts	Number of animals
Forward gratings with 1 cm spatial frequency, 10 mm/s of speed and varying angle to the fish.	0	1303	19
	15	1319	
	30	1269	
	45	1419	
	60	1407	
	75	1358	
	90	1125	
	105	929	
	120	763	
	135	723	
	150	646	
	165	553	
	180	613	
	195	643	
	210	672	
	225	760	
	240	828	
	255	1023	
	270	1170	
	285	1313	
	300	1498	
	315	1486	
	330	1497	
	345	1226	

Table 4.6. Forward OMR.

Stimulus type	Grating speed (mm/s)	Number of bouts	Number of animals
Forward gratings with 1 cm spatial frequency and varying speeds.	0	2422	12
	2.5	2135	
	5.0	1804	
	7.5	1515	
	10	1417	
	15	1207	
	20	1166	
	30	1081	
	40	1021	
	50	1045	

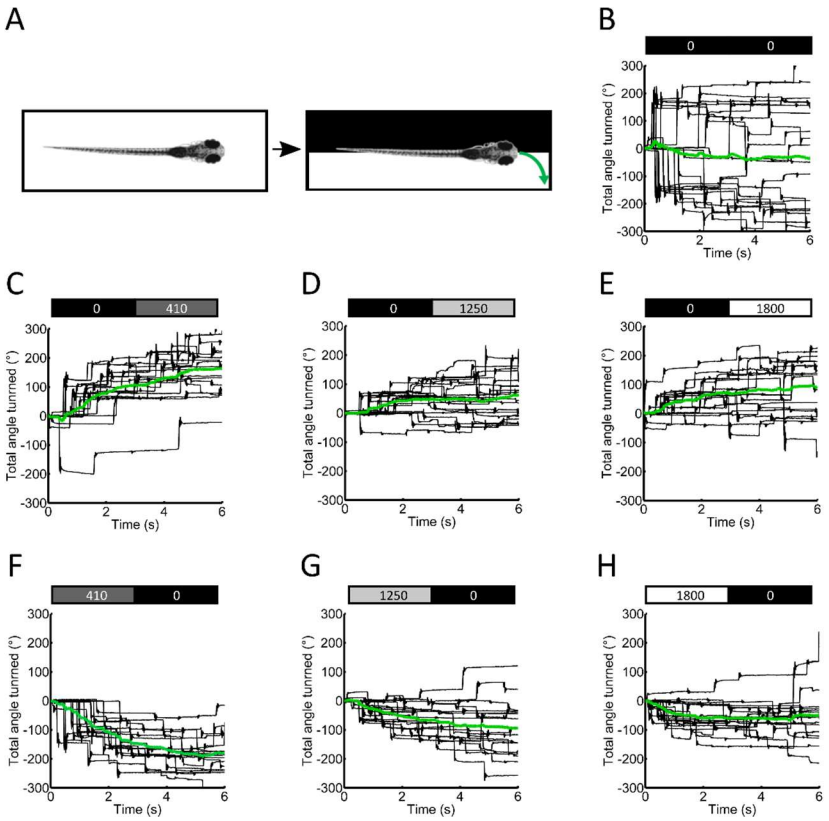

Figure 4.7 – Fish turn towards the light. (A) Phototaxis behavioural paradigm. (B-H) Total angle turned (°) versus time (s). Each plot has a distinct light contrast (top panels state light intensities in lux at the left and right of the fish). Black traces are individual trials. Green lines are averages between 6 fish.

Phototaxis

Larval zebrafish when presented with a gradient of light respond by first turning towards the light and then swimming forward, showing positive taxis for light (20). However, in such a behavioural paradigm, seen from the perspective of the larva, the stimulus is continuously changing through the trial, as the animal approaches the spot of light.

We used a behavioural paradigm based on that used in Huang *et al*, where a left/right illumination contrast with a sharp border at the midline, is dynamically updated so as to remain centered on the fish's body (30). This type of stimulus enables the fish to experience through the trial the same difference of light contrast between left and right and execute sequences of similar movements in response. In our protocol each trial lasts 12 seconds, but we noticed that often the fish would be driven to the edge of the arena before the end of the trial. This is the reason why only the first three bouts of the trial show a strong bias towards the light (data not shown). The trials are separated by periods of 1s of whole field light (Figure 4.7A). We systematically searched for many combinations of this stimulus space (Table 4.). As expected, in average, fish do not show a bias when presented with stimuli that has the same light intensity on both sides (Figure 4.7B). For stimuli where there is a light contrast between right and left fish move towards the side with higher light intensity (Figure 4.7C-H).

Table 4.7. Phototaxis.

Stimulus type [a]	Light intensity on the Left (Lux)	Light intensity on the right (Lux)	Number of bouts [b]	Number of animals
Two light rectangles locked to the fish's position.	1800	0	1118	30
	1250	0	1006	
	780	0	1073	
	410	0	957	
	100	0	977	
	0	0	966	
	1800	100	1222	
	1250	100	1132	
	780	100	970	
	410	100	913	
	100	100	1044	
	0	100	798	
	1800	410	1055	
	1250	410	1163	
	780	410	1134	
	410	410	1122	
	100	410	1116	
	0	410	898	
	1800	780	1209	
	1250	780	1021	
	780	780	1172	
	410	780	1245	
	100	780	1074	
	0	780	929	
	1800	1250	1193	
	1250	1250	1218	
	780	1250	1200	
	410	1250	1185	
	100	1250	1121	
	0	1250	1154	
	1800	1800	1228	
	1250	1800	1337	
	780	1800	1275	
	410	1800	1326	
	100	1800	1241	
	0	1800	1127	

[a] Stimuli were preceded by one second of whole field light at 2600 lux.
[b] The first three bouts after the stimulus onset were taken into account.

Acoustic Startle

Tactile stimuli delivered on the tail or the head (31), or acoustic/vibrational stimuli such as taps or pure tones, produce stereotypical escapes in zebrafish larvae (26). Lacoste *et al* used tactile and acoustic stimuli and showed that the response's kinematics where similar to both types of stimulation (32). Because the acoustic stimulation is easier to control and automate we chose this approach. We delivered pure tones of varied frequencies (200 to 2000 Hz) to larvae in total darkness. Because Mu and colleagues had described that the probability of startling was modified by a proceeding light flash we repeated the same experiment but with the tone preceded by a 150ms flash of light (see Table 4.8) (33). Overall, we did not see a substantial difference between the two experimental conditions, probably because even without the flash of light, larvae showed a probability to startle that was close to 100%, for most of the tone frequencies (Figure 4.8).

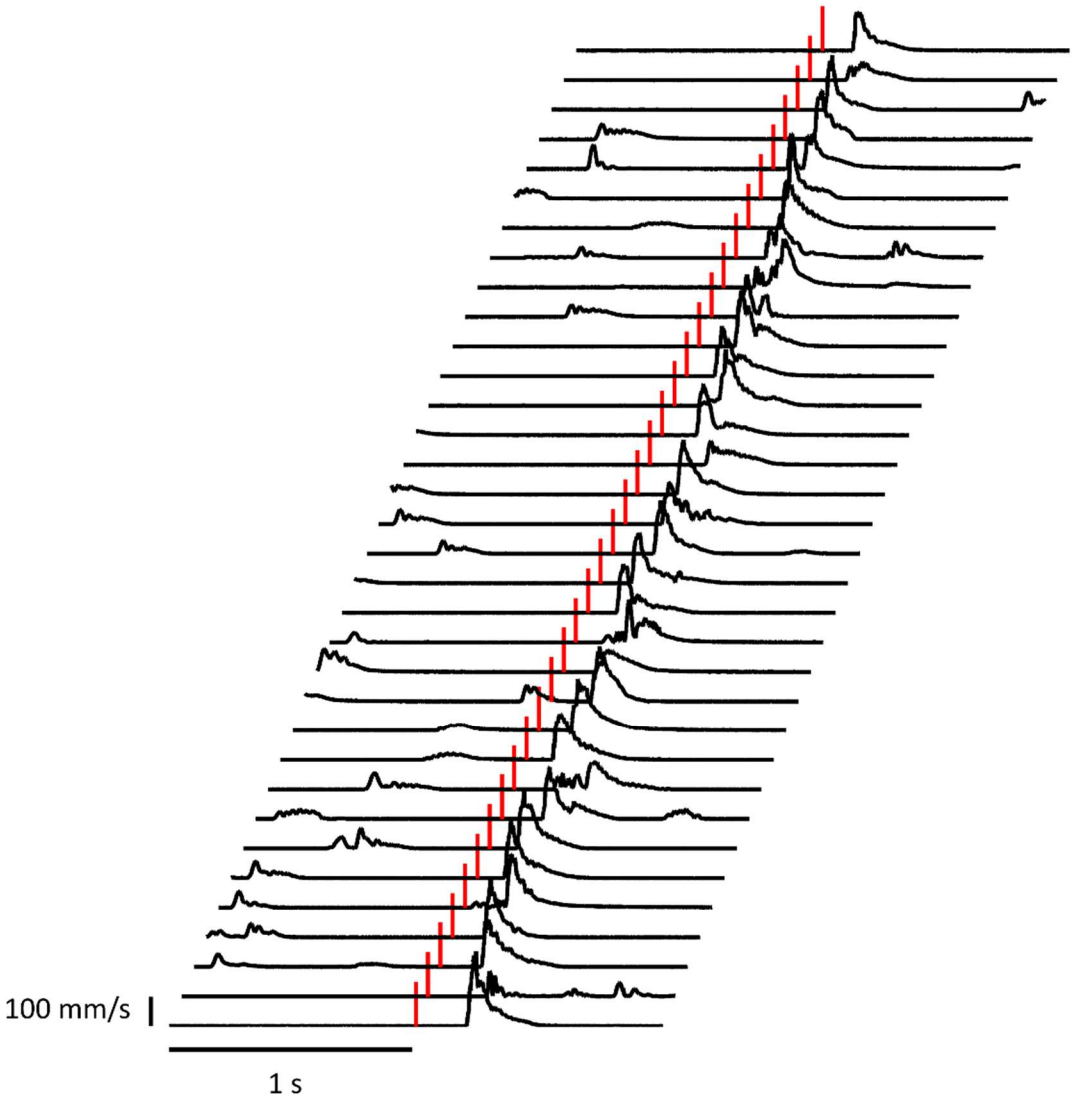

Figure 4.8 – Acoustic startles. Swimming speed (mm/s) versus time (s) of six fish responding to pure tones of 800hz. Each trace is an individual trial and the red line signals the tone delivery.

Table 4.8. Acoustic startle.

Stimulus type [a]	Tone frequency (Hz)	Number of bouts [b]	Number of animals [c]
Pure tone of 100 ms of duration.	200	878	483
	400	802	
	600	816	
	800	819	
	1000	773	
	1200	802	
	1400	747	
	1600	749	
	1800	594	
	2000	566	
Pure tone of 100 ms of duration preceded 400ms by 150 ms light flash with 5000 lux.	200	1231	497
	400	824	
	600	706	
	800	720	
	1000	721	
	1200	730	
	1400	688	
	1600	272	
	1800	586	
	2000	621	

[a] Stimuli are presented every 3 min.
[b] Only the first bout after the tone is considered.
[c] Experiments were done in groups of seven fish.

Expanding Spot Avoidance

Zebrafish larvae also show escape behaviour in response to approaching visual stimuli (24, 27, 34–36). We also performed such experiments by presenting from below spots that would expand at a fixed velocity and updated in closed loop with the fish's position (Figure 4.9A). We observed that fish would execute high velocity movements during the presentation of the expanding spots (black squares in Figure 4.9B). Also, when spots were displayed at different angular locations relative to the larvae, the responses were directed away from the stimuli (Figure 4.9C). As well as the angle that the spot approaches from, we also varied the spots' expansion velocity (Table 4.9) and observed that, for slower expansion velocities, fish perform sequences of bouts against the stimuli (data not shown).

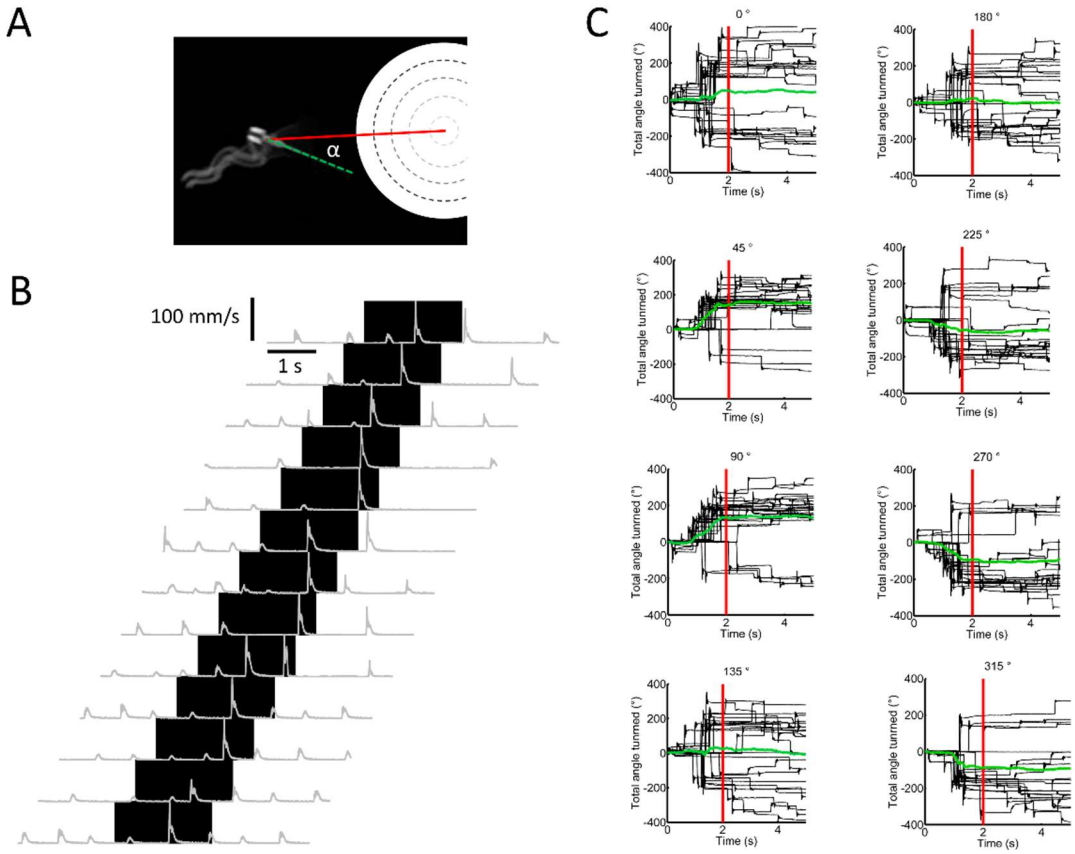

Figure 4.9 – Larvae avoid expanding spots. (A) Overlaid images of a larvae responding to spot expanding 2 mm/s in an angle α with the fish. (B) Speed (mm/s) versus time (s) of 13 larvae responding to a spot expanding from the front at 2 mm/s. (C) Total tail angle (°) versus time (s). In each panel the spots were presented and at different angle to the larvae. Black traces are individual trials; green traces are averages of seven fish.

Table 4.9. Avoidance to expanding spot

Stimulus type [a,b]	Angle of expanding circle (°)	Number of bouts [c]	Number of animals
Circle expanding at 20 mm/s and varying angle to the fish	0	791	95
	45	813	
	90	790	
	135	788	
	180	699	
	225	678	
	270	719	
	315	791	
Stimulus type [a,b]	Speed of expanding circle (mm/s) [d]	Number of bouts [c]	Number of animals
Circle expanding at varying velocities	2.5	5082	25
	5	2199	
	10	941	
	15	605	
	20	440	
	25	361	

[a] Stimuli are repeated every 2 min.
[b] The circle starts expanding 4 cm from the fish at a fixed velocity and is locked online with the larva's position.
[c] We only consider the bouts until the expanding circle reaches the fish.
[d] The expanding circles at different velocities were presented in four directions (0°, 90°, 180°, 270°).

Social Avoidance

We took advantage of the multi-tracking features of our low-resolution set-up to record how larvae interact with each other when in a group. We used groups of seven larvae in a 5 by 5 cm square arena and recorded their movements for 1 to 2 hour while in continuous light (1000 lux) and dark (0 lux). As previously described larvae at this age did not appear to show attraction towards each other (28) , but did show evidence of avoiding when being close to the other fish (6) (Figure 4.10).

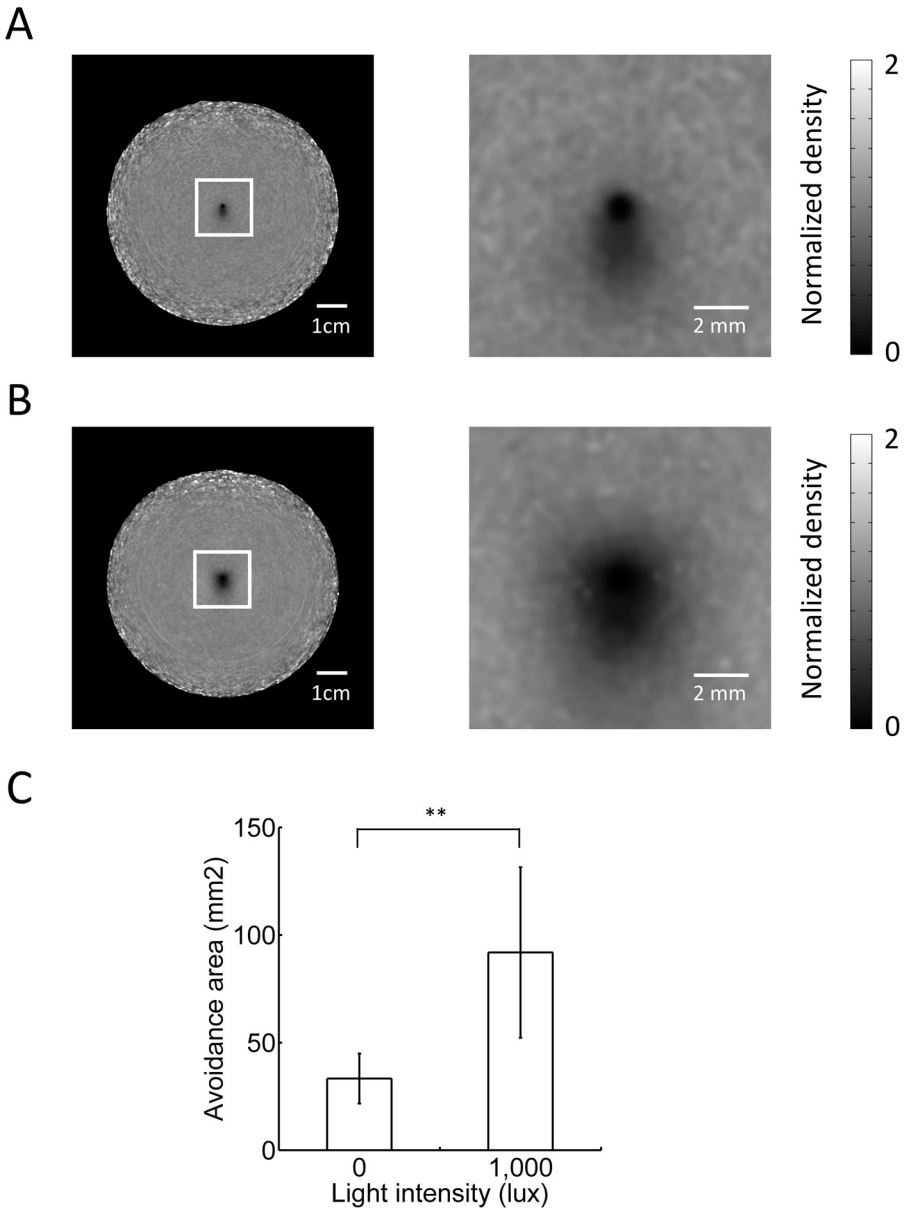

Figure 4.10 – Social avoidance. Position density distributions of seven fish swimming in the dark (A) and the light (B). The focal fish is in the center and facing up. Only the closest fish is considered for each frame. The actual densities are divided to a reference density distribution created by randomizing the focal fish's frame identity. The data is smoothed by gaussian filter with a kernel of 0.6 mm. Left panel is the amplification of the right panel's white square. (C) Average area of the avoidance halo in the light and dark. For every fish the halo area was found by finding the area of the zoomed in image that is 30% below density 1. (Mann–Whitney test, $p < 0.001$, $n = 25$). Error bars are standard errors of the mean.

We computed density distributions of the positions of other fish in relation to a focal fish and observed that larvae possess a "private space" (Figure 4.10A-B). This avoidance halo is both present in the light (Figure 4.10A) and in the dark (Figure 4.10B), however it is significantly larger when larvae interact in the light (Figure 4.10C). This suggests that this behaviour is mediated by at least two sensory modalities, with one of them being vision. We divided the movements that larvae executed in these data sets by selecting bouts where the nearest other fish was in one of four distinct 1 cm^2 areas around the focal fish; left front, right front, left back and right back.

Table 4.10. Social avoidance.

Stimulus type	Stimulus (lux)	Position of closest fish [a]	Number of bouts	Number of animals [b]
Continuous square of light	0	Left, up	5216	28
		Right up	5166	
		Left, down	5313	
		Right, down	5326	
	1000	Left, up	48905	175
		Right up	51578	
		Left, down	51169	
		Right, down	51880	

[a] Bouts that occur when another fish is at least 1cm from focal fish. Area was divided in four positions.
[b] Animals interacted in sets of 7 in a 5 x 5 cm arena.

DISCUSSION

In the past 20 years a large body of literature on the ethology of the zebrafish larvae has identified the "sign stimuli" that release many of the innate behaviours of this vertebrate model. We took advantage of this knowledge and constructed a behavioural system that allows for the measurement of the main moving body parts of the fish (tail and eyes) in the context of nine innate behaviours. With the exception of tactile escapes, these nine innate behaviours constitute the array of larval behaviours which have been associated with specific types of movement. Also, for each behaviour, we systematically sampled the space of stimuli that produce them.

It is known that zebrafish larvae have behavioural responses to other modalities that we did not include in our array of innate behaviours. For example, Haesemeyer and colleagues have described that larvae increase the probability of executing a swim when heated by a IR laser (37); Groneberg *et al* have reported that larvae are attracted to minute water vibrations and that ablation of the lateral line abolishes this behaviour (38); and fish also respond to chemical stimuli (39). While these behaviours are all interesting for future study, they are difficult to automate and standardize in the present version of our behaviour set-ups. If the movement responses produced by these modalities are represented in our stimuli library is an open question.

In spite of that, the nine innate behaviours we tested encompass all the movement responses (called bout types in fish) that have been described. Also, our behavioural set-ups enable recordings of up to nine fish simultaneously for an arbitrary number of hours, not needing any user input during the duration of the experiment. These versatile features make an ideal system to test genetic or pharmacological manipulations; giving a wide description of behavioural phenotypes. In addition, with minimal changes it would be possible to target the fish with light enabling the usage of optogenetic manipulations (14).

In the present chapter we described the fish speed and angle change in responses while performing distinct behaviours. In the next chapter we use these same data sets and focus on the swim bouts that were elicited in the context of these nine innate behaviours and ask if movements cluster into types or are better described by a continuum where clear boundaries do not exist.

EXPERIMENTAL PROCEDURES

Animal care

Fish were reared on a 14/12 hr light/dark cycle at 28 °C on sets of 20 in E3 water according to (40). Wildtype Tubingen zebrafish larvae (6-7 days post fertilization) were used for behavioural experiments. Animal handling and experimental procedures were approved by the Champalimaud Foundation Ethics Committee and the Portuguese Direcção Geral Veterenária and were done according to the European Directive 2010/63/EU.

Behavioural set-ups

The fish's movements were recorded using two custom made behavioural set-ups: a high resolution set-up that enabled the tracking of the tail and eyes of one fish in a 2.5 x 2.5 cm area, and a low resolution set-up that allowed for the tracking of the tail of up to nine fish in a 5 x 5 cm area. Both set-ups are similar except for the working distance of the camera and the machine vision lens that they are fitted: the high resolution set-up uses a Schneider apo-Xenoplan 2.0/35 lens and the low resolution set-up uses a Schneider apo-Xenoplan 2.0/24 lens. In both set-ups a high-speed infrared sensitive camera (MC1362, Mikrotron) is used to image the fish from above at 700 frames per second. Fish were illuminated by a 10 x 10 cm LED-based diffusive backlight (850 nm) placed below the fish arena and the camera was fitted with a 790 nm long pass filter that blocked the passage of visual light. Visual stimuli were projected by a DLP projector (BenQ) onto a 150 mm x 150 mm opal glass diffuser 5 mm below the fish. Acoustic stimuli were delivered by two HP mini speakers (Hewlett-Packard Company) attached to the arena. The conditions of the 255 individual stimuli are summarized on Table 4.1 to 4.10.

Fish tracking and tail and eye segmentation

Acquisition, stimulus presentation, fish tracking and segmentation of the eyes and tail were performed on-line by a custom written programs (C#, Microsoft and OpenCV imaging processing library). For the high resolution set-up, a model of the background is constructed by taking the mode of each pixel over a set of temporally spaced frames and is subtracted from each image (Figure 4.11A). The brightest pixel in the subtracted and inverted image corresponds to one of the eyes of the fish (left panel, Figure 4.11B). The other eye's location was found by finding the center of mass of the pixel values along an arc centred on the first eye. The shape of each eye was found by using their centers to seed a flood fill on a thresholded version of the image (central panel, Figure 4.11B). The orientation of each eye in the horizontal plane was determined by recording the angle between the principal axis of the eye and a line parallel to the midline of the larva (left panel, Figure 4.11B). The middle point between the center of mass of each eye was defined as the larva's location (left panel, Figure 4.11C). The direction of the tail was found by finding the maximum pixel value on a 0.9 mm diameter circle around this point, which corresponded roughly to the position of the swim bladder (central panel, Figure 4.11C). To evaluate tail curvature, we successively computed the angles of seven tail segments 0.30 mm long, by finding the center of mass of the pixel values along an arc centred on the end of the previous segment (left panel, Figure 4.11C).

The tracking of the fish for the low resolution set-up was performed according to (5, 19) . Briefly, the subtracted image (Figure 4.2C) is smoothed by using a boxcar filter, which transforms each larva in the image into a blob (Figure 4.2D). We consider the center of mass of each blob, which falls on a stereotyped location on the larva's body as the position of each fish (green points in Figure 4.2E). Identification of the fishes' orientation and tail tracking is performed as in the high resolution rig (Figure 4.2F). Fish were considered to cross when their positions would occupy the same position as a 0.35 mm square around their head. The frames where fish crossed were discarded.

Behavioural assays

Behavioural recordings were performed in acrylic transparent arenas that varied in size according to the particular set-up that was used and behaviour assay that was performed (Table 4.11).

Table 4.11. Experimental conditions of behavioural assays.

Behaviour	Arena Width (cm)	Arena Height (cm)	Depth (cm)	Number of fish recorded	Behavioural set-up
Spontaneous swimming	5 or 2.5	5 or 2.5	0.3	1	Both
Light Transitions	5 or 2.5	5 or 2.5	0.3	1 or 7	Both
Prey capture	2.5	2.5	0.3	1	High res.
Forward OMR	1	5	0.8	1	Low res.
Directional OMR	5	5	0.3	1	Low res.
Phototaxis	5	5	0.3	1	Low res.
Acoustic startle	5 or 2.5	5 or 2.5	0.3	1 or 7	Both
Spot avoidance	5	5	0.3	1	Low res.
Social avoidance	5	5	0.3	7	Low res.

Behaviours where stimuli were not updated with the fish's position were performed in groups of seven fish (Table 4.11). To confirm that results were not changed by the presence of other fish those behaviours were also performed with a single fish (Table 4.11).

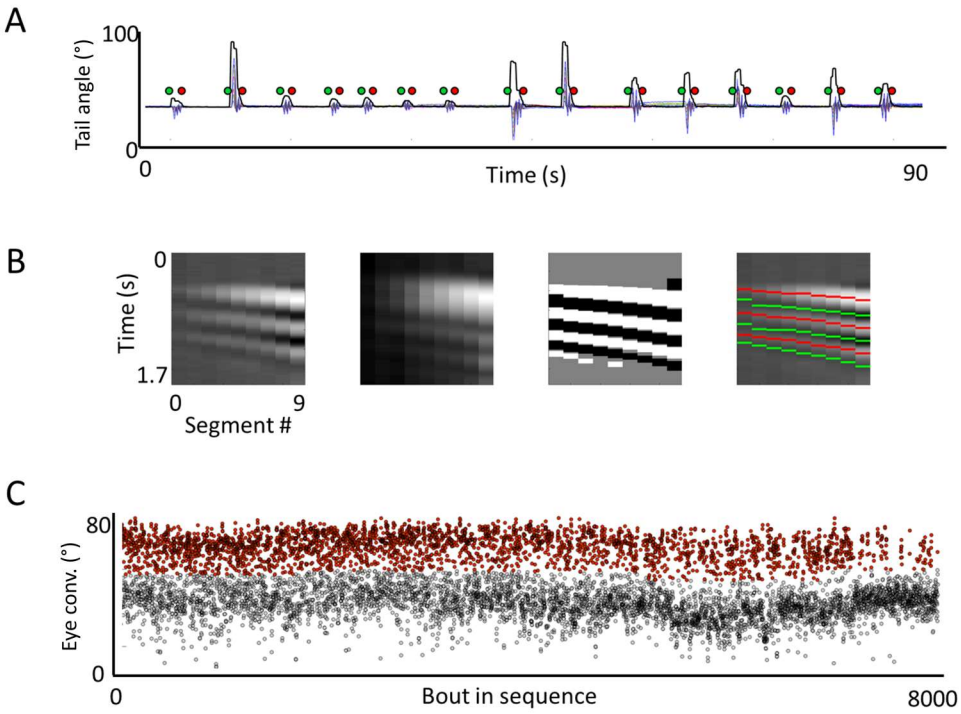

Figure 4.11 – Detection of bouts, half beats and eye convergence bouts. (A) End tail angle (°) versus time (s). Black line is the motion measure used to find the start (green points) and end (red points) of bouts. (B) Algorithm for half beat detection. Tail segment angle versus time of bout. First panel, raw data of tail angles; black negative angles, white positive angles. Second panel, the smoothed image of first panel is used as a threshold to find half beats (third panel). Fourth panel, for each half beat its most extreme value is detected by calculating the center of mass of the objects found in the third panel. (C) Eye convergence (°) versus number of bout in sequence. Red points are eye convergence bouts found automatically by fitting the data in a rolling window with a mixture of two gaussians.

Parameters and numbers of animals for the 255 stimuli that were shown to the fish are presented in Tables 4.1-10. Experiments would last from 30 min to 2 hours, depending on the particular duration of the protocols. OMR directional stimuli were designed according to (12) and OMR forward stimuli as described in (19). The phototaxis paradigm was based on that used in (30). For the looming avoidance data set the expanding circle started 4 cm away from the fish. The spot expanded at a fixed linear rate for each stimuli and its position was updated based on the fishs' location. For this data set we only consider the bouts that are executed from the start of the

stimuli to the moment the spot reaches the fish. For the acoustic startle and light transitions data sets only the first bout after the stimuli onset was considered for further analysis. For the Phototaxis behaviour we considered only the first three bouts performed by the fish after the stimulus onset. In the prey capture assays single fish were fed with 50-100 live paramecia and were allowed to hunt for 1-2h. A subset of the fish was raised with living paramecia from 4 days of age.

Bout detection

To detect bouts reliably, we calculated a measure that would roughly capture the tail's motion. To calculate this value, the raw tail angles were smoothed using a boxcar filter and the frame-to-frame change in tail curvature at each point along the tail was calculated. Next the cumulative sum of the absolute value of this measure along the length of the tail was calculated. To remove the effect of slow drifts in tail position and smooth out fluctuations due to the periodic beating of the tail, a minimum filtered version of this measure in a 571 ms window was first subtracted, and a maximum filter with a 30 ms window was applied. For each fish the 80^{th} percentile of tail motion measure was used as a threshold. This threshold was used on the tail motion measure to find the start and end of bouts (Figure 4.11A).

Half beat detection

To detect half beats we created for each bout a dynamic threshold that changes through time and though tail segments by using a boxcar filter to smooth the original tail curvature values (first and second panel of Figure 4.11B). The beginning and ending of half beats were found by using this dynamic threshold on the tail curvature data (third panel of Figure 4.11B). The most extreme value of each half beat was determined, for each segment, by calculating the maximum value of the tail angle (fourth panel of Figure 4.11B).

Eye kinematics

For every bout where the eyes were tracked, an eye convergence index was determined by calculating the average of the subtracted angles of the two eyes (24). Bouts were categorized in eye convergence and non-eye convergence bouts by fitting the distribution for each fish with a mixture of two gaussians whose parameters were optimized by the expectation maximization algorithm (41) (Figure 4.11C).

REFERENCES

1. Baek J-H, Cosman P, Feng Z, Silver J, Schafer W (2002) Using machine vision to analyze and classify Caenorhabditis elegans behavioral phenotypes quantitatively. *J Neurosci Meth* 118:9–21.

2. Swierczek N, Giles A, Rankin C, Kerr R (2011) High-throughput behavioral analysis in C. elegans. *Nat Methods* 8:592–8.

3. Straw A, Branson K, Neumann T, Dickinson M (2010) Multi-camera real-time three-dimensional tracking of multiple flying animals. *J Royal Soc Interface Royal Soc* 8:395–409.

4. Branson K, Robie A, Bender J, Perona P, Dickinson M (2009) High-throughput ethomics in large groups of Drosophila. *Nat Methods* 6:451–7.

5. Burgess HA, Granato M (2007) Modulation of locomotor activity in larval zebrafish during light adaptation. *J Exp Biol* 210:2526–39.

6. Mirat O, Sternberg JR, Severi KE, Wyart C (2013) ZebraZoom: an automated program for high-throughput behavioral analysis and categorization. *Front Neural Circuits* 7:107.

7. Pérez-Escudero A, Vicente-Page J, Hinz R, Arganda S, Polavieja G (2014) idTracker: tracking individuals in a group by automatic identification of unmarked animals. *Nat Methods* 11:743–8.

8. Dell A et al. (2014) Automated image-based tracking and its application in ecology. *Trends Ecol Evol* 29:417–28.

9. Legradi J, Abdellaoui N, Pomeren M, Legler J (2015) Comparability of behavioural assays using zebrafish larvae to assess neurotoxicity. *Environ Sci Pollut R* 22:16277–16289.

10. Naumann E, Kampff A, Prober D, Schier A, Engert F (2010) Monitoring neural activity with bioluminescence during natural behavior. *Nat Neurosci* 13:513–20.

11. Portugues R, Feierstein CE, Engert F, Orger MB (2014) Whole-Brain Activity Maps Reveal Stereotyped, Distributed Networks for Visuomotor Behavior. *Neuron* 81:1328–1343.

12. Orger MB, Kampff AR, Severi KE, Bollmann JH, Engert F (2008) Control of visually guided behavior by distinct populations of spinal projection neurons. *Nat Neurosci* 11:327–33.

13. Curado S, Stainier D, Anderson R (2008) Nitroreductase-mediated cell/tissue ablation in zebrafish: a spatially and temporally controlled ablation method with applications in developmental and regeneration studies. *Nat Protoc* 3:948–954.

14. Douglass AD, Kraves S, Deisseroth K, Schier AF, Engert F (2008) Escape behavior elicited by single, channelrhodopsin-2-evoked spikes in zebrafish somatosensory neurons. *Curr Biol* 18:1133–7.

15. Wyart C et al. (2009) Optogenetic dissection of a behavioural module in the vertebrate spinal cord. *Nature* 461:407–10.

16. Budick SA, O'Malley DM (2000) Locomotor repertoire of the larval zebrafish: swimming, turning and prey capture. *J Exp Biol* 203:2565–79.

17. Dunn TW et al. (2016) Brain-wide mapping of neural activity controlling zebrafish exploratory locomotion. *Elife* 5.

18. Orger MB, Smear MC, Anstis SM, Baier H (2000) Perception of Fourier and non-Fourier motion by larval zebrafish. *Nat Neurosci* 3:1128–33.

19. Severi KE et al. (2014) Neural control and modulation of swimming speed in the larval zebrafish. *Neuron* 83:692–707.

20. Burgess H, Schoch H, Granato M (2010) Distinct Retinal Pathways Drive Spatial Orientation Behaviors in Zebrafish Navigation. *Curr Biol* 20:381–386.

21. Fernandes A et al. (2012) Deep brain photoreceptors control light-seeking behavior in zebrafish larvae. *Curr Biology Cb* 22:2042–7.

22. Chen X, Engert F (2014) Navigational strategies underlying phototaxis in larval zebrafish. *Front Syst Neurosci* 8:39.

23. Borla MA, Palecek B, Budick S, O'Malley DM (2002) Prey capture by larval zebrafish: evidence for fine axial motor control. *Brain Behav Evol* 60:207–29.

24. Bianco I, Kampff A, Engert F (2011) Prey capture behavior evoked by simple visual stimuli in larval zebrafish. *Front Syst Neurosci* 5:101.

25. O'Malley D, Kao Y-H, Fetcho J (1996) Imaging the Functional Organization of Zebrafish Hindbrain Segments during Escape Behaviors. *Neuron* 17:1145–1155.

26. Burgess HA, Granato M (2007) Sensorimotor gating in larval zebrafish. *J Neurosci* 27:4984–94.

27. Dunn TW et al. (2016) Neural Circuits Underlying Visually Evoked Escapes in Larval Zebrafish. *Neuron* 89:613–28.

28. Dreosti E, Lopes G, Kampff A, Wilson S (2015) Development of social behavior in young zebrafish. *Front Neural Circuits* 9.

29. Portugues R, Engert F (2009) The neural basis of visual behaviors in the larval zebrafish. *Curr Opin Neurobiol* 19:644–7.

30. Huang K-H, Ahrens M, Dunn T, Engert F (2013) Spinal projection neurons control turning behaviors in zebrafish. *Current biology* 23:1566–73.

31. Liu Y-CC, Bailey I, Hale ME (2012) Alternative startle motor patterns and behaviors in the larval zebrafish (Danio rerio). *J Comp Physiol A Neuroethol Sens Neural Behav Physiol* 198:11–24.

32. Lacoste A et al. (2015) A convergent and essential interneuron pathway for Mauthner-cell-mediated escapes. *Curr Biology Cb* 25:1526–34.

33. Mu Y, Li X, Zhang B, Du J (2012) Visual Input Modulates Audiomotor Function via Hypothalamic Dopaminergic Neurons through a Cooperative Mechanism. *Neuron* 75:688–699.

34. Pelkowski S et al. (2011) A novel high-throughput imaging system for automated analyses of avoidance behavior in zebrafish larvae. *Behav Brain Res* 223:135–44.

35. Barker AJ, Baier H (2015) Sensorimotor Decision Making in the Zebrafish Tectum. *Curr Biol* 25:2804–2814.

36. Yao Y et al. (2016) Visual Cue-Discriminative Dopaminergic Control of Visuomotor Transformation and Behavior Selection. *Neuron* 89:598–612.

37. Haesemeyer M, Robson DN, Li JM, Schier AF, Engert F (2015) The Structure and Timescales of Heat Perception in Larval Zebrafish. *Cell Syst* 1:338–348.

38. Groneberg A, Herget U, Ryu S, Marco R (2015) Positive taxis and sustained responsiveness to water motions in larval zebrafish. *Front Neural Circuits* 9.

39. Prober D et al. (2008) Zebrafish TRPA1 Channels Are Required for Chemosensation But Not for Thermosensation or Mechanosensory Hair Cell Function. *J Neurosci* 28:10102–10110.

40. Martins S et al. (2016) Toward an Integrated Zebrafish Health Management Program Supporting Cancer and Neuroscience Research. *Zebrafish*.

41. Krishnan T, McLachlan GJ (2008) *The EM algorithm and extentions* (Wiley-Interscience).

CHAPTER 5 – Unsupervised Behavioural Categorization Reveals New Swim Types of the Zebrafish Larvae.

ABSTRACT

A central question in neuroethology is: what is the set of movements a given animal species is able to perform? Can movements be clustered into different types or are they better described as a continuum where no distinct boundaries exist. Currently it is unknown, for any vertebrate animal, its full repertoire of movements. If they are organized in different classes, this may reflect the underlying organization of the neuronal substrates that control movement.

With the aim of describing the locomotor repertoire of an animal we turned to the zebrafish larva as a model. While this animal possesses the basic architecture of the vertebrate brain, its behaviour consists, for the most part, of simple innate, reflexive swimming and oculomotor behaviours. We recorded millions of larval swim events for a wide range of behaviours and found that, for any stimuli presented, the fishes' movements form clusters for many kinematic parameters. We used the principal components of these parameters to create a space that preserves the structure of the data. We used density valley clustering, a robust, general-purpose clustering algorithm, to identify eleven distinct swim types that zebrafish larvae use. This solution proved to be consistent for a movement space of higher dimensionality. This classification provides a quantitative framework, which can be used to study sequences of movements, to better understand behavioral phenotypes, and to investigate the neural control of behavior.

INTRODUCTION

Animals use sequences of movements to produce complex and adaptive behaviours that enable them to accomplish ethologically relevant goals such as escaping predation, exploring the environment or hunting. Although, these movements are distinct between each other and can be identified by careful human observation (1) it is unknown if they are better described as a continuum where no distinct boundaries exist or if they naturally cluster into distinct types. If distinct classes of movements exist, this may be a reflection of the underlying neural organization of the brain, and thus the knowledge of the movement repertoire of an animal could shed light into the neural circuits that control movement.

Recent studies have used unsupervised categorization methods to find behavioural motifs in many model organisms: *C. elegans* (2) , *Drosophila melanogaster* (3, 4), the zebrafish larva (5) and the mouse (6). However, the solutions obtained by these studies seem to be closer to postures or movemes than to the movements or actions identified in classical ethological studies. In addition, there is the possibility that behavioural motif classes represent the extremes of modulation of a single movement that would become evident by sampling more systematically the stimuli space to which animals respond.

We choose to use the zebrafish larvae model to investigate this question. Contrary to most animals, the larvae of the zebrafish use an intermittent style of locomotion where tail movements (called swim bouts) are always followed by a period of tail stasis (interbouts), enabling the easy detection of individual movement events (7). Besides, these vertebrates are morphologically simple using mostly horizontal deflections of their tail, and movements of their two pectoral fins to swim, as well as rotating their eyes. This enables simple tracking of movement kinematics on large quantities of individuals and with high temporal resolution (8, 9). Also, already by five days after fertilization the larvae of the zebrafish have a diverse repertoire of innate behaviours that enable them to explore the world (10–12), maintain their position in a changing

environment (9, 13), hunt (14, 15), escape from threatening stimuli (7, 16–18), and avoid other fish (8, 19) (for a useful review see (20)).

Here, we describe a novel unsupervised method to categorize movement types, based on kinematic parameter distributions and density valley clustering to infer the number of clusters that exist on the data (see chapter 3). We applied this automatic approach to a collection of millions of swim events that larvae executed while performing nine innate behaviours (see chapter 4). We found that for every stimulus we presented the fishes' movements form two or more clusters in kinematic space. Furthermore, we propose that the larvae's repertoire of movements is composed of eleven distinct types that are used in stereotypical sequences to form behaviours. This classification can be used to study behavioural phenotypes, movement sequences, and the neural basis of behaviour.

RESULTS

Zebrafish Larvae Movements Form Clusters in Kinematic Space

Zebrafish larvae utilize a distinct gait to move at fast velocities (10, 21, 22). We were able to detect this movement type, in the forward OMR behaviour, because several kinematic parameters showed non-continuous distributions that naturally formed clusters (see Chapter 2 and (9)).

So we wondered if bout kinematic distributions would also form clusters for other innate behaviours of the zebrafish larvae and if unbiased identification of these clusters could be used to find additional swim types that these animals use.

To answer this question, we used a large collection of swims gathered in the context of nine innate behaviours that larvae reliable exhibit (see chapter 4). Overall this collection of swims is composed of 1794076 bouts of 3204 fish that were actively engaged in one of the following behaviours: spontaneous swimming (7, 23), forward OMR (9), directional OMR (24), phototaxis (25), responses to light transitions (7),

avoidance of expanding spots (18), prey capture (14), acoustic startle (17) and social avoidance (8).

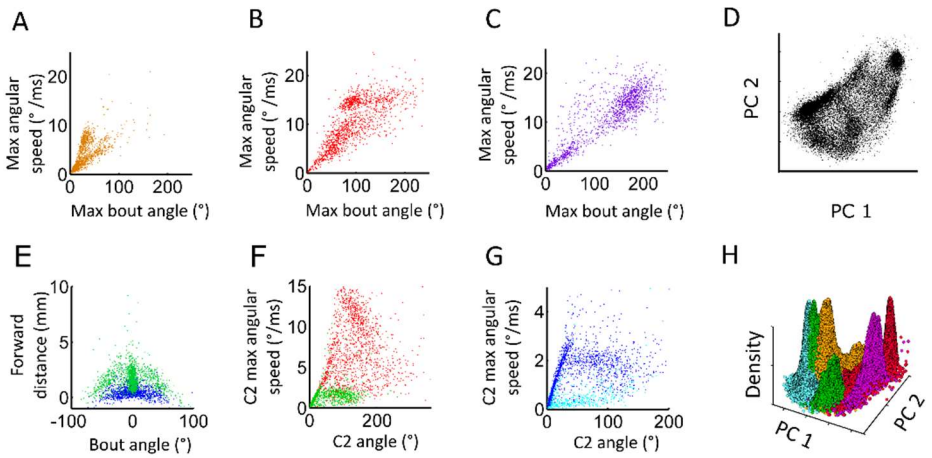

Figure 5.1 – Bouts form clusters in principal components of kinematic space. Colours represent different behaviours: forward OMR (orange), acoustic startle (red), response to dark onset (purple), directional OMR (green), spontaneous swimming (blue), and prey capture (cyan). (A-C) Max bout angle (°) versus Max angular speed (°/ms). (E) Angle at end of bout (°) versus forward distance (mm). (F-G) C2 angle (°) versus C2 max. angular speed (°/ms). (D) Two first principal components of 73 kinematic parameters of bouts performed on the behaviours present in A-C and E-G. (H) Local densities of data in D.

We observed that all of the 255 stimuli we presented to the larvae (Tables 4.1 – 10 in chapter 4) showed kinematic distributions with at least two clusters (Figure 5.1). Different bout clusters are distinguished by differences in a distinct set of kinematic parameters (compare Figure 5.1A-C to Figure 5.1E-G) and it is not obvious which parameters would be most useful to characterize specific types of movements. Therefore, we calculated a large set of kinematic parameters based on tail and head movements of the fish (Tables 5.1-2) and performed PCA on all the bouts that were collected. If we normalize the number of bouts per behaviour and calculate the local density for each behaviour, we can observe that the clustered structure in the data is preserved in the first components of PC space (Figure 5.1D-5.1H). As a starting approach we considered the first three principal components (~70 per cent of the

variability) to form a common movement space where structure is preserved and similar swims occupy close by positions.

Density Valley Clustering of Stimuli Data Sets

Since structure in movement space is such a prominent feature for all the larval behaviours that were tested, we wondered if we could use an unsupervised clustering method to find the natural groups of bouts that are present on the data. Such an approach could permit the finding of novel swim types.

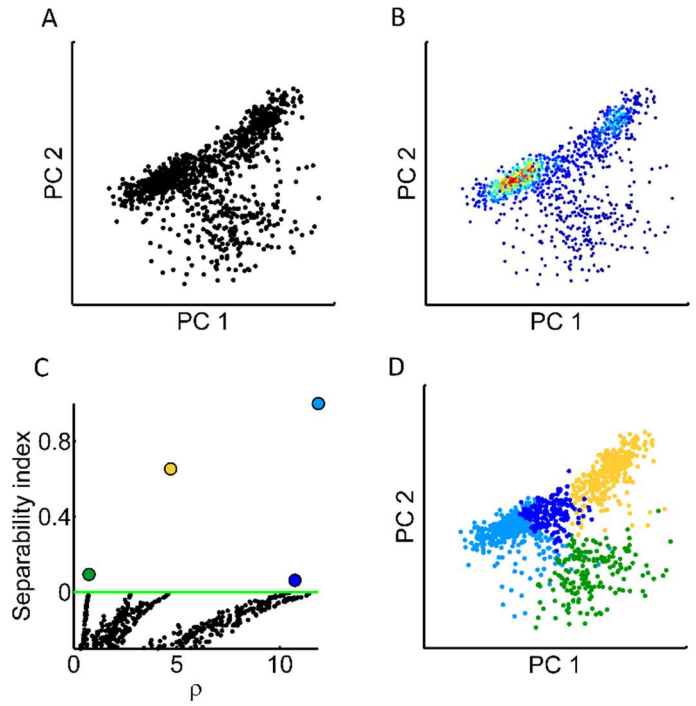

Figure 5.2 – Density valley clustering applied on an example data set. (A) The first two dimensions of the movement space for the OMR speed 50 mm/s data set. (B) Local densities of data in A calculated by adaptive gaussian method (33). (C) Decision plot for data in A. Green line is the CI onion threshold. Colour points are cluster centers found automatically. (D) Final assignment of data in A.

We applied density valley clustering (see Chapter 3) to the bout data embedded on the three dimensional movement space and calculated the number of clusters and their

boundaries for each stimulus that we presented to the larvae (see Experimental Procedures). As can be seen for an example data set in Figure 5.2 (forward OMR gratings at 50 mm/s), bouts form groups that exist in distinct levels of density and separation (Figure 5.2 A-B), but the heuristic of the density valley clustering is able to capture automatically the major clusters that are present on the data (Figure 5.2 C-D).

In total the clustering of the data set associated with every stimulus condition yielded a collection of 372 groups of swims obtained free from human intervention or any parameter optimization.

Ensemble Density Valley Clustering Reveals Eleven Swim Types

It is possible that many of the groups of swims that were obtained from our clustering approach are similar between each other and could be considered to be movements of the same "type". If that is the case it is expected that corresponding groups of swims are present in similar stimuli and occupy close by places in the movement space. To test this, we looked into the clustering solutions of the acoustic startle behaviour, because it was previously shown that fish would startle to pure tones using two bout types: short latency C-starts (here called C-starts) and long latency C-starts (LLC) (17). As can be seen in Figure 5.3 the algorithm found two clusters for most of the pure tones between 200 and 1800 Hz. Furthermore, clusters obtained from different data sets occupy similar positions on the movement space (see red clusters in Figure 5.3). Because the movement space is common for all stimuli, the bouts of these clusters must have similar kinematic values. Thus, it is conceivable that groups of swims from different data sets can represent instances of the same type of movement produced as a response to different stimuli.

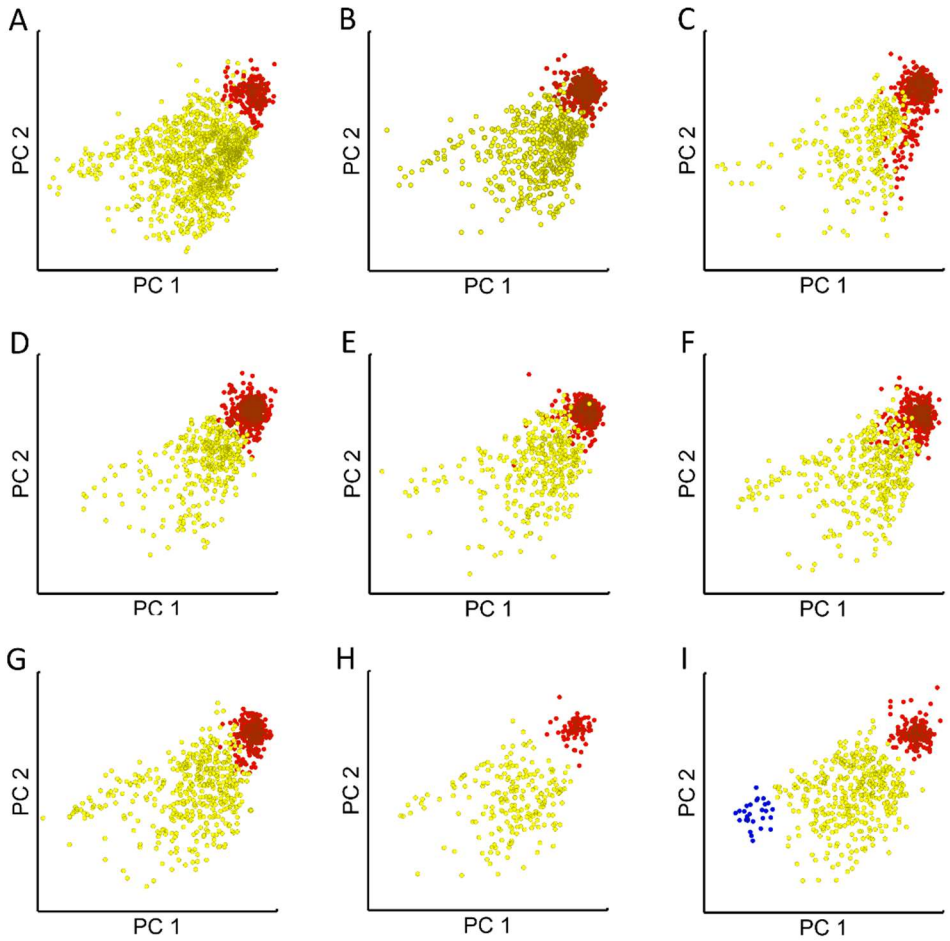

Figure 5.3 – Density valley clustering produces reliable solutions. Similar clusters (red and yellow points) are found for the different acoustic startle data sets. Each plot has one of the data sets where the frequency of the tone varies: 200Hz (A), 400Hz (B), 600 Hz (C), 800 Hz (D), 1000 Hz (E), 1200 Hz (F), 1400 Hz (G), 1600 Hz (H) and 1800 Hz (I).

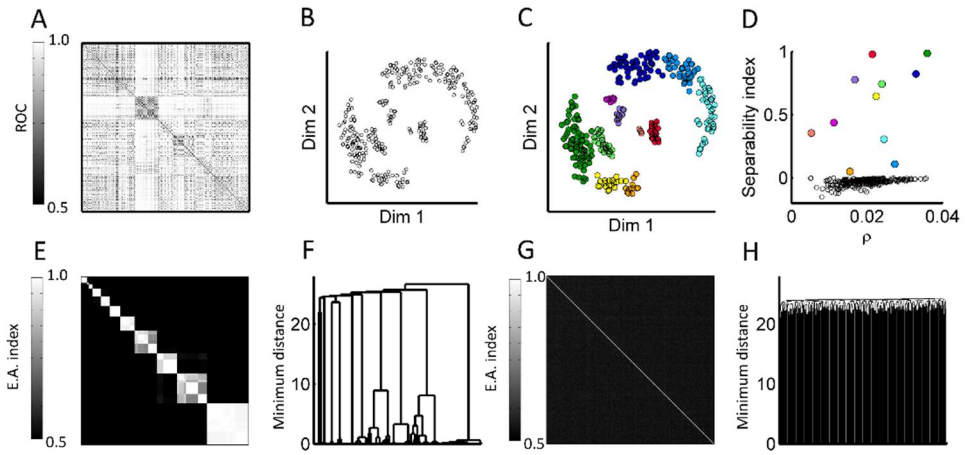

Figure 5.4 – Ensemble density valley clustering. (A) Pairwise ROC values for groups of swims. (B) Example of t-SNE space for data in A. (C) Assignment by density valley clustering of data in B. (D) Decision plot of data in B. (E) Evidence accumulation matrix of 200 t-SNE solutions solved by density valley clustering. The matrix is ordered by single link clustering. (F) Dendrogram of E. (G) Voting matrix of shuffled data. (H) Dendrogram of G.

We reasoned that a way to find the repertoire of swim types that fish used through the nine innate behaviours is to calculate the number of distinct clusters that exist in the collection of groups of swims. In order to perform this clustering step the groups of swims must be placed in a space where similar groups are close together while distinct groups are far apart.

To create such a space, we generated a measure of pairwise cluster similarity based on the receiver operating characteristic (ROC, see Experimental Procedures) (Figure 5.4A). Because pairwise distances are not sufficient for the density valley clustering, which requires an estimate of local point density, we embedded the points in a three dimensional t-SNE space based on the ROC similarity measure (Figure 5.4B) and performed automatic clustering on this new space (Figure 5.4C-D) (26).

t-SNE is a dimensionality reduction technique that has a non-convex objective function that is initiated randomly, so every time it runs it gives a different solution (26). To find the most common solution, and demonstrate that the clusters were partitioned in a consistent way, we used the density valley clustering to partition an

ensemble of 200 ROC based t-SNE spaces and used a voting system to compute the probability that data points belong to the same group (27). After, the voting matrix was reordered using single link clustering (28). As a result, we obtained a clear and consistent solution of 11 groups of clusters, as can be seen by the reordered voting matrix and the respective dendrogram (Figure 5.4E-F). To confirm that this result is not due to the sporadic arrangement of the data we did the same operations on a voting matrix where the cluster identity was randomized and recovered no discernible structure (Figure G-H).

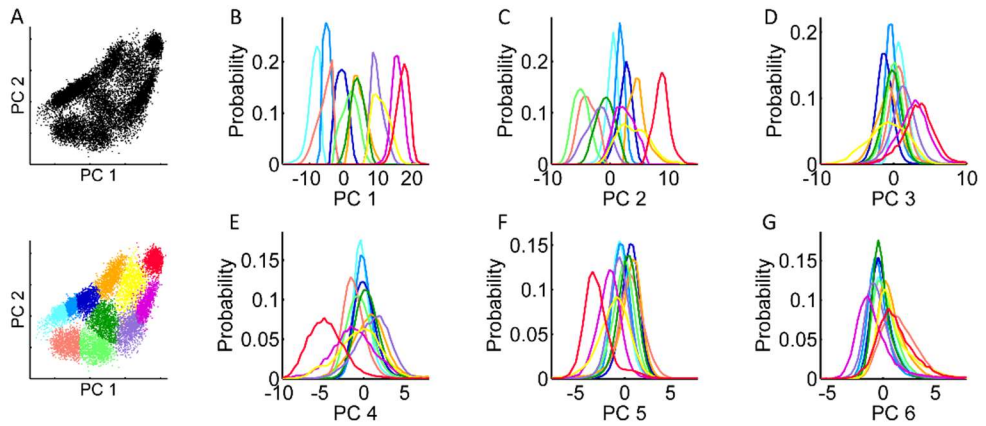

Figure 5.5 – Principal component distributions of the eleven bout types. (A) Bout map that was used to categorize all the data. Upper panel: data without being categorized. Lower panel: assigned bout map. (B-G) Colours represent same categories as in lower panel A. Bout distributions of the weights of the first principal component (B), second principal component (C), third principal component (D), fourth principal component (E), fifth principal component (F), and sixth principal component (G).

We propose that the eleven groups of clusters that we obtained from the ensemble density valley clustering correspond to the swim types that larvae used to respond to all the stimuli we showed them on our experiment (see Table 4.1-10 chapter 4). So, using equal numbers of bouts from each of the eleven categories we created a bout map and used K-nearest neighbour algorithm to label every bout that exists in our full data sets (Figure 5.5A).

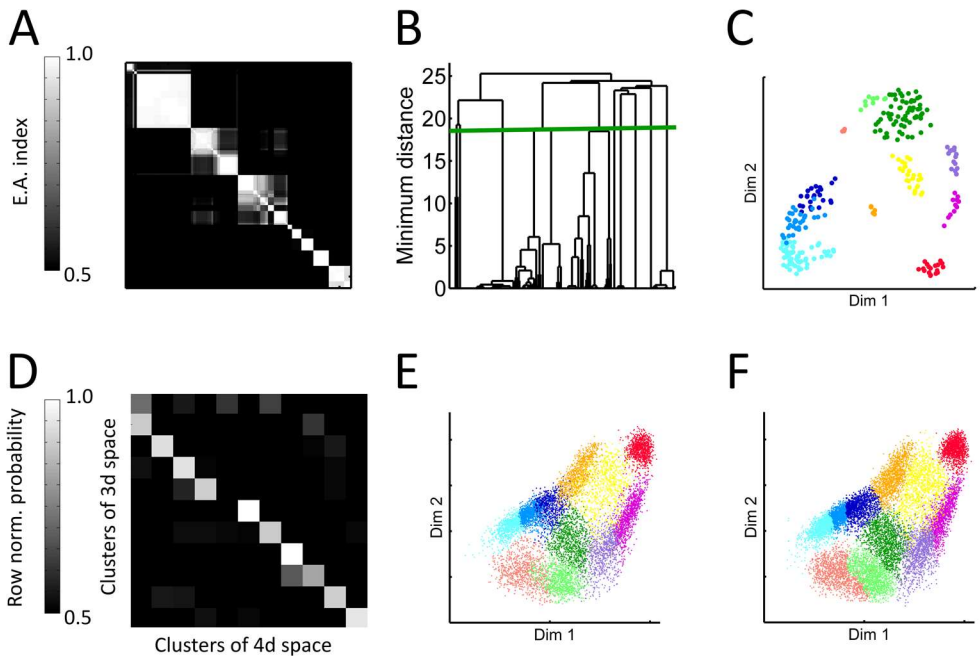

Figure 5.6 – Clustering solution for four-dimension movement space. (A) Evidence accumulation matrix of 200 t-SNE solutions solved by density valley clustering for movement space with 4 PCs. (B) Dendrogram of A. Green line is where the dendrogram was cut to find eleven bout types. (C) Example of t-SNE space categorized with the threshold in B. (D) Row-normalized confusion matrix comparing the solution found in 3d space and 4d space, applied to the full data set. (E) Bout map in first 2 dimensions of 3d movement space. (F) Bout map for the 4d space.

The movement space we used for our clustering analysis is based on the first three principal components obtained from performing PCA on 73 kinematic parameters. It may be that other principal components would form subspaces where new clusters could emerge. If this were the case, we would expect that the bout type distributions for other principal components would have a multimodal shape. We plotted for each bout type the histogram of the first six principal components and observed that all the distributions showed a unimodal shape (Figure 5.5B-G). We see this as an indication that further increasing the dimensionality of the data would not further unravel more bout types.

However, it remains a possibility that additional clusters could exist that separate along a direction that is not strongly aligned with a particular PC direction. To test

this hypothesis, we repeated the previous analysis (Figure 5.4) using the movement space based in all kinematic parameters, but including 4 principal components. We observe that using a higher dimensional movement space gives clustering solutions that are compatible with the 11 bout types (green line Figure 5.6A-B), although the dendrogram under these conditions does not show a clearly distinct largest jump, unlike the dendrogram obtained using just 3 principal components (compare Figure 5.4F to left panels of Figure 5.6B), making the selection of the preferred solution less clear. In spite of that, if one looks at the solutions of the t-SNE spaces that were produced for the 4 dimensional movement space it is evident that the eleven bout type solution is capturing the major clusters that are present in the data (for one example see Figure 5.6C). It is possible that although the same number of clusters are found for both movement spaces there is a poor correspondence between the assigned points to each cluster, indicating that in practice the two movement spaces produced different results. To test this, we assigned all the data using both 3D and 4D solutions and calculated the confusion matrix between the two outputs. For most of the clusters there is a very high correspondence between the two classifications (Figure 5.6D). The high correspondence between both solutions is also present at the level of the bout maps, where it can be clearly seen that the eleven clusters occupy roughly the same places on both movement spaces (compare Figure 5.6E-F)).

Bout Types Show Distinct Movement Shapes That Are Stereotypical Within Group

Next, we asked how similar are the tail movements for bouts within each type. To determine “the most typical bout” of each type we used the bout map and found the closest bout to the center of mass of each cluster (black circles in Figure 5.7A). After, we plotted the angle relative to the body of each tail segment versus time, and observed that every bout type consists of a distinctive sequence of tail movements (Figure 5.7B-L).

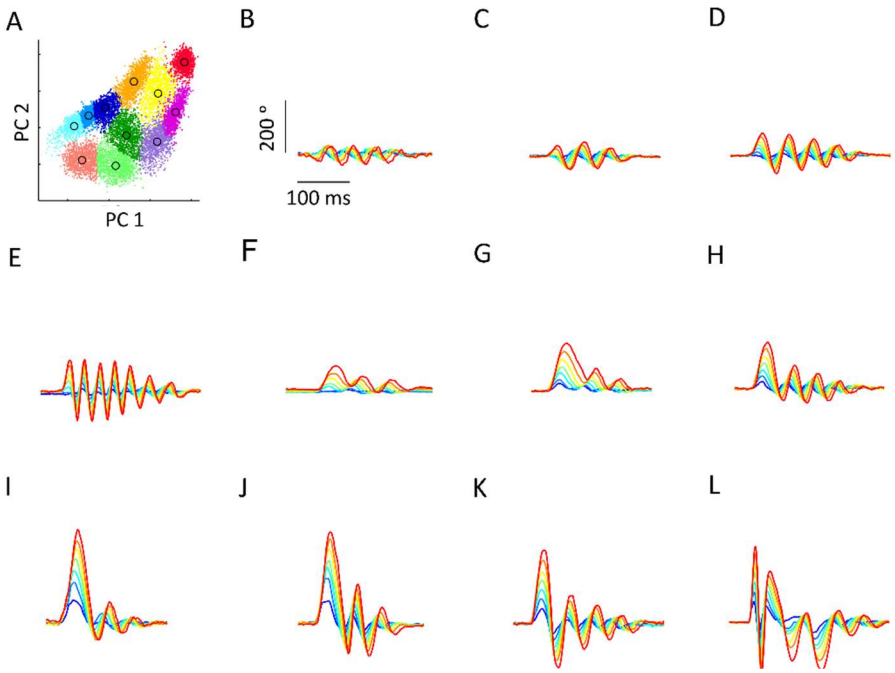

Figure 5.7 – Bout types show distinct tail movements. (A) Principal component 1 versus principal component 2 of bout map that was used to categorize all the data. Black circle mark bout closest to center of mass of each cluster. (B-L) Cumulative sum of tail segment (°) versus time (ms). (B) Approach swim. (C) Slow swim 1. (D) Slow swim 2. (E) Burst swim. (F) J-turn. (G) High angle turn. (H) Routine turn. (I) Shadow avoidance response. (J) O-bend. (K) Long latency C-start. (L) C start.

Is the central bout representative of the cluster? In other words, within category, are the bouts stereotypical? To answer this question, we picked fifty random bouts of each type and plotted the last segment's cumulative tail angle on top of each other (Figure 5.8). For some bout types, most examples show very similar dynamics throughout to the centre of mass bout, while others show very stereotyped patterns for at least the first 2-3 half beats (compare Figure 5.6 to Figure 5.8).

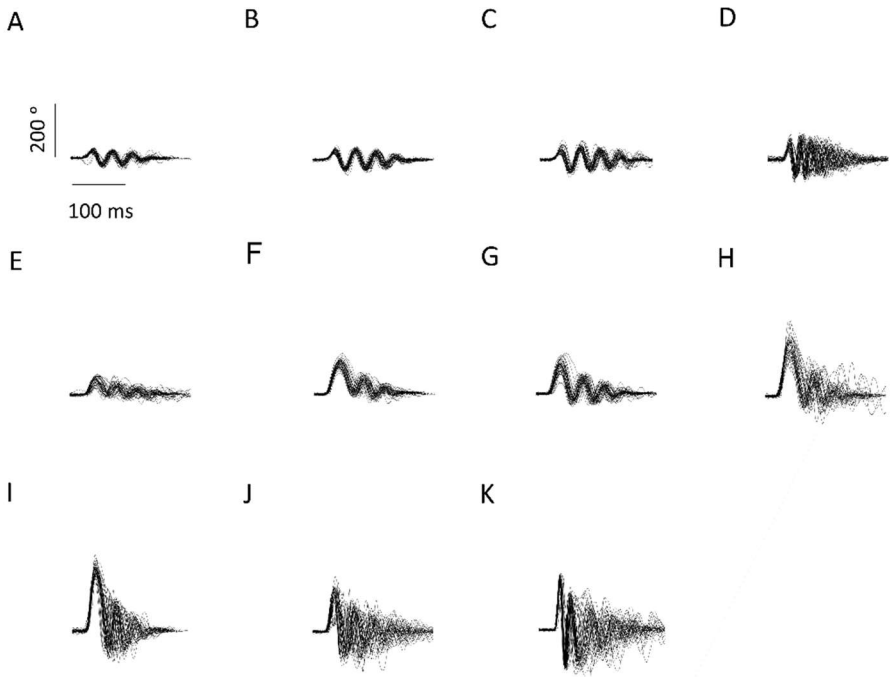

Figure 5.8– Bout are stereotypical within type. (A-K) Cumulative sum of last segment (°) versus time (ms) of fifty randomly picked bout. (A) Approach swims. (B) Slow swims 1. (C) Slow swims 2. (D) Burst swims. (E) J-turns. (F) High angle turns. (G) Routine turns. (H) Spot avoidance responses. (I) O-bend. (J) Long latency C-starts. (K) C-starts.

Having established that the bout types are different between each other and stereotypical within category we wondered what is the pattern of recruitment of bout types for each stimulus we presented to the fish. To understand if larvae modulate the type of bout within each behaviour we ordered the stimuli by increasing strength and computed the probability of each bout type for every stimulus (Figure 5.9). Overall we can observe that there are no stimuli that elicit all 11 bout types. Also, with the exception of dimming stimuli, all other stimuli produce a combination of two or more bout types. Also, in many of the behaviours, namely: spontaneous swimming at different light intensities, forward OMR, directional OMR, phototaxis, acoustic startle and, expanding spots at different speeds, we observe that the probabilities of bout types change smoothly as the relevant stimulus parameter is varied. As we previous demonstrated in the context of speed control, this modulation of bout type probabilities by stimuli provides an example of how swim bout classification makes

it possible to understand better the underlying mechanisms behind changes in the zebrafish's motor behaviour.

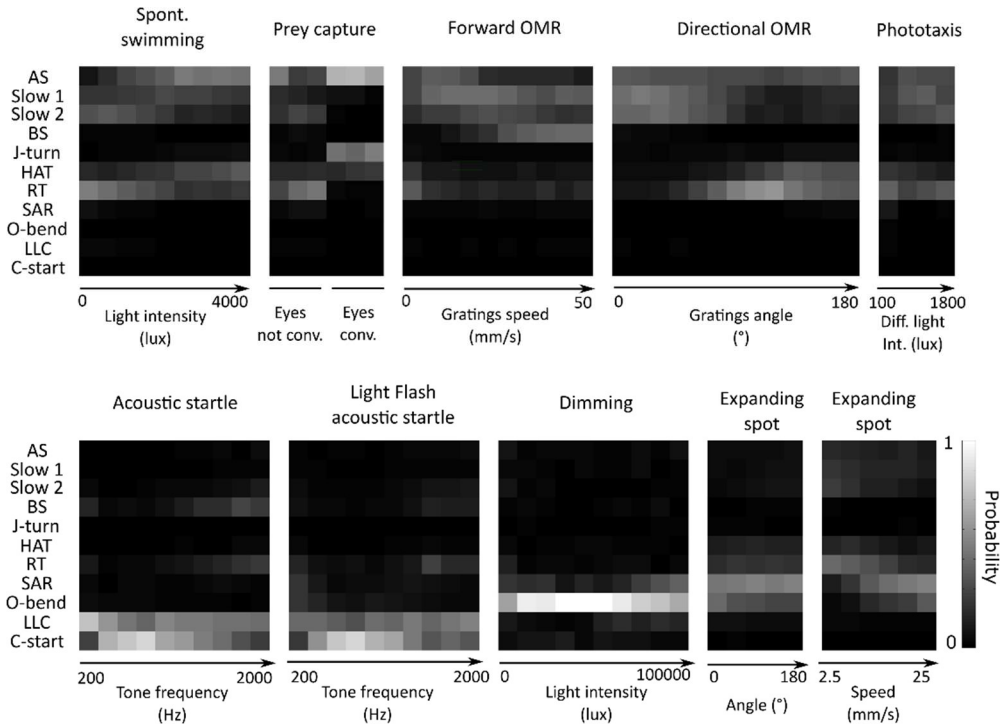

Figure 5.9 – Probability of bout types per stimuli. Stimuli are organized by behaviour and ordered in increasing strength. On the top of each panel it is labelled the behaviour name and in the bottom it is labelled the stimuli and that parameter that was changed.

Finally, we gave a name to each bout type. We started by finding which of the eleven categories we obtained had already been reported. To do that, we considered that a bout category that had similar kinematic parameters to the ones previously reported would be the same type of movement and gave it the name that is common in the literature. Of the eleven bout types we obtained in our data sets, seven fulfilled this criteria, namely: slow swims (10), burst swims (10), J-turns (14), routine turns (10), O-bends (7), long latency C-starts (17) and (short latency) C-starts (17). There were four bout types that did not, in our estimation, clearly map onto types previously reported in the literature and we gave them names of our own choosing.

One of the bout types we characterized is a small forward bout type (Figure 5.6B) that fish use during spontaneous swimming in the light and during prey capture, so we called it approach swim (Figure 5.9). Our categorization revealed two types of slow swims (Figure 5.6C-D), so we differentiated them by calling them type 1 and type 2. Another uncharacterized type of swim is a turn like movement that occurs spontaneously in the light and during orienting OMR gratings of large angles (Figure 5.9). We named this bout type, the high angle turn (HAT), because it is characterized by making movements with large lateral displacements and very small forward displacements (Figure 5.6G). The fish avoided expanding spots with a particular movement type that is similar to O-bends but formed their own clusters, so we called them spot avoidance responses (SAR) (Figure 5.6I).

In summary, the eleven bout types are stereotypical within groups and are used by the fish to interact with different stimuli in a highly organized fashion. Four of the bout types that were detected by our analysis were previously unknown.

Novel Swim Types Can Be Detected Without Using Density Valley Clustering

There is the possibility that the new bout types are the result of some artefact from the density estimation or the clustering procedure. If this were the case, we should not be able to detect them without using our clustering analysis. To test this, we went to the stimuli where they were present with high probabilities and plotted contour plots of the area of the movement space where they should be present. Here we use contour plots as an independent method to the density valley clustering to observe local density and detect the clusters that are present on the data.

It can be observed in the “spontaneous swimming in the dark” collection of stimuli that there are, for the most part, two clusters present in the region of the slow swims (Figure 5.10 A). These clusters correspond to slow swims type 1 and slow swims type 2. On the spontaneous swimming in the light, the two “darker stimuli” are quite similar to the spontaneous swimming in the dark stimuli (two first panels of Figure

5.10 C). However, all the brighter stimuli have one group of movements that occupy a lower and more rightwards position than the slow swims (compare Figure 5.10 A with Figure 5.10 C) that is very similar to the position of the cluster that fish produce when hunting and converging the eyes (compare Figure 5.10 C with Figure 5.10 D). We see this as evidence that these clusters correspond to the approach swims and are distinct to the two types of slow swims that are present in the spontaneous swimming in the dark. In addition, we can observe all the three types of clusters when fish are in the presence of paramecia but not converging their eyes (Figure 5.10 B).

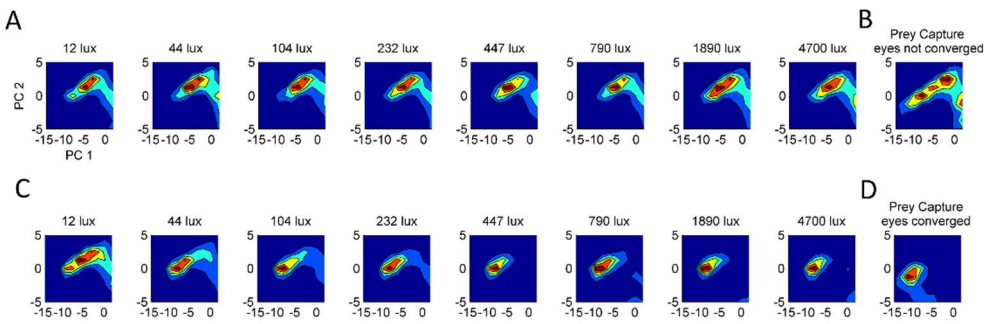

Figure 5.10 – Approach swims, slow swims 1 and slow swims 2 form distinct clusters. (A-D) Movement space contour plots centred on the slow swims area. (A) Spontaneous swimming in the dark stimuli. Above each panel is the light intensity of the preceding light stimulus. (B) Prey capture bouts when the larva's eyes are not converged. (C) Spontaneous swimming in the light. On top of the panels is the light intensity of the stimuli. (D) Prey capture bouts when the larva's eyes are converged.

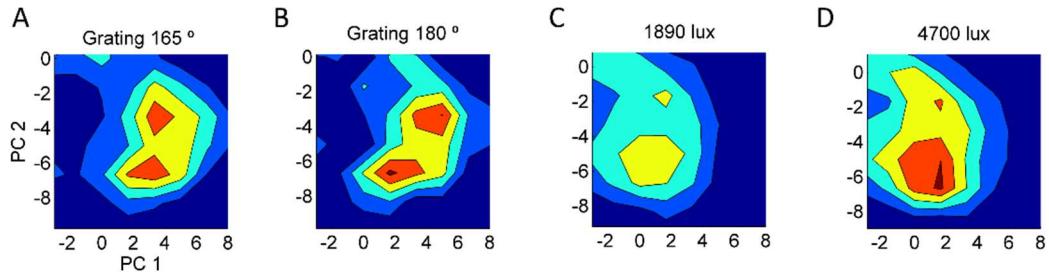

Figure 5.11 – HAT and routine turns show distinct clusters. (A-D) Movement space contour plots centred on the turns area. (A-B) Directional OMR stimuli. (C-D) Spontaneous swimming in the light stimuli.

We have no stimulus on our collection where the fish perform only high angle, but not routine turns (Figure 5.9). So the question for these bout types is if there are

stimuli that show two clusters that correspond to each bout type. To assess this, we choose the stimuli with the most extreme angles of the directional OMR behaviour (Figure 5.11A-B) and the brightest stimuli of the spontaneous swimming (Figure 5.11 C-D). If we focus on the area of the movement space where turns are present we can detect for both behaviours two density peaks on the contour plots. The lower peak corresponds to HATs and the higher peak to routine turns.

For the expanding spots data sets our categorization detects a mixture of O-bends (that are present in a pure form for dimming stimuli) and SARs (Figure 5.9). We plotted the contour plots of these stimuli and focused on the area of the SARs/O-bends (Figure 5.12). For the expanding spot stimuli, it can be seen two clusters; a lower one that corresponds to the SARs, and a higher one that is very similar to O-bends (compare Figure 5.12A-E to Figure 5.12F).

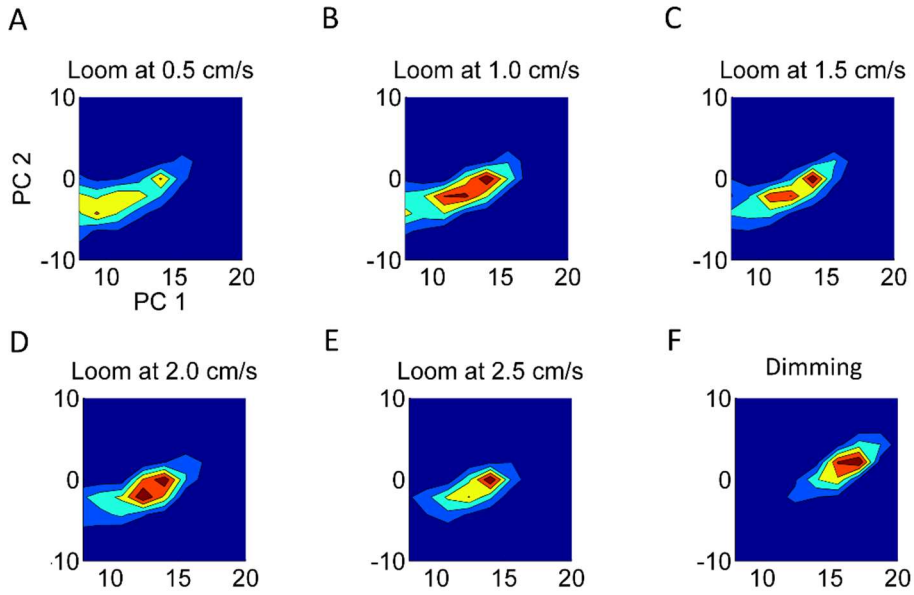

Figure 5.12 –SARs and O-bends show distinct clusters. (A-F) Movement space contour plots centred on the O-bends/SAR area. (A-D) Expanding spot at different speeds data sets. (F) Example dimming stimuli.

Meaning of the Movement space

The eleven bout types were found by performing clustering analysis in a movement space that was created by performing PCA on kinematic parameters (Tables 5.1-2). The movement space is composed of three PC axes that are composed of distinct loadings of the 73 kinematic parameters. Every bout exists in this space, so we wondered about the kinematic parameter composition of the axes of this movement space. Are the major contributors consistent within each axis? We found that within each axis the major loadings are kinematic parameters whose distributions for the different bout types show very similar patterns (Figure 5.13). For the first PC axis the three largest contributions are from kinematic parameters that capture angular velocity (Figure 5.13A); for the second PC axis it is maximum bend amplitudes at different points along the tail during the second half beat (Figure 5.13B); finally, the third PC gets its major contributions from parameters which reflect the speed and distance the fish moves, either laterally, or forward (Figure 5.13C). This consistency within each PC allows us to attach an approximate meaning to each direction in the movement kinematic space. The other PCs that were not included in the movement space are composed of mixtures of types of kinematic parameters, which do not lend themselves to a simple interpretation (data not shown).

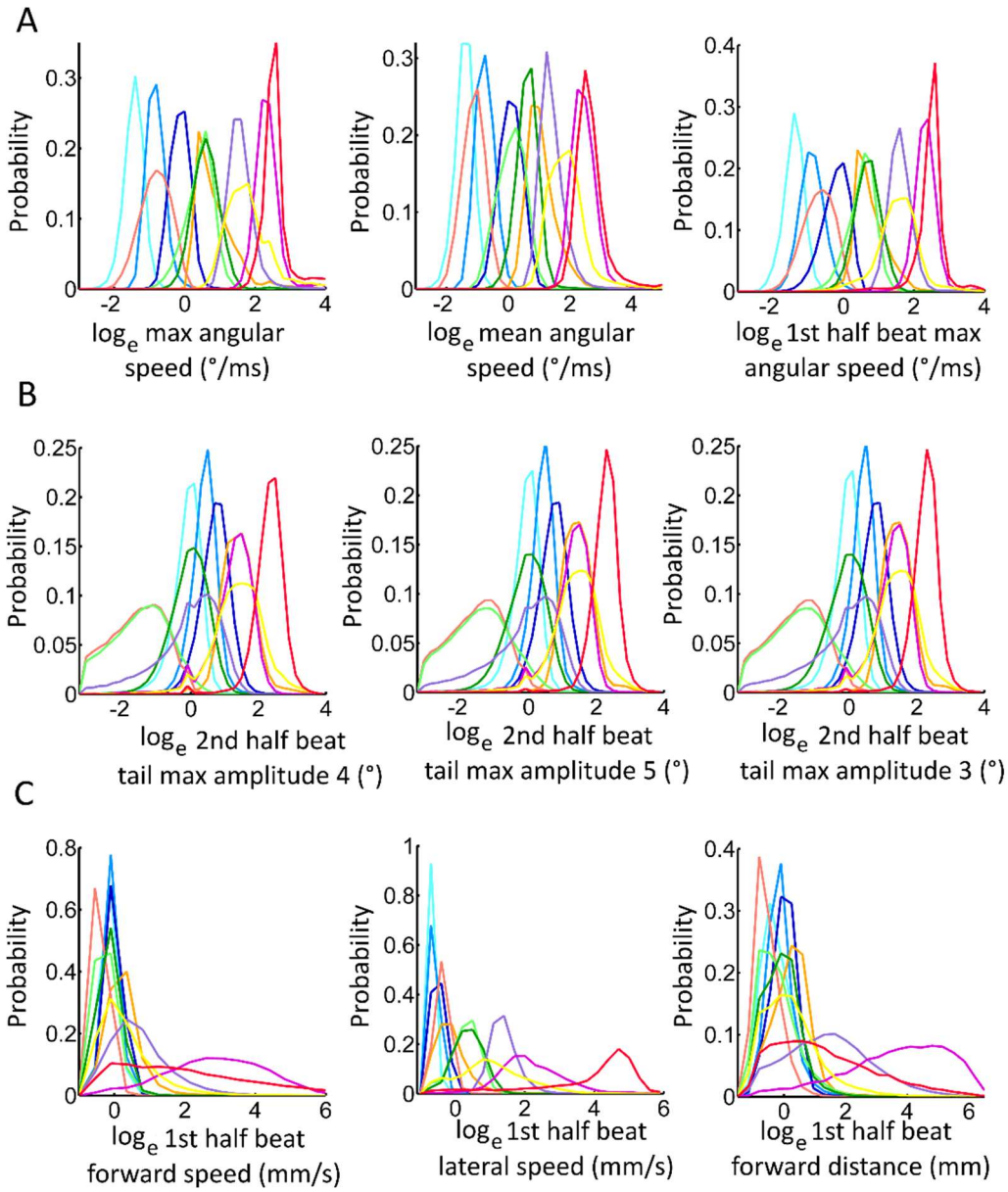

Figure 5.13 – Axes of movement space are formed by consistent kinematic parameters. Bout type probability distributions of the three major kinematic parameter contributors to the first PC (A), second PC (B), and third PC (C). AS (cyan), Slow 1 (blue), slow 2 (dark blue), BS (orange), J-turn (salmon), HAT (light green), RT (dark green), SAR (dark purple), O-bend (light purple), LLC (yellow), and C-start (red).

Bout Type Classification Works in Distinct Movement spaces

Although each of the movement space axes reflects a restricted subset of kinematic parameters this does not mean that those are the same kinematic parameters which would best be used to distinguish different bout types.

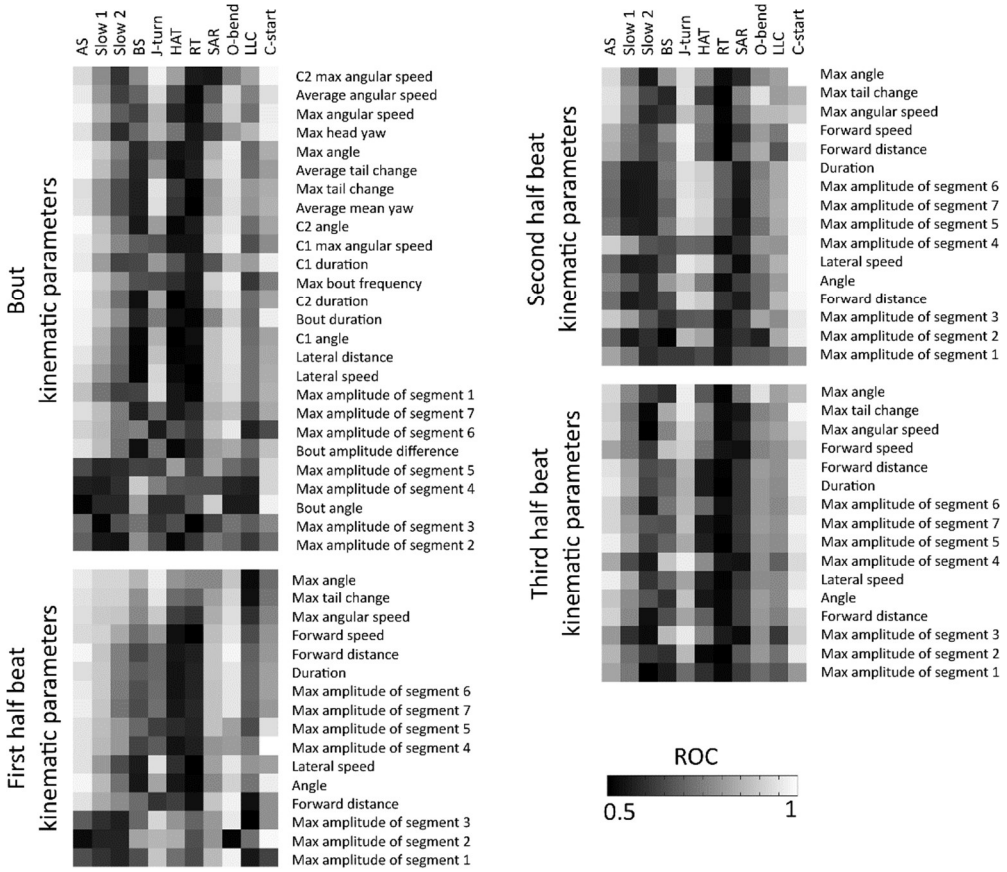

Figure 5.14 – Kinematic parameter ROC analysis. ROC was performed for the distribution of a particular bout type (column) against the distribution of all the other bout types together for a particular kinematic parameter (row).

Using the ROC analysis, we measured the similarity between the distributions of a particular bout type against the remaining ten for the 73 kinematic parameters (Tables 5.1-2). As can be seen in Figure 5.14, besides the kinematic parameters that constitute the movement space, there are many others that are able to separate well a single bout type from the others.

So, although the low-dimensional kinematic space in which we perform the clustering analysis is formed from a particular subset of kinematic parameters, it is still possible to use other kinematic variables to categorize the 11 bout types.

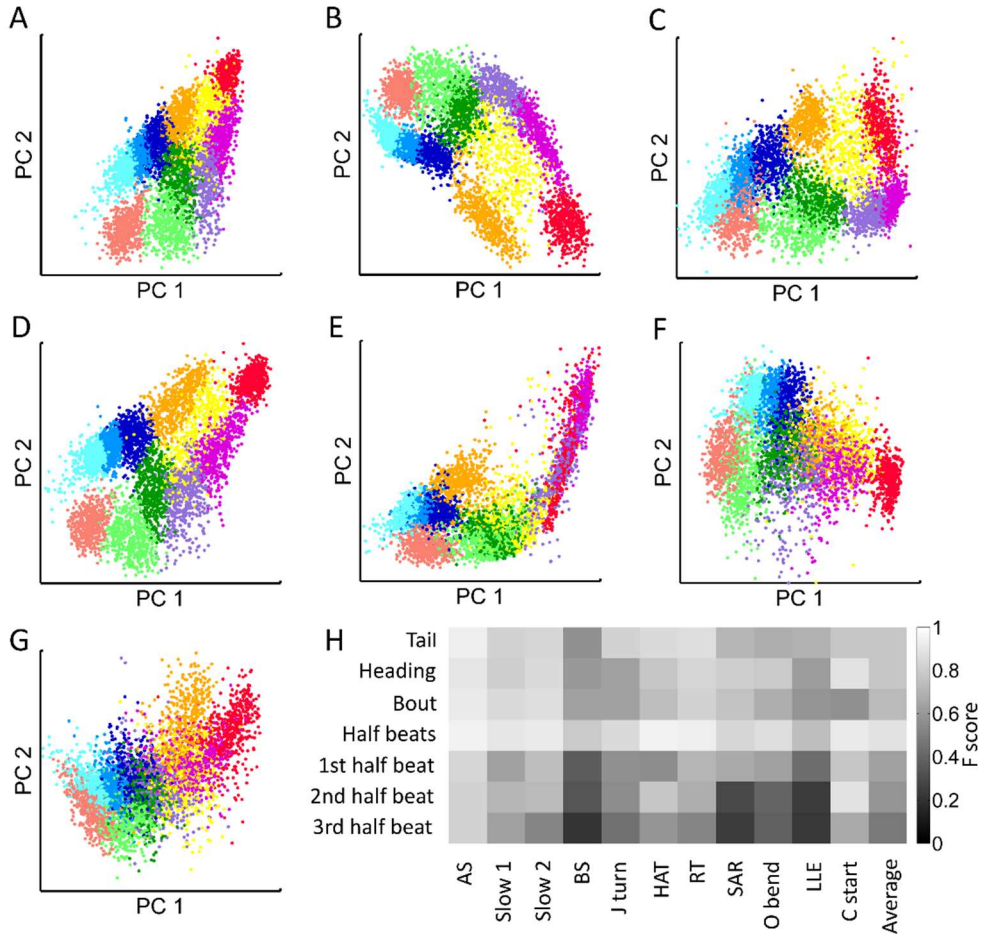

Figure 5.15 – Movement spaces created with subsets of kinematic parameters. Colours represent bout types categorized according to movement space obtained with the 73 kinematic parameters. AS (cyan), slow 1 (blue), slow 2 (dark blue), BS (orange), J-turn (salmon), HAT (light green), RT (dark green), SAR (dark purple), O-bend (light purple), LLC (yellow), and C-start (red). Movement spaces obtained with subsets of kinematic parameters: tail (A), heading (B), bout (C), half beats (D), 1st half beat (E), 2nd half beat (F), and 3rd half beat (G). (H) Classification F-scores of movement spaces in (A-G) compared with the original movement space.

The set of kinematic parameters we use to perform the initial PCA defines the movement space where the data exists and the analysis is performed. We choose to

form a movement space that was as unbiased as possible to any particular aspect of the zebrafish swimming. So we used a library of 73 kinematic parameters based on heading and tail movements of bouts, first, second and third half beats (Tables 5.1-2). However, many data sets only provide access to a subset of kinematic data. Many systems output position and heading data without detailed tail kinematics, while for head-restrained fish, the tail movement is known, but not the movement of the fish. To understand if it is possible to separate the eleven bout types using more limited parameter spaces, we placed our bout map into seven movement spaces composed of subsets of kinematic parameters and used the k nearest neighbour algorithm to reassign the identity of the full set of bouts (Figure 5.15A-G). After, we compared the new assignments with the original assignment obtained from the movement space constructed with the totality of the kinematic parameters. For the movement spaces constructed with: tail only, heading only, global bout parameters or the kinematic parameters of the individual half beats, the average performance is over 80 per cent (Figure 5.15H). When only information from a single half beat is used the F-scores for some of the bout types become much lower (F-score < 50). This may be relevant when trying to classify the bout type online as quickly as possible, based on limited information, for example. This analysis tells us that the eleven bout types still retain their identity, over a variety of movement parameter spaces, but that this is dependent on creating movement spaces that contain information of at least the first three half beats.

We interpret these results as evidence that the eleven bout types do not depend on a restricted set of kinematic parameters to separate them, but, on the contrary, may be distinguished using a variety of different variables.

DISCUSSION

Previously it was proposed for forward swimming (22) and prey capture (29) that zebrafish larvae movements do not form distinct groups, but may be better described

as a continuum that fish gradually control to perform distinct behaviours. However, in several cases there is good evidence for neural circuits dedicated to creating particular types of movements. It was shown that activity in an array of six segmentally arranged neurons including the Mauthner cells is correlated with the production of one particular type of escape, the C-start (16), and is necessary (31) for its production. Reticulospinal neuron activity during the optomotor response appears to indicate the existence of distinct groups of neurons mediating forward swimming and turns, which is also confirmed by ablation studies (24, 25). Also, at the level of the spinal cord it was described that there are pools of premotor interneurons, that are genetically, morphologically and functionally distinct, and are differentially recruited at different frequencies of swimming (22, 32). We found that for any of the 255 stimuli we presented in our experiment that swim bouts would form at least 2 kinematic clusters, indicating that the existence of distinct swim types is a general feature of larval behaviour.

However, it would be possible that these clusters would be formed due to gaps in distributions formed by "missing data" that would be present in other stimuli or behaviours. In order to test this hypothesis, we gathered data from nine innate behaviours of the zebrafish larvae and sampled the stimuli space for these behaviours systematically. Next, we embedded the clusters found for each stimuli in a common t-SNE space based on the clusters pairwise ROC values. We would expect that if the movement continuum hypothesis would be correct that the groups of swims would tile the t-SNE space in a homogenous way. What we observed instead was that these groups of swims form eleven groups, many of those completely separated from the others. The nine innate behaviours that constitute the bout collection we used for our analysis were chosen so that larvae would execute all the previously described bout types (chapter 4). Also, for every behaviour, we sampled the stimuli space abundantly, to try to gather the most diverse collection of movements possible. We cannot exclude the hypothesis that there are other untested stimuli or behaviours that can produce other bout types, but we find it unlikely that further experiments would

completely fill in the gaps between clusters that we observe in the t-SNE space, so as to produce a smooth continuum of movement types.

Our analysis is based in a common movement space that was created from the three principal components of 73 kinematic parameters. We choose to use many and diverse kinematic parameters (Tables 5.1-2) because we wanted our movement space to be as unbiased as possible. The PCA enabled us to decrease the dimensionality of the data, and also to make a movement space that is not distorted by the inclusion of many correlated kinematic parameters. We wondered if increasing the dimensions of the movement space would reveal more bout types. We started by looking at the unused principal components of the movement space and saw that the distributions of the bout types along these axes remain unimodal. Next, we repeated the same analysis with a movement space with 4 PCs and obtained the same results. In spite of that, we cannot rule out that further increasing the dimensionality of the movement space could allow the discovery other bout types.

It is also possible that using distinct combinations of kinematic parameters to produce other movement spaces would yield different solutions, however, since evidence of multimodal structure is common in all kinematic parameters we calculated we see no reason to exclude any subset of them. For many commonly used tracking systems, it may be impractical to calculate many of the kinematic parameters we used in our analysis. To determine if our categorization would be feasible to use in such cases we created movement spaces that would only contain subsets of kinematic parameters, and found that our categorization could applied in most cases.

Previously other unsupervised methods were used to find behavioural motifs for several model organisms (2–6). For the most part these methods found types of postures or movemes and need another analysis step to aggregate those into movements or actions. Thus, those methods are unable to capture behavioural motifs that exist in the movement time scale. Our approach is based on the discreteness of movements that zebrafish larvae perform (swim bouts) and the clustering step is

performed on complete movements themselves, making it a good method to study similarities between actions.

Of the eleven bout types we found seven that correspond clearly to ones that have already been described in previous ethological studies (7, 10, 14, 17, 30). This indicates that our categorization method is capturing a solution that is consistent with the human ability to discern movement patterns. We also found four novel bout types, namely: approach swims, two types of slow swims, two novel types of turns, HAT and SAR. It may be that these new bout types were found because of the large number of bouts collected and large stimulus space explored, that we used, but another possibility is that our movement classification approach detects swim types that humans may struggle to differentiate by simple observation. The movement categories should, ultimately, reflect the organization of neural activity in the larva's brain, so, an important test of the correctness of our classification will be whether, at some level, each bout type can be associated with engagement of a distinct underlying neural mechanism.

EXPERIMENTAL PROCEDURES

Behavioural Experiments and Tail Kinematics

Animal care, behavioural assays, automatic fish tracking, bout, half beat, and eye convergence detection are described in Chapter 4. A custom-made script written in Matlab 2010 (Mathworks) was used to compute 73 kinematic parameters for each detected bout (see summary in Tables 5.1-2).

Table 5.1. Bout kinematic parameters description.

Name	Description	Units
Max angular speed	Maximum value of the angular speed of the larva's body.	°/ms
Mean angular speed	Average value of the angular speed of the larva's body.	°/ms
Max head yaw	Maximum peak-to-peak amplitude of the angle of a line between the swim bladder and a point between the two eyes of the larva	°
Mean head yaw	Average peak-to-peak amplitude of the angle of a line between the swim bladder and a point between the two eyes of the larva	°
Max angle	Maximum change in the larva's body angle.	°
Bout end angle	The change in angle of the fish at the end of the bout.	°
Bout lateral distance	Absolute lateral distance	mm
Bout lateral speed	Absolute lateral speed	mm/ms
C1 angle	Body angle change at the end of the first half beat	°
C2 angle	Body angle change at the end of the second half beat	°
C1 max angular speed	Maximum angular velocity during the first half beat	°/ms
C2 max angular speed	Maximum angular velocity during the second half beat	°/ms
C1 duration	Duration of the first half beat	ms
C2 duration	Duration of the first half beat	ms
Tail max amplitude 1	Absolute maximum tail bend amplitude at 0.3 mm from the swim bladder	°
Tail max amplitude 2	Absolute maximum tail bend amplitude at 0.6 mm from the swim bladder	°
Tail max amplitude 3	Absolute maximum tail bend amplitude at 0.9 mm from the swim bladder	°
Tail max amplitude 4	Absolute maximum tail bend amplitude at 1.2 mm from the swim bladder	°
Tail max amplitude 5	Absolute maximum tail bend amplitude at 1.5 mm from the swim bladder	°
Tail max amplitude 6	Absolute maximum tail bend amplitude at 1.8 mm from the swim bladder	°
Tail max amplitude 7	Absolute maximum tail bend amplitude at 2.1 mm from the swim bladder	°
Max bout frequency	The maximum of the inverse of the time between successive extreme tail positions in the same direction	Hz
Max tail change	Maximum value of tail motion measure [a]	a.u.
Mean tail change	Average value of tail motion measure [a]	a.u.
Bout duration	Duration of bout	ms

[a] See Experimental Procedures Chapter 4 for details.

Table 5.2. First second and third half beat kinematic parameters description [b].

Name	Description	Units
End angle	The angle of the fish at the end of the half beat.	°
Max angle	Maximum value of the larva's angular displacement.	°
Max angular speed	Maximum value of the angular velocity of the larva.	°/ms
Forward distance	Forward distance that fish moves during half beat	mm
lateral distance	Absolute lateral distance that fish moves during half beat	mm
Forward speed	Forward speed that fish moves during half beat	mm/s
Lateral speed	Lateral speed that fish moves during half beat	mm/s
Tail max amplitude 1	Absolute maximum tail bend amplitude at 0.3 mm from the swim bladder	°
Tail max amplitude 2	Absolute maximum tail bend amplitude at 0.6 mm from the swim bladder	°
Tail max amplitude 3	Absolute maximum tail bend amplitude at 0.9 mm from the swim bladder	°
Tail max amplitude 4	Absolute maximum tail bend amplitude at 1.2 mm from the swim bladder	°
Tail max amplitude 5	Absolute maximum tail bend amplitude at 1.5 mm from the swim bladder	°
Tail max amplitude 6	Absolute maximum tail bend amplitude at 1.8 mm from the swim bladder	°
Tail max amplitude 7	Absolute maximum tail bend amplitude at 2.1 mm from the swim bladder	°
Max tail change	Maximum value of tail motion measure [b]	a.u.
Duration	Half beat duration	ms

[b] See Experimental Procedures Chapter 4 for details.

[a] For each kinematic parameter in this table an individual value is calculated for the first, second and third half beat.

Clustering of Stimuli Data Sets

PCA was performed on the 73 kinematic parameters (see Tables 5.1-2) of the 1794076 bouts that in total fish performed. To create bout movement spaces, we used between 3 and 4 principal components. After, the bouts were divided by the stimuli that produced them (see Tables 4.1-10 of chapter 4) with the exception of the bouts of the phototaxis and light dark transitions data sets, where bouts from some similar

stimuli were pooled together. Next, we calculated the local densities using adaptive gaussian density estimation (33) using all the bouts present in each stimulus data set. Next, for each data set we picked randomly 3000 bouts and applied the slow version of the density valley clustering (see chapter 3). For data sets with less than 3000 bouts the entirety of the data set was used. Confidence intervals were obtained by finding the probability of the existence of a second cluster in reference distributions obtained using the onion method (chapter 3) for a varying number of bouts (100,1000,2000,5000) randomly chosen from the overall data. The number of clusters for each data set were determined by finding the cluster centers that had higher SI value than the confidence interval value with the closest number of bouts to the data set.

Analysis to Determine the Number of Bout Types

For each pair of clusters, we projected their points onto the line between the two cluster centers and performed ROC analysis on the projected data. The ROC values for pairs of clusters that came from the same stimuli were inferred by averaging the 15^{th} nearest neighbours (34).

The ROC values were embedded in an ensemble of 200 three dimensional t-SNE spaces (perplexity of 30) (26), and density valley clustering was applied to find the number of groups of clusters present in each t-SNE space (chapter 3). To determine what is the most common solution for the ensemble of t-SNE spaces we used a voting system to combine the clustering results (27). We assume that t-SNE points that belong the most common solution will be co-located in the same cluster for different t-SNE solutions. So we perform single link clustering on the voting matrix of the ensemble of t-SNE solutions (28). The final number of bout types is obtained by the longest surviving partition in the resulting dendrogram.

REFERENCES

1. Tinbergen N (1951) The study of instinct. Available at: http://psycnet.apa.org/psycinfo/2004-16480-000.

2. Stephens GJ, Johnson-Kerner B, Bialek W, Ryu WS (2008) Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput Biol* 4:e1000028.

3. Berman GJ, Choi DM, Bialek W, Shaevitz JW (2014) Mapping the stereotyped behaviour of freely moving fruit flies. *J R Soc Interface* 11.

4. Vogelstein JT et al. (2014) Discovery of brainwide neural-behavioral maps via multiscale unsupervised structure learning. *Science* 344:386–92.

5. Girdhar K, Gruebele M, Chemla YR (2015) The Behavioral Space of Zebrafish Locomotion and Its Neural Network Analog. *PLoS ONE* 10:e0128668.

6. Wiltschko AB et al. (2015) Mapping Sub-Second Structure in Mouse Behavior. *Neuron* 88:1121–35.

7. Burgess HA, Granato M (2007) Modulation of locomotor activity in larval zebrafish during light adaptation. *J Exp Biol* 210:2526–39.

8. Mirat O, Sternberg JR, Severi KE, Wyart C (2013) ZebraZoom: an automated program for high-throughput behavioral analysis and categorization. *Front Neural Circuits* 7:107.

9. Severi KE et al. (2014) Neural control and modulation of swimming speed in the larval zebrafish. *Neuron* 83:692–707.

10. Budick SA, O'Malley DM (2000) Locomotor repertoire of the larval zebrafish: swimming, turning and prey capture. *J Exp Biol* 203:2565–79.

11. Burgess H, Schoch H, Granato M (2010) Distinct Retinal Pathways Drive Spatial Orientation Behaviors in Zebrafish Navigation. *Curr Biol* 20:381–386.

12. Fernandes A et al. (2012) Deep brain photoreceptors control light-seeking behavior in zebrafish larvae. *Curr Biology Cb* 22:2042–7.

13. Orger MB, Smear MC, Anstis SM, Baier H (2000) Perception of Fourier and non-Fourier motion by larval zebrafish. *Nat Neurosci* 3:1128–33.

14. Borla MA, Palecek B, Budick S, O'Malley DM (2002) Prey capture by larval zebrafish: evidence for fine axial motor control. *Brain Behav Evol* 60:207–29.

15. Bianco I, Kampff A, Engert F (2011) Prey capture behavior evoked by simple visual stimuli in larval zebrafish. *Front Syst Neurosci* 5:101.

16. O'Malley D, Kao Y-H, Fetcho J (1996) Imaging the Functional Organization of Zebrafish Hindbrain Segments during Escape Behaviors. *Neuron* 17:1145–1155.

17. Burgess HA, Granato M (2007) Sensorimotor gating in larval zebrafish. *J Neurosci* 27:4984–94.

18. Dunn TW et al. (2016) Neural Circuits Underlying Visually Evoked Escapes in Larval Zebrafish. *Neuron* 89:613–28.

19. Dreosti E, Lopes G, Kampff A, Wilson S (2015) Development of social behavior in young zebrafish. *Front Neural Circuits* 9.

20. Portugues R, Engert F (2009) The neural basis of visual behaviors in the larval zebrafish. *Curr Opin Neurobiol* 19:644–7.

21. Green MH, Ho RK, Hale ME (2011) Movement and function of the pectoral fins of the larval zebrafish (Danio rerio) during slow swimming. *J Exp Biol* 214:3111–23.

22. McLean DL, Masino MA, Koh IYY, Lindquist WB, Fetcho JR (2008) Continuous shifts in the active set of spinal interneurons during changes in locomotor speed. *Nat Neurosci* 11:1419–29.

23. Dunn TW et al. (2016) Brain-wide mapping of neural activity controlling zebrafish exploratory locomotion. *Elife* 5.

24. Orger MB, Kampff AR, Severi KE, Bollmann JH, Engert F (2008) Control of visually guided behavior by distinct populations of spinal projection neurons. *Nat Neurosci* 11:327–33.

25. Huang K-H, Ahrens M, Dunn T, Engert F (2013) Spinal projection neurons control turning behaviors in zebrafish. *Current biology* 23:1566–73.

26. Maaten L, Hinton G (2008) Visualizing data using t-SNE. *Journal of Machine Learning Research* 9:85.

27. Fred A, Jain A (2002) Data clustering using evidence accumulation. *Pattern Recognition* 4:276–280.

28. Sibson R (1973) SLINK: An optimally efficient algorithm for the single-link cluster method. *The Computer Journal* 16:30–34.

29. Patterson BW, Abraham AO, MacIver MA, McLean DL (2013) Visually guided gradation of prey capture movements in larval zebrafish. *J Exp Biol* 216:3071–83.

30. Liu K, Fetcho J (1999) Laser Ablations Reveal Functional Relationships of Segmental Hindbrain Neurons in Zebrafish. *Neuron* 23:325–335.

31. Lacoste A et al. (2015) A convergent and essential interneuron pathway for Mauthner-cell-mediated escapes. *Curr Biology Cb* 25:1526–34.

32. Ampatzis K, Song J, Ausborn J, Manira A (2014) Separate microcircuit modules of distinct v2a interneurons and motoneurons control the speed of locomotion. *Neuron* 83:934–43.

33. Alex Ihler and Mike Mandel (2003) Kernel Density Estimation Toolbox for MATLAB (R13) .

34. Chiu C-C, Chan S-Y, Wang C-C, Wu W-S (2013) Missing value imputation for microarray data: a comprehensive comparison study and a web tool. *Bmc Syst Biol* 7:S12.

CHAPTER 6 – Zebrafish Larvae Organize Bout Types into Complex Behavioural Sequences

ABSTRACT

Animal behaviour spans multiple levels of complexity. One framework for understanding this complexity proposes that the underlying processes of behaviour are organized in a hierarchy with multiple levels of integration. Zebrafish larvae also organize their behaviour over multiple levels. They produce short tail oscillations, whose sequences constitute swim bouts. In their turn, swim bouts can be organized into sequences. It was shown that sequences of bouts can be used to control speed and explore the environment.

Here we report that zebrafish use stereotypical sequences of bouts in many behaviours. We use features from these sequences to identify behavioural states that roughly correspond to fish's behaviours.

INTRODUCTION

Behaviour is a very diverse phenomenon that spans over variable degrees of complexity and time scales. In order to explain this complexity, Nikolaas Tinbergen proposed that the mechanisms underlying behaviour are arranged in a hierarchical system that is composed of various levels of integration (1). Furthermore, this author proposes that there are in the nervous system motor centres that integrate specific sensory and internal cues and channel the behavioural choices of the animal towards increasingly specific movements, culminating in the highly stereotypical "consummatory acts" (1). Recently, it was reported that a neural architecture compatible with the Tinbergen hierarchical model exists in the feeding and withdrawal neural networks of *Lymnaea stagnates* (2).

If animal behaviour is well described by a hierarchical architecture is still an open question. To shed light into this question it is important to identify repertoires of behavioural units at distinct levels for a wide range of behaviours and species. For the most part, such repertoires of behavioural units have been created by ethologists carefully observing animal behaviour and establishing subjective criteria that define such categories (for example see (3–5)) (for an exhaustive review see (1)). Recently, computational methods have been created that identify behavioural units automatically and without human supervision ((6–10) and chapter 5). However, in order to understand if behaviour is organized in a hierarchy, and to describe such a system, it is essential to create theoretical frameworks that are able to identify and bridge distinct behavioural levels.

One approach consists in creating mathematical models based on simple mathematical principles, whose parameters are inferred from real data, and that reproduce the main features of animal behaviour. It is possible that these models create realistic outcomes and could be used to study *in silico* if behaviour is organized in distinct levels (for an example with fish (11, 12)).

Another approach consists in using computational models with certain assumptions about the behavioural data that try to explicitly explain how simpler behavioural units

are assembled into more complex patterns of behaviour (7, 13). To avoid imposing a unique set of assumptions on the data Wiltschko *et al* have used a library of computational models, each one with a specific hierarchical structure, and found the one that best explains behavioural data. This approach was used to find behavioural motifs from postures in the mouse (10). However, it is still limited by the fact that its outcome is dependent on the list of models that compose the library.

A distinct method, based in using a clustering algorithm that produces hierarchical solutions, was used to find "behaviotypes" of Drosophila larvae (9). This method tries to find the best hierarchical structure that explains the data, being able to produce hierarchies of behaviours that were not a priori determined. However, the distinct levels it produces do not represent behavioural patterns at distinct time scales.

Finally, one may not impose any model on the data and use machine learning techniques to find commonly repeated sequences of behavioural units. Such an approach was used to create a library of behavioural motifs from postures in *C. elegans* (14, 15) and it implicitly assumes that higher levels of complexity are composed of sequences of behavioural units of the lower levels.

Zebrafish larvae organize their behaviour in three very distinct levels (see chapter 1); they combine tail oscillations (half beats) into discrete movements, called swim bouts. These discrete units of behaviour can happen in isolation (for example as a startle response to a sound), but may also exist in sequences of bouts that last through the duration of the sensory stimulus.

We have previously described that during locomotion larvae use sequences of bouts that are composed of distinct gaits to control speed (chapter 2 and (16)). Also, Dunn and colleagues have reported that during swimming in a featureless environment fish organize their bouts in sequences of left and right-directed swims, enabling a better exploration of the environment (13).

Here, we show that the organization of swim bouts into stereotypical sequences is widespread through many of the fish' behaviours. We use a simple computational method, based on clustering of probabilities of swim bouts and their transition

probabilities to create a "sequence space" that bridges the gap between two levels of behaviour; the bout and bout-sequence levels. By using this space, we observe that distinct stimulus parameters modulate sequences of bouts differently. We also find that bout sequences form six groups that to a large extent overlap with the categories of behaviour that the fish were performing.

RESULTS

Zebrafish Larvae Organize Swim Bouts into Sequences for Many Behaviours

Of the nine behaviours we tested, two are composed of single movement events and not sequences of movements, that is, the O-bend response to dimming stimuli and startle responses to acoustic stimuli. For the other seven behaviours the zebrafish larvae produce sequences of bouts during the duration of the stimuli. However, do larvae place bouts randomly within these sequences, or is there an order that is specific for each behaviour? Are the transitions between bouts modulated by the stimulus?

To answer these questions, we computed the bout transition probabilities for each of the behaviours during which larva respond with multiple bouts i.e., forward OMR, directional OMR, phototaxis, spontaneous swimming in the dark, spontaneous swimming in the light, prey capture, and avoidance of expanding spots.

Overall, it can be seen in Figure 6.1 that for all the stimuli we included in this analysis the transition probabilities between bout types are not random, implying that the bout sequences are ordered for all the behaviours we analysed. Another striking feature that can be observed in the ethograms is that they look similar for stimuli from the same behaviour. In spite of that, the ethograms are not identical for the stimuli that compose the same behaviours but seem to change in an orderly manner with the stimulus parameters (Figure 6.1).

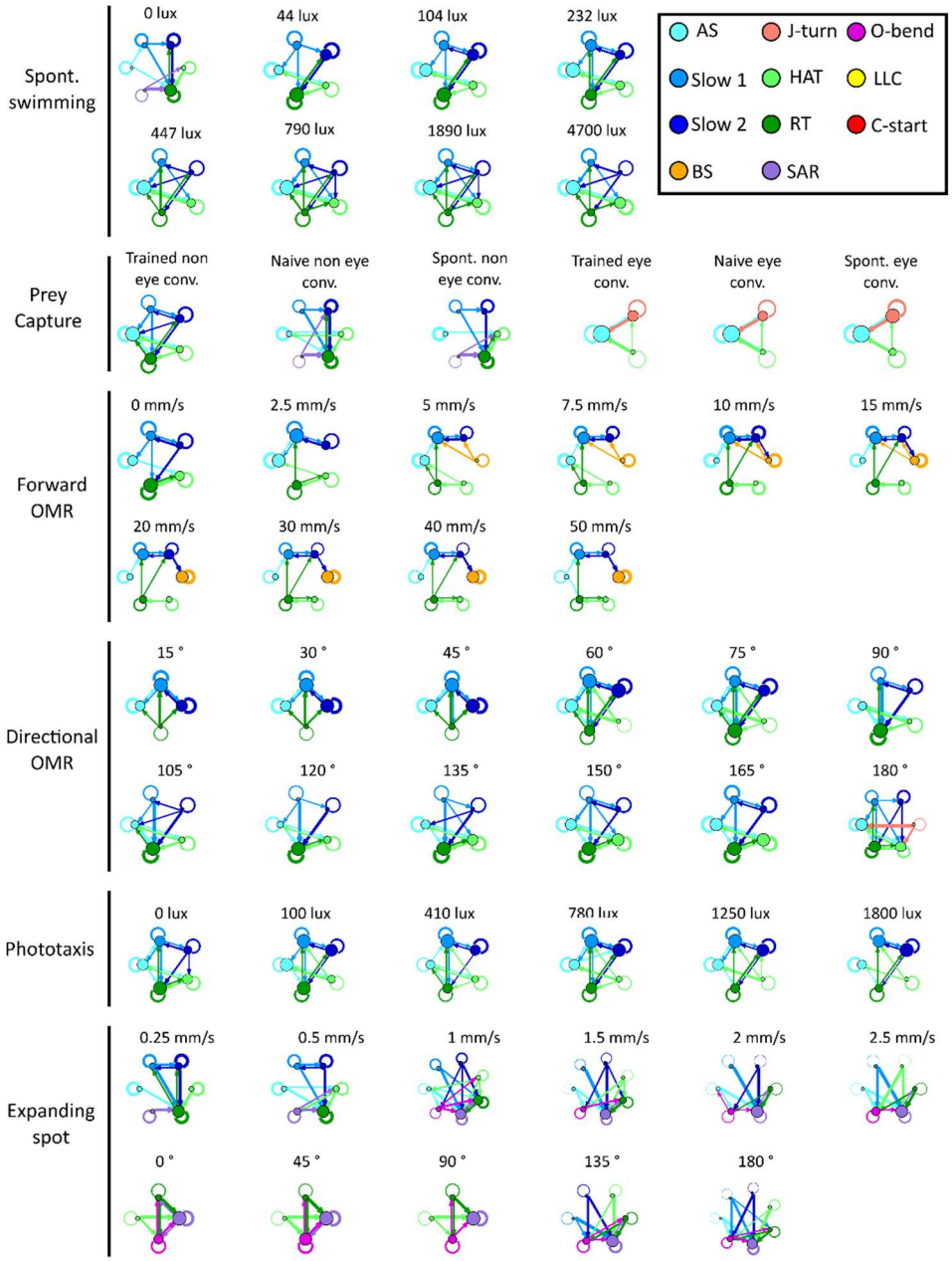

Figure 6.1 – Larvae execute sequences of bouts for several behaviours. Bout transitions for several stimuli. Size of the circles represent the probability of bout type and arrow widths represent the transition probabilities between movements. The size of outer circles represents repetitions of same bout type. Colour of circles represent bout types and arrows are coloured according to the initial bout type in the transition. Displaying bout types with probability > 0.05 and transitions probabilities > 0.15.

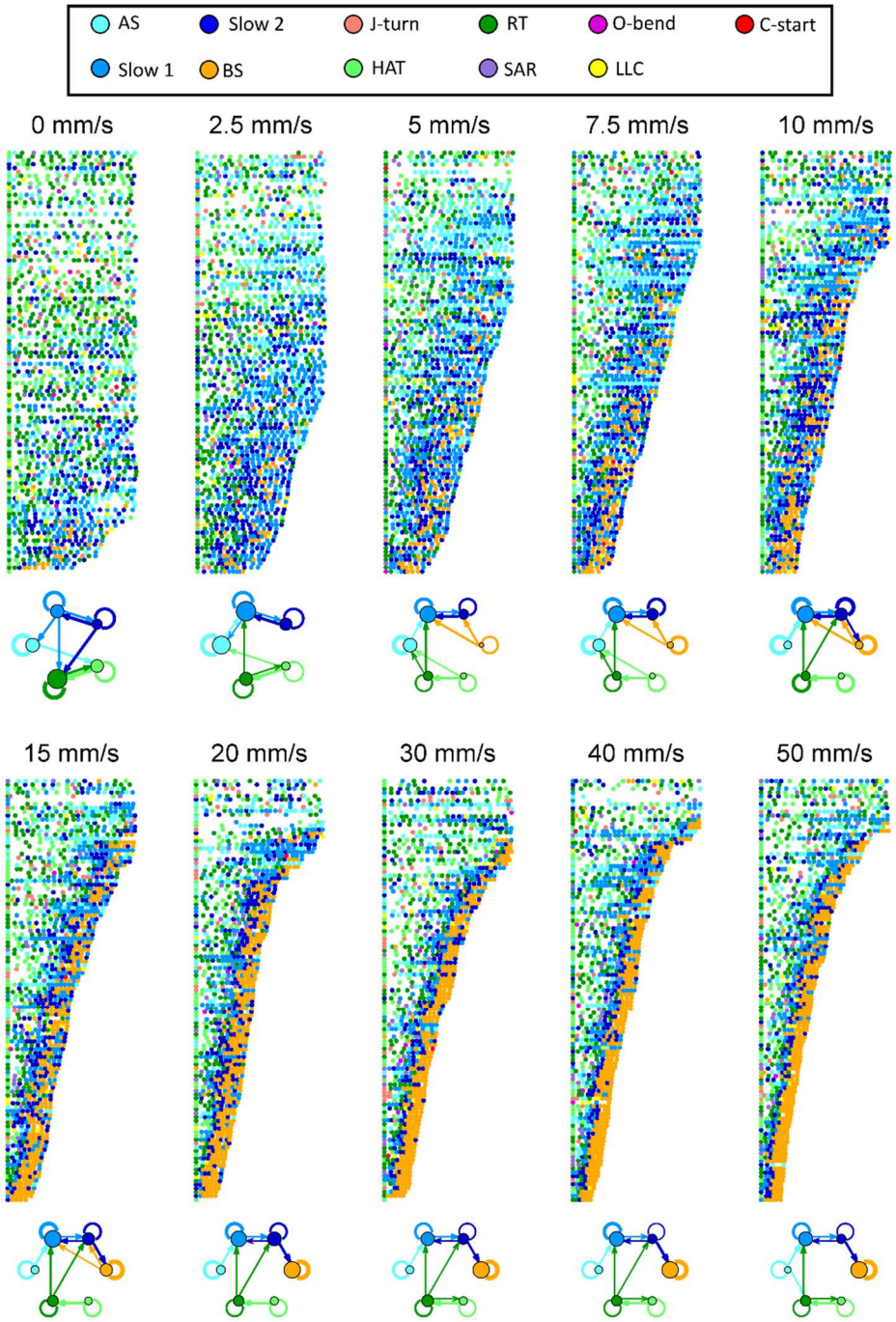

Figure 6.2 – Sequences of bouts for forward OMR. Behavioural raster plots for trials of the same stimulus. Trials are ordered by decreasing duration. Beneath each raster plot there is the corresponding ethogram. On top of each raster is the grating speed. Bout types legend is in the upper box.

Stimulus Modulates Probability and Transitions Between Bout Types

The bout sequences for these behaviours are not random and they seem to be modulated by the stimulus parameters. But exactly what about the sequences is being modulated by the stimulus?

For some behaviours, new bout types appear and increase in frequency as the relevant stimulus parameter is varied. This is particularly striking for the forward OMR behaviour, where burst swims start to appear for gratings moving at 5mm/s and steadily increase to one quarter of the bouts for higher speed gratings (Figure 6.2). This bout type modulation can be clearly visible if one makes behavioural raster plots where each bout type is indicated by a point with a distinct colour (see orange stripes in high speed gratings of Figure 6.2).

Another feature that is commonly being modulated by stimuli is the probability of transitions between states. For example, if one looks more closely at the ethograms of the directional OMR (Figure 6.3), it becomes apparent that the grating orientation not only modulates the bout type probabilities (routine turns and HATs increase while forward swims decrease with the gratings angle), but also there are consistent changes in the transitions between states. For the gratings with low angles, routine turns mostly transition towards forward bout types (AS and slow swims 1 and slow swims 2) that in turn transition between themselves (Figure 6.3A). For gratings with angles between 60 and 90 degrees there are bidirectional transitions between the forward bouts and the routine turns (Figure 6.3B). For the gratings with higher angles these transitions become inverted, being oriented towards the routine turns and HATs (Figure 6.3C-D).

The bout probability and the bout transitions not only change with stimuli for the OMR behaviours but also for the other behaviours for which we varied the stimulus (Figure 6.1).

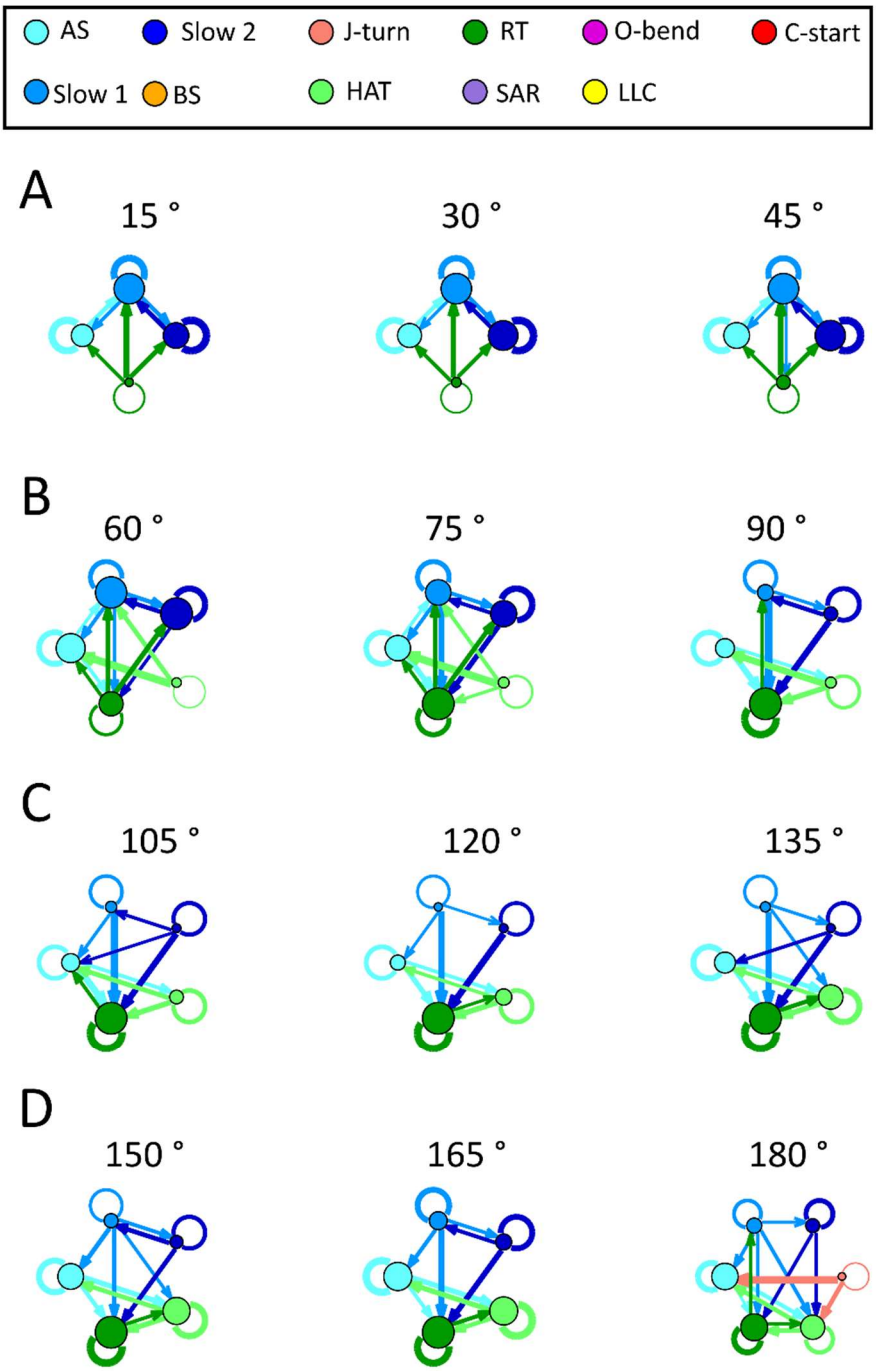

Figure 6.3 – Ethograms for directional OMR. Each ethogram represents the bouts probabilities and transition probabilities for the trials with a particular moving grating orientation. (A) Moving gratings of low angles. (C-D) Moving gratings of intermediate angles. (E) Moving gratings of high angles.

Information Analysis of Sequences

The frequency of bout types and the transitions between states are modulated for most of the behaviours we analysed. So it is likely that these features are useful to predict what is the next action that is going to be performed in a bout sequence. Calculating the entropy at different levels of knowledge of a sequence provides a measure of the predictability of a behavioural sequence and the level on which dependencies exist in the sequence (4).

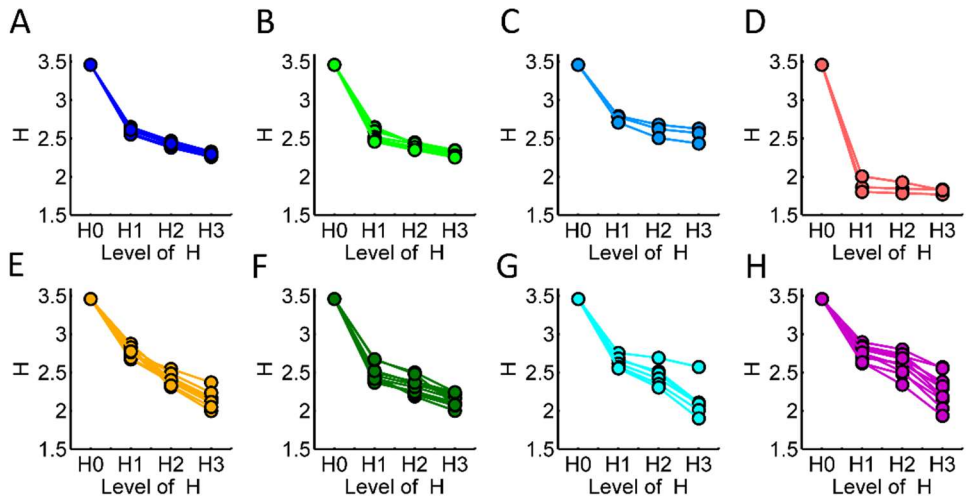

Figure 6.4 – Information analysis for sequences of different behaviours. Each line corresponds to the entropy at different gaps in the sequence of a stimulus. (A) Spontaneous swimming in the dark. (B) Spontaneous swimming in the light. (C) Prey capture when the eyes are not converged. (D) Prey capture when the eyes are converged. (E) Forward OMR. (F) Directional OMR. (G) Phototaxis. (H) Expanding spot avoidance.

For every stimulus we calculated the entropy for levels 0, 1, 2 and 3 (see Experimental Procedures). For all behaviours the probability of bout types is a major factor in reducing uncertainty (H1) (Figure 6.4). Also, for all behaviours (with the exception of prey capture) the knowledge of the immediate preceding action (H2) provides a meaningful reduction of the uncertainty (Figure 6.4). The third order dependencies seem to be mostly important for OMR, phototaxis and spot avoidance behaviours (Figure 6.4E-H).

Stimulus Produce Independent Trajectories in Sequence Space

Having established that larvae, for many of the behaviours, organize their bout types into sequences that show dependencies until at least the third level we wondered if we could use the transitions between states to construct a "sequence space" where stimuli with similar bout sequences occupy nearby positions.

We constructed such space by using the bout type probabilities, and the second and third order transitions (Figure 6.5) and decreased the dimensionality of the data to three dimensions by using the t-SNE method (17).

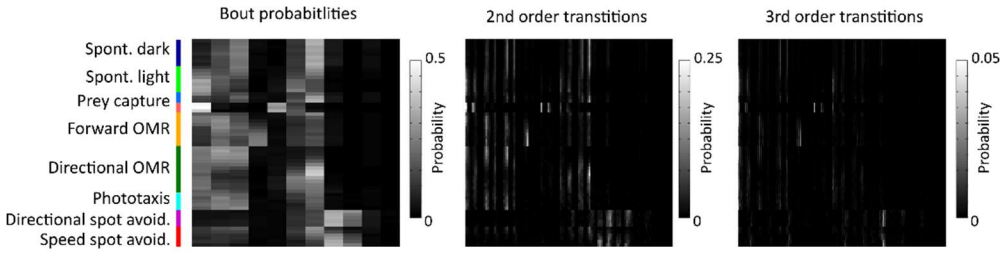

Figure 6.5 – Information used to construct sequence space. Rows represent different stimuli grouped by behaviours according to key and ordered by increasing strength. Columns are "features" that were used to construct sequence space.

Each stimulus exists in this sequence space and the location it occupies within it depends on the particular bout sequence that fish did in response to that stimulus. For six of our behaviours (i.e. spontaneous swimming in the dark, spontaneous swimming in the light, forward OMR, directional OMR, phototaxis, and spot avoidance) we controlled the stimulus and varied its parameters. So we wondered if distinct stimuli modulate bout sequences in specific ways. To test this, we labeled for each behaviour its data points according to the value of the stimulus parameter that was varied (Figure 6.6). The only behaviour that did not display an orderly trajectory was the spontaneous swimming in the dark, for which we had previously noticed that the fish's locomotion would not vary with the level of the preceding light stimulus, which was what we manipulated (Figure 6.6H) (see chapter 4). The other behaviours show orderly trajectories in the sequence space (Figure 6.6A-G). If we overly the distinct

stimulus trajectories we can see that they, for the most part, take a distinct path through the space, implying that each stimulus modulates the bout sequences in a specific way (Figure 6.6H).

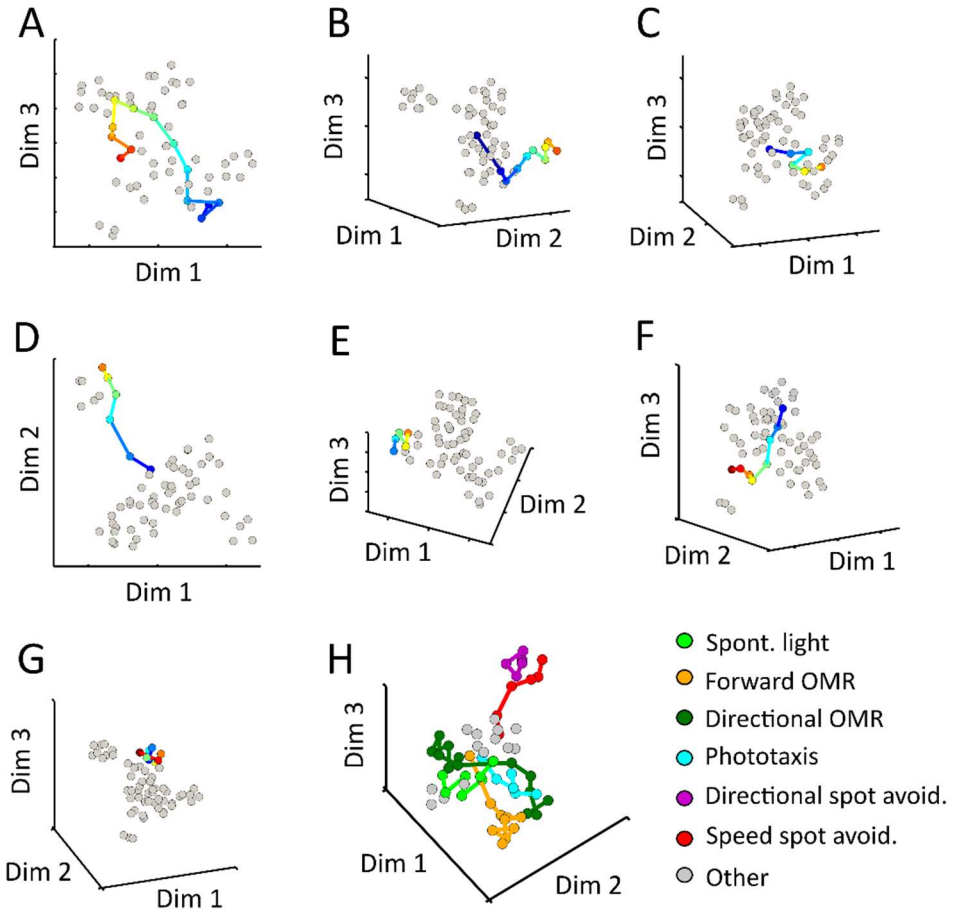

Figure 6.6 – Stimulus form trajectories through sequence space. Sequence space created by using the t-SNE method on bout transition probabilities. (A) Directional OMR. (B) Forward OMR. (C) Phototaxis. (D) Expanding spots at various speeds. (E) Expanding spots at various directions. (F) Spontaneous swimming at different light intensities. (G) Spontaneous swimming in the dark. Grey points correspond to other behaviours and colours change from blue to red with increasing stimulus strength. (H) Overlay of stimulus trajectories for each behaviour. Left panel has the legend of behaviours.

Stimulus Sequences Cluster into Behavioural States

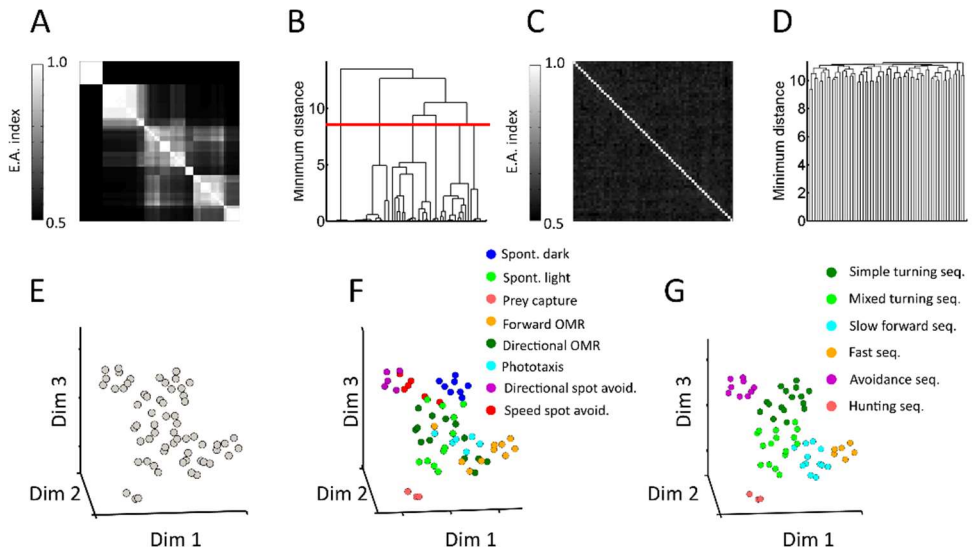

Figure 6.7 – Stimulus sequences form clusters in sequence space. (A) Evidence accumulation matrix of 200 t-SNE solutions solved by density valley clustering. The matrix is ordered by single link clustering. (B) Dendrogram of A. (C) Voting matrix of shuffled data. (D) Dendrogram of C. (E) Example of t-SNE solution. (F) Same t-SNE space as in E with data points coloured by behaviour. (G) Same t-SNE space as in E with data points coloured according to clustering solution obtained in B.

We also noticed that data points formed clusters in the sequence space (Figure 6.6G). Since similar sequences will occupy similar locations in this space we wondered if the clusters that we observed correspond to "behavioural states" that engaged when fish are responding to particular stimuli. To answer this question, we created an ensemble of 200 behavioural t-SNE spaces and determined the number of clusters on the t-SNE ensemble (see Experimental Procedures) (Figure 6.7). We obtained a dendrogram with a cut-off, based on the max jump criterion, at the level of 6 clusters (Figure 6.7A -B). Also, if we randomize the cluster identity we recover no discernible structure (Figure 6.7 C-D). The fact that the sequence space has structure means that the fishes' behavioural sequences, in response to different stimuli, are composed of a set of distinct groups that are more similar to sequences within their group, than to others. Do these groups of sequences correspond to "behavioural states" that fish manifested in response to distinct types of stimuli? Do these clusters of sequences have any correspondence to the seven behaviours that went into this analysis?

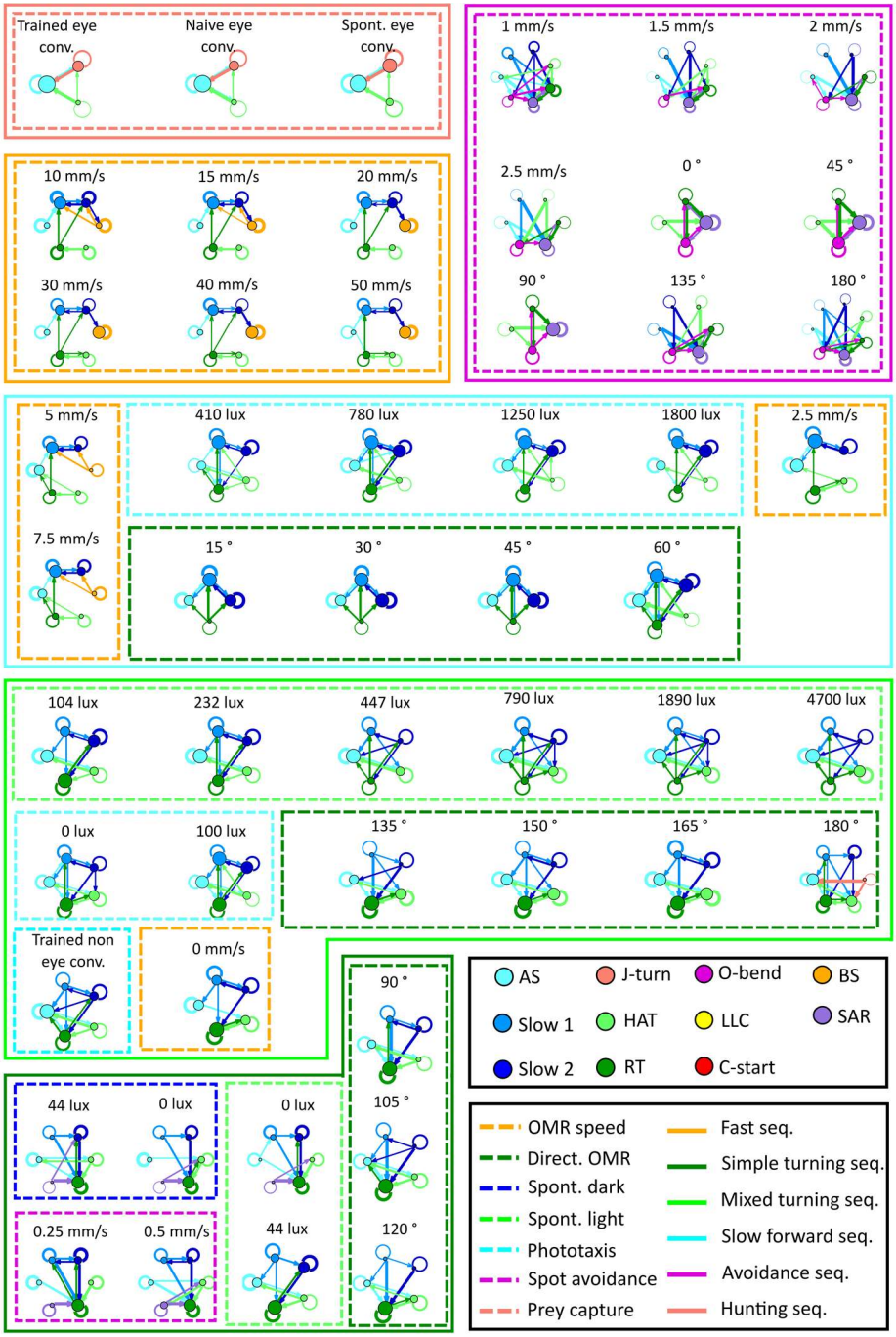

Figure 6.8 – Behavioural states correspond to similar ethograms. Ethograms are organized according to the six groups that were obtained from the clustering analysis (solid boxes). Inside each solid box the ethograms are divided according to behaviours (dashed boxes).

To clarify this question, we labelled every stimulus on the t-SNE space with the behaviour that the stimulus belongs to (Figure 6.7F) and compared this plot to the same data points colour coded by the obtained clustering solution (Figure 6.7G). It can be observed that there is a very high correspondence for the prey capture, spot avoidance and forward OMR behaviours (compare Figure 6.7F-G). For the spontaneous swimming, phototaxis and directional OMR behaviours the evoked sequences are spread across multiple categories.

DISCUSSION

Dunn *et al* have observed that fish in a homogenous environment swim by executing alternating sequences of repeated turns to the left and to the right (13). Also, these authors have modelled this behaviour using a hidden Markov chain analysis and have shown that such pattern of sequences sample better the environment than random walking (13). We have reported that to move at high speeds larva form highly structured bout sequences where slow swims are followed by burst swims ((16) and chapter 2).

Here, we show that bouts are organized in sequences for other behaviours: directional OMR, avoidance to expanding spots, spontaneous swimming in the dark, phototaxis and prey capture. If for these behaviours these sequences have an adaptive value is an open question, which could be addressed by computational modelling of these behaviours.

We took a computational approach to identify a space where variations in behaviour at the bout-sequence level can be visualized. We tried, in our approach, to keep the analysis simple, and not base it on any particular model of the fish's behaviour. It relies on behavioural categories that were obtained by an unsupervised data driven method (see chapter 5) and on the observation that those units of behaviour form sequences with certain levels of dependencies. This has the advantage that it does not require any parameter estimation or more specific assumptions about how larval

behaviour is generated. In this sense, it is a data driven method much like (9, 14). However, it also has its limitations, it relies on stimuli to define the epochs that are used to calculate the transitions between bouts. So the "features" used to create the sequence space are "average" features of the sequences that fish performed during those stimuli, and it is not able to separate distinct types of sequences that happen during the same stimulus. A good example of a behaviour where this happens is the forward OMR behaviour, where it is likely that fish transition between an initial turning sequence state to a fast sequence state during every trial (Figure 6.2).

We observed that stimuli form clusters in the sequence space. These groups may be seen as average behavioural sequence states that fish had during the presentation of the stimulus. Three of the clusters we obtained directly correspond to individual behavioural paradigms, so we have named them according to those behaviours: fast, avoidance and hunting sequences (see salmon, orange, and purple solid boxes in Figure 6.8). The other three clusters our analysis detected are composed of mixtures of behaviours (see dashed boxes on Figure 6.8). However, if one looks at the individual ethograms that compose these mixed behavioural states, one can observe that they share similar features within group. The first mixed cluster is composed of ethograms that for the most part are composed of slow forward bouts and transitions that show a progression from turns to forward swims (cyan solid box on Figure 6.8), so we named this group of sequences "slow forward sequences". The sequences on the second mixed cluster are composed of a large proportion of turns of two types (HATs and routine turns). Also the transitions are, for the most part, directed toward the turn states. We have called these sequences "mixed turning sequences". Finally, the sequences of the last mixed cluster are for the most part composed of routine turns, so we have named these "simple turning sequences".

If these categories of sequences have underlying corresponding neural mechanisms is an open question that should be investigated.

EXPERIMENTAL PROCEDURES

Information Analysis of Sequences

Information analysis measures the degree to which a sequence is predictable in regard to a particular level of knowledge. The lowest level of knowledge of a behavioural sequence is the number of actions an animal is able to perform and its entropy is calculated by:

$$H_0 = log_2(Number\ states)$$

The next level of knowledge of a sequence is defined as knowing the probabilities of the behavioural states:

$$H_1 = \sum P_i log_2(1/p_i)$$

where P_i is the observed probability of each state. This level is pictured in the ethogram plots by the size of the circles. The following level is the transition between individual states (the arrows in the ethogram plot) and is calculated by:

$$H_2 = \left(\sum P_{pair}\left(log_2(1/P_{pair})\right) - H_1\right)$$

where P_{pair} is the observed probability of each sequential pair of actions. Similarly, H_3 entropy is calculated by;

$$H_3 = \left(\sum P_{triplet}\left(log_2(1/P_{triplet})\right) - H_2\right)$$

Where $P_{triplet}$ is the observed probability of each possible sequence of three actions (4).

Analysis to Determine the Number of Behavioural States

The bout probability, second, and third order bout transitions values were embedded in an ensemble of 200 three dimensional t-SNE spaces (perplexity of 30). One of this

spaces was used to visualize the stimulus trajectories. The number of clusters that were present in the t-SNE ensemble was determined using the same method we previously applied to find the number of bouts types that fish performed (see chapter 4).

REFERENCES

1. Tinbergen N (1951) The study of instinct. Available at: http://psycnet.apa.org/psycinfo/2004-16480-000.

2. Pirger Z et al. (2014) Interneuronal Mechanism for Tinbergen's Hierarchical Model of Behavioral Choice. *Curr Biol* 24:2018–2024.

3. Dawkins M (1976) Hierachical organization and postural facilitation: Rules for grooming in flies. *Animal Behaviour*. Available at: http://www.sciencedirect.com/science/article/pii/S0003347276800036.

4. Berridge K, Fentress J, Parr H (1987) Natural syntax rules control action sequence of rats. *Behav Brain Res* 23:59–68.

5. Seeds AM, Ravbar P, Chung P, Hampel S, Midgley FM (2014) A suppression hierarchy among competing motor programs drives sequential grooming in Drosophila. *Elife*. Available at: http://elifesciences.org/content/3/e02951.abstract.

6. Stephens GJ, Johnson-Kerner B, Bialek W, Ryu WS (2008) Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput Biol* 4:e1000028.

7. Girdhar K, Gruebele M, Chemla YR (2015) The Behavioral Space of Zebrafish Locomotion and Its Neural Network Analog. *PLoS ONE* 10:e0128668.

8. Berman GJ, Choi DM, Bialek W, Shaevitz JW (2014) Mapping the stereotyped behaviour of freely moving fruit flies. *J R Soc Interface* 11.

9. Vogelstein JT et al. (2014) Discovery of brainwide neural-behavioral maps via multiscale unsupervised structure learning. *Science* 344:386–92.

10. Wiltschko AB et al. (2015) Mapping Sub-Second Structure in Mouse Behavior. *Neuron* 88:1121–35.

11. Mwaffo V, Anderson R, Butail S, Porfiri M (2014) A jump persistent turning walker to model zebrafish locomotion. *J Roy Soc Interface* 12:20140884–20140884.

12. Zienkiewicz A, Barton D, Porfiri M, Bernardo M (2015) Data-driven stochastic modelling of zebrafish locomotion. *J Math Biol* 71:1081–1105.

13. Dunn TW et al. (2016) Brain-wide mapping of neural activity controlling zebrafish exploratory locomotion. *Elife* 5.

14. Brown AEE, Yemini EI, Grundy LJ, Jucikas T, Schafer WR (2013) A dictionary of behavioral motifs reveals clusters of genes affecting Caenorhabditis elegans locomotion. *Proc Natl Acad Sci USA* 110:791–6.

15. Schwarz R, Branicky R, Grundy L, Schafer W, Brown A (2015) Changes in Postural Syntax Characterize Sensory Modulation and Natural Variation of C. elegans Locomotion. *Plos Comput Biol* 11:e1004322.

16. Severi KE et al. (2014) Neural control and modulation of swimming speed in the larval zebrafish. *Neuron* 83:692–707.

17. Maaten L, Hinton G (2008) Visualizing data using t-SNE. *Journal of Machine Learning Research* 9:85.