博士論文

論文題目　**Bioinformatics for Understanding Animal Behavior**

(動物行動を理解するためのバイオインフォマティクス技術の開発)

氏名　福永　津嵩

Acknowledgements

This dissertation could not have been possible without support of many people. I would like to take this opportunity to express my gratitude to all of the people who helped me during my Ph.D. study.

First, I am deeply grateful to my supervisor, Associate Professor Wataru Iwasaki for providing me with an opportunity to try this new and exciting research area. I was often helped by his insightful advice based on his broader perspective. Next, I want to thank Associate Professor Hisanori Kiryu as my supervisor in master's degree course. I learned many basic research skills from him. My scientific senses of values are very affected by his strict attitude towards science. I would also like to thank Professor Akihiro Nakaya at Osaka University as my supervisor in bachelor's degree course. He gave me constructive comments on my research and beneficial advice on career design.

I would like to show my appreciation to my Ph.D. thesis committee members, who are Professor Koji Tsuda, Professor Shinichi Morishita, Associate Professor Kei Ito, Associate Professor Wataru Iwasaki, and Associate Professor Hideaki Takeuchi at Okayama University for valuable comments to this thesis.

I would like to show my gratitude to Ms. Shoko Kubota and Associate Professor Shoji Oda, who are co-authors of our paper published in *Computational Biology and Chemistry*. They kindly provided the video data and valuable comments on this research. In particular, Dr. Oda's expert knowledge on medaka biology and unique viewpoints gave me insights into integration of ethology and bioinformatics. I also express my respect to Mr. Rito Takeuchi and Mr. Osamu Yamanaka for developing UMATracker system, which is a GUI system of GroupTracker algorithm. I would also like to thank Dr. Haruka Ozaki, Dr. Goro Terai, and Professor Kiyoshi Asai, who are co-authors of our paper published in *Genome Biology*. Thanks to their comments and suggestions, I could deepen this research. I also express my appreciation to all co-authors of our papers published in *PLOS ONE*, *Molecular Biology and Evolution*, and *Royal Society Open Science*. Specifically, I thank Dr. Takashi P. Sato, Dr. Masaki Miya and Dr. Mutsumi Nishida for considerable suggestions to software development from viewpoint of icthyology.

I thank the former and current members in Iwasaki Laboratory, especially Dr. Shotaro Hirase, Dr. Ching-Chia Yang, Dr. Akira Iguchi, Dr. Sira Sriswasdi, Dr. Motomu Matsui, Dr. Seishiro Aoki, Mr. Satoshi Hiraoka, Mr. Yohei Kumagai, and Mr. Yoshinori Nii for helpful discussions and continuous encouragement. I also express my respect and gratitude to laboratory secretaries and department staff, especially Ms. Yoko Nomura, Ms. Miyuki Nakasendo, Ms. Isako Oda, and Ms. Naoko Tomioka for their appreciated

support of my laboratory life.

I would like to offer my special thanks to graduates and students of the Department of Computational Biology and Medical Sciences. In particular, I thank Dr. Haruka Ozaki, Mr. Keisuke Tada, Mr. Hirotaka Matsumoto, Mr. Ryota Mori, Mr. Takamasa Imai, Mr. Suguru Nishijima, Mr. Yuki Kashihara, and Ms. Risa Kawaguchi for the wonderful time spent together. I also wish to express my gratitude to Professor Toshihisa Takagi, Professor Masanori Arita at National Institute of Genetics, Associate Professor Michiaki Hamada at Waseda University, Senior Assistant Professor Susumu Yoshizawa, Senior Assistant Professor Masahiro Kasahara, and Senior Assistant Professor Kengo Sato at Keio University for valuable suggestion and beneficial advice in my research life.

I would like to show my appreciation to Japanese Society for the Promotion of Science (JSPS) Research Fellow (DC2), and JSPS Global COE program "Genome BigBang" led by Professor Shinichi Morishita, for financial support throughout my Ph.D. course. The computations in this research were performed using the supercomputing facilities at the Human Genome Center in The University of Tokyo and National Institute of Genetics in Research Organization of Information and Systems.

The trigger of me becoming interested in integration of applied mathematics and ethology was my high school's biology class about behavioral ecology. Without this class presented by Mr. Shigeki Okuyama, who was my high school biology teacher, this dissertation would not have been possible. I owe my deepest gratitude to him.

Last but not least, I thank my girlfriend, Yumiko Tatsuta, and my parents, Tsuneharu Fukunaga and Tsuyuko Fukunaga for their mental support. I dedicate this dissertation to them.

February 19, 2016
Tsukasa Fukunaga

Chapter 1

General Introduction

1.1 Computational Ethology: Integration of Bioinformatics and Ethology

From ancient times, mankind has been interested in the unique behavior of diverse animal species. In the 4th century BC, Aristotle presented a broad overview of ethological knowledge, derived from careful observation, in his book "History of Animals"[1]. Since then, countless ethological studies based on natural historical approaches have been performed, which have provided fascinating insights into animal behavior. As in other fields of biology, experimental methods were introduced to the discipline of ethology at the beginning of the 20th century AD. Karl von Frisch, Nikolaas Tinbergen, and Konrad Lorenz pioneered the field of "experimental ethology", and were awarded the Nobel Prize in Physiology and Medicine in 1973. Currently, as large-scale data represented by genomic sequencing data are being introduced to biology, big behavioral data are also being introduced to ethology [2]. This new area of ethology, based on the analysis of big behavioral data by computer science, is termed "Computational Ethology", and has attracted much attention recently [3].

Here, I will discuss the advantages of computational ethological approaches from the viewpoint of the quantification of animal behavior. In order to examine theories and hypotheses about animal behavior, statistical analysis must be performed to investigate the significance of the hypothesis. For that, animal behavior must be quantified under some conditions. Count-based quantification, which measures the frequency (or the number of times) with which animals perform a specific behavior as observed by human eye, is the most popular quantification method due to its simplicity. However, count-based quantification suffers from two disadvantages.

The first disadvantage of count-based quantification is the loss of qualitative information on behavior (Fig. 1.1a). This is illustrated by the following example: when evaluating communication between two individuals, if the number of communications is counted when the distance between the two individuals is smaller than a given value, information about the speed of each individual is lost. If the speed has significance for

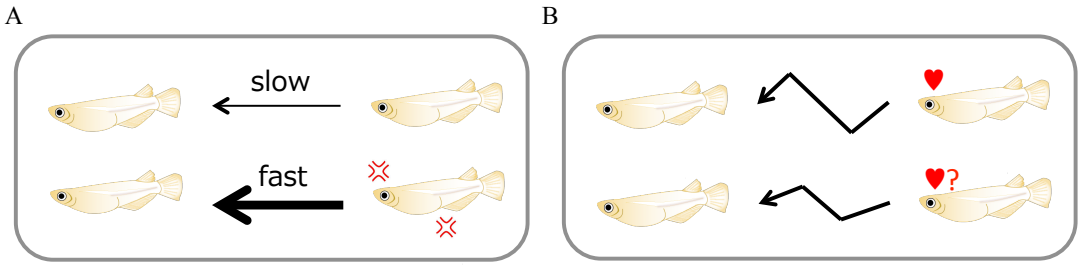


Fig. 1.1 Schematic illustration of defects caused by count-based quantification. (A) When one counts these two approach as same behavior, speed information drops out. (B) When one counts complex behavior, the subjectivities of observers influence count results.

communication between these two individuals, evaluation based solely on the distance-based index results in an incomplete understanding of the behavior begin studied. In the worst case scenario, this may result is incorrect interpretation of the behavior. The loss of speed information may be prevented by counting separately the communication with different speed. However, a large amount of information, such as approach angles or movement trajectories, would still be lost. As the classification criteria for avoiding loss of information increase, misclassification by human error is also expected to increase. Furthermore, as increasing classification criteria leads to an increase in the amount of data required for statistical analysis, long-term observation becomes necessary, and this study become more laborious and time-consuming.

The second defect of count-based quantification is that subjectivity of each observer influences the results (Fig. 1.1b). When one counts the frequency of complex behavior such as aggressive behavior and courtship behavior, unification of the evaluation criteria between different observers is difficult. In other words, when one counts a behavior that is composed of subtle movements, different observers may report different count results.

Video-based quantification, which quantifies behavior by video recording and analysis of the video data based on computer science, potentially overcomes these two defects of count-based quantification. This approach enables the simultaneous quantification of various parameters, such as velocity and direction, without loss of raw behavioral data. If necessary, researchers may obtain new parameters from the original movie by developing new measurement tools. In addition, this quantification method enables the analysis of long-term video data. Furthermore, as the usage of the same software and video data gives consistent results, the subjectivity of the data analyst does not influence the analysis. Therefore, computational Ethology, based on video-based quantification, is expected to become an important discipline in the investigation of animal behavior.

In addition to video-based quantification, logger-based quantification has also at-

tracted a large amount of attention recently. In this field of research, referred to as "Biologging Science", animal behavior is quantified by attaching loggers, such as GPS-based devices and accelerometers, to animals that are allowed to freely perform the relevant behavior [4]. Then, the instruments are retrieved and the behavior of interest is quantified. An advantages of logger-based quantification is that it enables the investigation of behavior in environments where direct observation may be difficult, e.g. in the deep-sea and the sky [5, 6]. In addition, non-behavioral information, such as air temperature and the wind direction, may additionally be obtained by using various types of loggers simultaneously. However, a disadvantage of logger-based quantification is that it has limited applicability in small animals, including numerous model organisms, because attachment of loggers to small animals is challenging. Moreover, unlike video-based quantification, obtaining information on the subtle movements of animals may be difficult. Although logger-based quantification is of great interest as a behavior quantification method, this thesis will focus on video-based quantification.

1.2 Video Tracking System for Quantification of Animal Behavior

In this section, I review previously developed animal tracking software. Tracking refers to the acquisition of movement trajectories of individuals via the computational recognition of each individual from video data. This task constitutes a most basic and important step in computational ethology [7]. While bioinformatics involves the computational analysis of molecular data, such as sequence data or protein structure data, the field of "bioimage informatics", which is concerned with the development of software for the analysis of biological image data or video data, has grown rapidly in recent years [8, 9, 10]. Object tracking is a frequently investigated subject in this research area, and numerous software products for tracking cells or nuclei have been developed [7, 11].

In the first step of a tracking algorithm, unnecessary background objects are removed by image processing in order to obtain pixel data related only to the animal of interest in each image frame. In this step, conventional image processing methods such as background subtraction and binarization are frequently used [12]. However, when experimental conditions are not suitable for image processing, e.g. in the inappropriate light conditions, this extraction step cannot be achieved solely by simple image processing. In such cases, refinement of the experimental conditions is a simpler, easier, and more accurate solution than the development of new and complex image-processing algorithm. For example, Simon *et al.* prevented overlaps between each individual in a *Drosophila* tracking system by using a chamber with sloped walls instead of vertical ones , thereby

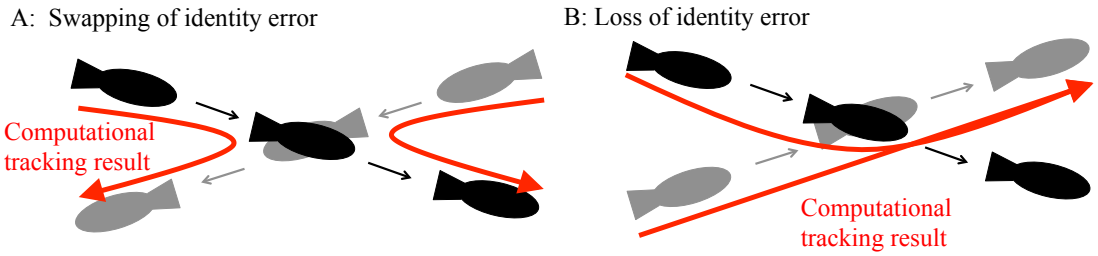


Fig. 1.2 Occlusion problem in multiple animal tracking. (A) Swapping of identity error. (B) Loss of identity error.

enabling the tracking software to recognize each individual easily [13]. However, the development of tracking systems for video recording in open fields is challenging, as it is difficult to control experimental conditions in such environments, which lowers the quality of the video data obtained.

In the second step, the tracking algorithm determines the movement trajectories of each individual by concatenating animal objects along the time sequence. When only a single animal is filmed, the trajectory of the animal may be obtained by a simple concatenation of extracted animal objects. However, when multiple animals are filmed, a problem related to the correspondence of animal identities between different frames arises. This problem may be resolved by ensuring correspondence so that the sum of the distance moved by each animal is minimal between successive frames. This method is based on the assumption that each position of the animal does not suddenly change between successive frames. However, contacts and overlaps between several animals may cause misidentification, such as "swapping of identity error" and "loss of identity error" (Fig. 1.2). This problem, termed the "occlusion problem", remains an unresolved challenge to the development of tracking software.

Some researchers have solved the occlusion problem by devising video recording conditions. The simplest solutions are embedding sensors into animals or physically marking each animal with different signs or colors [14, 15, 16]. While these methods prevent occlusion problem with high accuracy, there is a risk that physical interference with experimental animals influences the behavior of individuals. In addition, when the number of individuals is large, preparation of computationally distinguishable marks or colors is not easy. Another solution to the occlusion problem involves three-dimensional video recording by multiple video cameras [17]. In this method, although animals may overlap with each other when viewed from one camera angle, they should not overlap with each other when filmed from different camera angles. While this method also provides an effective solution to the occlusion problem, it is necessary to set up a transparent experimental tank and multiple video cameras whose viewing fields encompass the full

range of animal activity. Therefore, some researches put instruments in video view for investigating the reaction behavior to the instruments, but this three-dimensional video recording method cannot be applied to the researches [18, 19].

Other researchers have tackled the occlusion problem by developing new tracking algorithm. Delcourt *et al.* assumed that the motion state for each animal represents uniform linear motion, and assigned identities to each animal so that their movement trajectories were more similar to uniform linear motion [20]. Unfortunately, this method lacks high accuracy as animals frequently show movements that deviate from uniform linear motion to large extents. Recently, Prez-Escudero *et al.* developed idTracker, which computationally discriminates between animals on the basis of natural characteristic marks indistinguishable to the human eye [21]. When the resolution of video data is high, idTracker is able to accurately solve occlusion problems for various species such as zebrafish, mice, and flies. On the other hand, this method cannot be applied to species whose pattern on body surface patterns are unclear, for example the *himedaka* variety of *Oryzias latipes*.

In the final step of the tracking algorithm, the shape of each animal is assessed. Several tracking software products are capable of performing previous steps simultaneously with this step. This step may be omitted when only the movement trajectories are required for subsequent analysis. The development of general algorithm that targets every species is challenging, because animals exhibit a large diversity of morphologies. To date, a number of species-specific video tracking systems have been developed mainly for model organisms such as nematodes [22, 23, 24], mice [15, 25, 26], fruit flies [27, 28], ants [29], and fish [20, 30, 31].

This section concludes with a brief discussion of the execution time of tracking software. In many cases, tracking software and video recording are run separately, because the execution time of many tracking software is longer than the video recording time. On the other hand, several real-time tracking systems have been developed for interfering in animal behavior [32, 33]. For example, FlyMAD software is capable of targeting freely moving flies with an infrared laser using real-time tracking [32].

1.3 Methods for the Analysis of Tracking Data

After the movement trajectory of each animal is obtained, the tracking data are analyzed. Some researches only carry out general statistical tests of basic parameters such as inter-individual distance and velocity of each individual, whereas other studies utilize more sophisticated methods for the analysis of tracking data. In this section, I review three methods for the analysis of tracking data: behavioral annotation analysis, behavioral

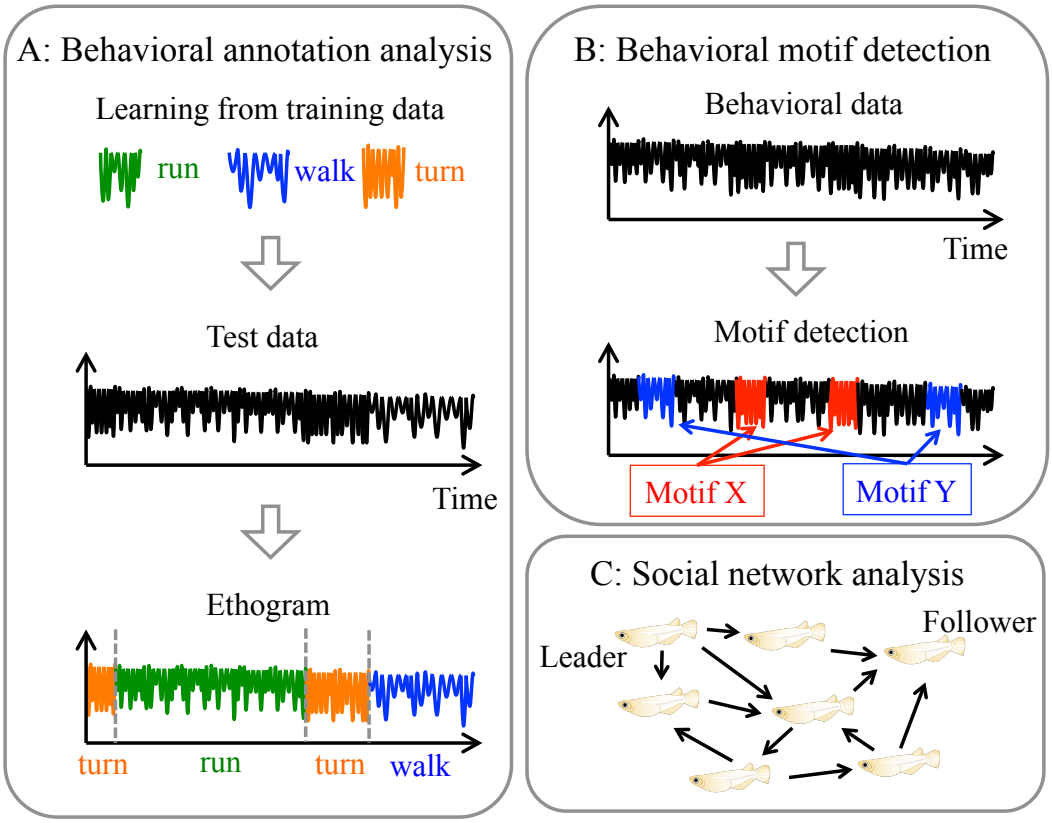


Fig. 1.3 Schematic illustration of sophisticated analytic method of tracking data. (A) Behavioral annotation analysis. (B) Behavioral pattern analysis. (C) Social network analysis.

pattern analysis, and social network analysis (Fig. 1.3). These methods are adaptations of supervised learning, unsupervised learning, and network science to tracking data analysis.

Behavioral annotation analysis involves the classification of animal behavior into categories such as "walk", "run", and "turn", and annotates the behavioral category of each animal along time sequences (Fig. 1.3A). When the data size of the video is small, this task may be performed manually. However, when the data size of video is large, manual annotation is extremely laborious. Numerous annotation systems based on supervised learning have been developed for the automatic annotation of animal behavior [26, 27, 28, 34, 35].

In behavioral annotation analysis, tracking results such as the velocity and shape of each animal are regarded as input vectors, and behavioral categories are regarded as output values. First, training datasets are constructed by manual annotation of small-sized datasets. Several support tools for manual annotation have been developed. For example, JAABA software enables the interactive annotation of behavior using a graphical

user interface [36]. Next, the classifier is learned from training datasets using conventional supervised learning methods such as support vector machine and random forest [37, 38]. Finally, behavioral categories of unannotated large-scale dataset are predicted by adapting the learned classifier to the dataset. One of the advantages of behavioral annotation analysis is that interpretation of analysis results is relatively easy, as behavioral categories are determined on the basis of the expert's knowledge and experiences. However, this is also disadvantage in that the detection of undefined behavior is difficult in principle. In addition, the subjectivity of each annotator has the potential to influence the construction of training datasets. In practice, disagreements of manual annotation results between different annotators are common [34]. Therefore, behavioral annotation analysis may lack objectivity, which is a big advantage of computational ethology-based approaches.

Behavioral pattern analysis involves the detection of characteristic and frequently appearing behavioral patterns from tracking data using unsupervised learning (Fig. 1.3B) [39, 40, 41, 42]. Unlike behavioral annotation analysis, this method possesses an advantage in that it is possible to detect unknown behavior and the subjectivity of the analyst does not affect the results. On the other hand, interpretation of detected behavioral motifs is not easy. In other words, the merits and demerits of behavioral annotation analysis and behavioral pattern analysis exhibit complementary relationships.

Social network analysis expresses animal groups as a network by regarding each individual and inter-individual relationship as a node and an edge, respectively. Then, the dominance hierarchy or social structure of animal groups is determined by applying network theory to the drawing network. Social network analysis was used in ethology before the introduction of big behavioral data. In a pioneering study, Croft *et al.* and Lusseau constructed a social network for guppies or dolphins using the mark-recapture method and individual recognition, respectively [43, 44]. In these studies, edges represent individual pairs belonging to same group. They revealed the existence of significantly familiar pairs that has numerous edges. In addition to the analysis of such basic network structures, big behavioral data enables the investigation of the dynamics of network structure or the type of a relationship between individuals. For example, Mersch *et al.* studied the time-series changes in ant social network structures based on long-time video recording [45]. They investigated time-series change in division of jobs, such as foraging and nursing for worker ants, and analyzed the relationships between the changes in network structure and job category for each ant. Nagy *et al.* discovered a hierarchy in pigeon flocks by describing pigeon social structure as a directed graph [46]. By assuming that followers change their direction of travel after the leader changes the direction, researchers detected leader-follower relationships in pigeon groups by correlation analysis

with time delay of traveling direction change. Although this study was performed using logger-based quantification, the analyticcal method may be applied to the analysis of tracking data obtained by video-based quantification [21].

1.4 Purpose of This Thesis

While bioinformatics for understanding animal behavior has flourished in recent years, there are still many unsolved problems. In this thesis, I especially grappled with the following three research tasks: 1) Solution of the occlusion problem described in section 1.2. 2) Development of analytic method of tracking data described in section 1.3. 3) Revealing the molecular mechanism of animal behavior based on other omics data.

The following chapters are organized as follows. Chapter 2 describes tracking software, called GroupTracker, which is a multiple animal tracking system that accurately tracks individuals even under severe occlusion. Chapter 3 shows bioinformatic analysis of *C.elegans* tracking data. Chapter 4 demonstrates that several RBPs related to neuronal disorder bind to their target molecules under specific RNA secondary structural contexts. In Chapter 5, conclusions of this thesis are presented with discussion and future work. A part of this thesis is based on the following publications written by the author and others: [47, 48].

Chapter 2

GroupTracker: Video Tracking System for Multiple Animals under Severe Occlusion

2.1 Introduction

In this section, I present a Gaussian mixture model-based, multiple animal tracking system that accurately tracks individuals even under severe occlusion. Severe occlusion occurs not only under typical experimental settings but also during interesting inter-individual behaviors such as courtships ([49]). Thus, most studies so far required laborious manual annotations of identities and positions of individuals, and the ability to perform large-scale systematic analyses is greatly inhibited.

Recently, the Gaussian mixture model has been adopted by several multiple animal tracking methods, where animal individuals are represented by components of a Gaussian mixture ([27, 28, 50, 15]). Through this approach, latent variables such as true positions of individuals are explicitly represented. The associated probability models and numerical methods are also well-established. Although a Gaussian distribution cannot represent, for example, bending shapes of a nematode, it has been successfully applied to many animals such as mice and fruit flies ([27, 15]). Nevertheless, methods adopting the Gaussian mixture model also suffer from the severe occlusion problem, because the maximum likelihood estimation of the Gaussian mixture model is theoretically an ill-posed problem under the condition where multiple components can overlap ([51]).

My key idea was the introduction of constraints to the eigenvalues of the covariance matrices of the Gaussian mixture components, by taking advantage of the fact that the size of each individual usually remains almost constant during a video sequence. I developed algorithm that effectively estimates the Gaussian mixture parameters under these additional constraints, and implemented a publicly available software tool named 'GroupTracker' (GROUP: Gaussian Reinterpretation of OcclUsion Problem).

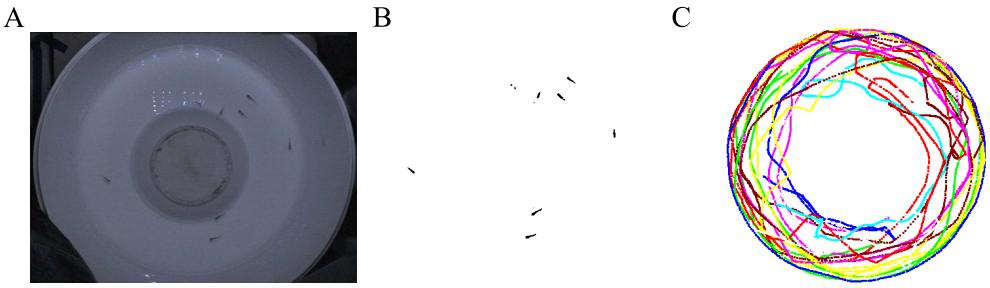


Fig. 2.1 (A) An image frame in a video sequence that contained eight individual medaka fish. (B) The same image frame, after the preprocessing step. (C) Tracks of eight individuals from a one-minute video segment. Colors represent different individuals.

2.2 Material and Methods

2.2.1 Video Sequence Dataset

Medaka fish (*Oryzias latipes*) was selected for demonstrative purposes in this study. As fish swim around in three dimensions and frequently overlap each other, they are suitable for evaluating multiple animal tracking system under occlusion conditions. It should be noted that, partly because of these characteristics, tracking systems for fish are underdeveloped compared with those for other organisms ([52]). Furthermore, medaka fish has been used as a model organism in many fields of animal sciences. It shows various interesting behaviors that involve inter-individual interactions such as schooling and aggressive behaviors ([53, 54, 55]), while rich resources are available for its neurobiology and genomics ([56, 57]).

Five ten-minute video sequences that recorded one, two, four, eight, and sixteen individuals were prepared. Medaka fish (Hd-rR strain) were hatched and bred in laboratory aquariums. In each case, equal numbers of female and male individuals (one female in the case of one individual) at six months of age (adult, body lengths $\approx$ 3 cm) were transferred to a white, opaque, cylindrical ring-shaped, plastic water tank (outer radius = 46 cm, inner radius = 24 cm, depth $\approx$ 4 cm, water temperature = 26°C; Fig. 2.1A). This shape of the tank enhanced the schooling behavior of medaka. A white polarized LED lamp (10.7 cm $\times$ 22.5 cm) located above the tank was used as the light source during video recording (Fig. 2.2). A high-definition digital video recorder (HDR-HC9 Sony Corp., Japan) was set approximately 140 cm above the water surface. A polarizing filter (VF-37CPKS, Sony Corp., Japan) was used to reduce light reflection. Videos were recorded in eleven-minute sequences during daytime (from 2pm to 5pm) using default

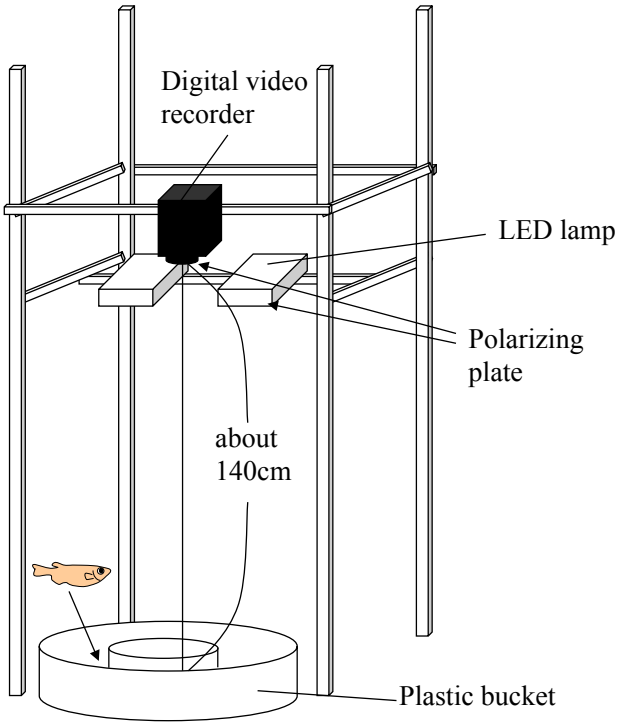


Fig. 2.2 A schematic illustration of the video-recording apparatus.

video settings. Blackout curtain was set up surrounding the entire apparatus to prevent external (human) interference. Final Cut Pro (Apple Inc., U.S.A.) was used to convert the videos into the Motion JPEG format (frames per second = 30, resolution of the image frames = 872×480). The first one-minute segment was deleted from each video sequence.

2.2.2 Method Overview

The method consists of three major steps: preprocessing, tracking, and post-processing. At the preprocessing step, objects outside of the movable areas (i.e., outside of the water boundary in case of fish) are removed and pixels composing the animal shapes are extracted from every image frame using conventional image-processing methods ([12]) (Fig. 2.3A). Then, the tracking step determines the precise position of each individual by fitting the Gaussian mixture model to the preprocessed image frames (Fig. 2.3B). The post-processing step consists of three minor steps: identity-swapping alert, identity-swapping correction, and head-direction determination. At the identity-swapping alert step, the system alerts the user to image frames that may contain identity-swapping errors. The identity-swapping correction step then automatically correct a portion of

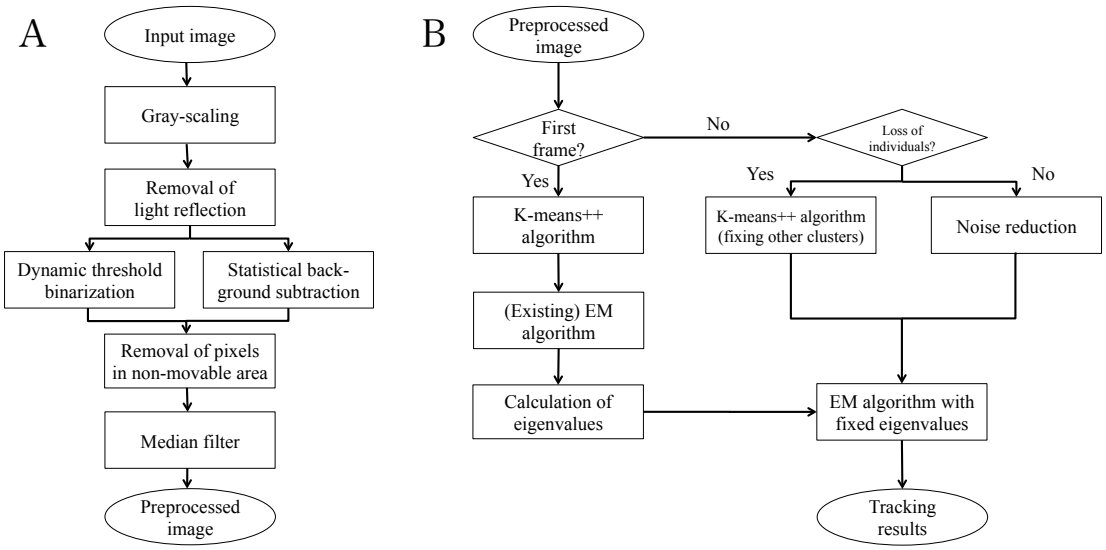


Fig. 2.3 Overviews of the (A) preprocessing step and (B) tracking step.

these errors. Finally, at the head-direction determination step, the direction of the head of each individual is determined in each image frame.

2.2.3 Preprocessing Step

At this step, first, every image frame in the video sequence is converted to 8-bit grayscale (into the 0—255 range from dark to bright by the NTSC conversion) and, to remove light reflection, any values higher than the threshold value of 100 is set to this value. Next, dynamic threshold binarization and statistical background subtraction are conducted to select pixels that likely constitute animal shapes. The former technique selects every pixel whose brightness value is lower than a dynamic threshold that is the average brightness value of the surrounding pixels (5×5 square pixels) plus or minus a user-defined value. Because medaka's body were darker than the surrounding environment, the user-defined value was set to -5. The latter technique selects every pixel whose brightness value is lower than a static threshold calculated as follows. Thirty image frames are collected at even intervals from the entire video sequence and, for each pixel coordinates, the mean μ and variance σ of the brightness values are calculated. The static threshold is then set to $\mu - 2\sigma$. Common pixels selected by both techniques are obtained and a median filter is applied to remove noises. Finally, the remaining pixel set is passed on to the tracking step.

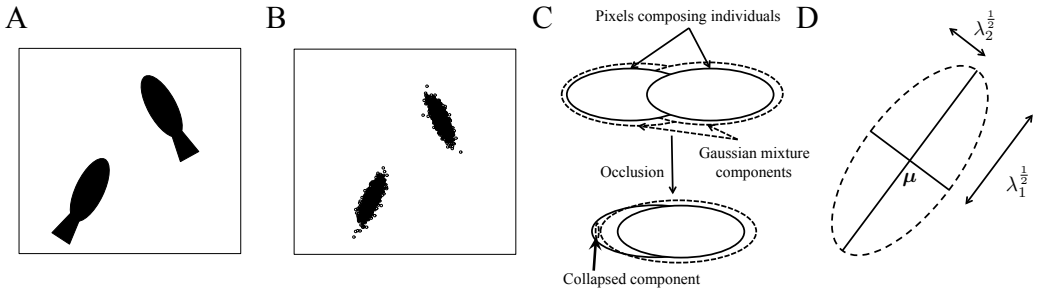


Fig. 2.4 (A) A schematic illustration of a preprocessed image frame. (B) Two-dimensional Gaussian mixture representation of the same image frame. (C) A schematic illustration of a case that two Gaussian mixture components overlap and one of component collapses to a single pixel. (D) A schematic illustration of the interpretation of eigenvalues and a covariance matrix of a Gaussian distribution. λ_i represents the two eigenvalues, while μ and the ellipse represent the mean value and a constant probability density contour, respectively.

2.2.4 Tracking Step

At this step, the two-dimensional Gaussian mixture model is applied to the preprocessed images (Figs. 2.4A and 2.4B) using the same number of mixture components as that of animal individuals. Hence, the mean value and covariance matrix of each component represent the position and shape of each individual, respectively.

First, the system processes the first image frame. K-means++ algorithm ([58, 59]) is applied to divide the pixels identified during the preprocessing step into K clusters, where K is the number of individuals. Because K-means++ algorithm can converge to local optima, the clustering process is repeated $R = 100$ times and the result with the smallest K-distance calculated as follows is chosen.

$$K\text{-distance} = \sum_{k=1}^{K} \sum_{\mathbf{x} \in C_k} \frac{1}{|C_k|} (d(\mathbf{x}, \mathbf{c_k}))^2$$

where $\mathbf{x}$ is a pixel coordinate, C_k is a cluster, $\mathbf{c_k}$ is the coordinate of its centroid, and $d(\cdot, \cdot)$ is the Euclidean distance. Then, the mean value μ_k and the covariance matrix $\mathbf{\Sigma}_k$ of each mixture component are set to $\mathbf{c_k}$ and K-distance $\times 0.1 \times \mathbf{I}$, where $\mathbf{I}$ is the identity matrix, respectively. The mixture ratio of each component π_k is set to $1/K$.

Then, for each successive image frame, the parameters of the Gaussian mixture distributions are estimated by the Expectation-Maximization (EM) algorithm ([51]) using the parameter estimate of the previous frame as the initial values. This relies on an assumption that the position and shape of an individual do not change abruptly between adjacent frames, which is generally true when the number of frames per unit time is

sufficiently large. It should be noted that this approach naturally preserves the identities of individuals in most cases.

In its original formulation, the EM algorithm described is as follows ([51]). The log-likelihood function is defined as:

$$\ln p(\mathbf{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}$$

where N is the number of pixels determined during the preprocessing step of each image frame and $\mathcal{N}$ is the Gaussian probability density function. The E step calculates $\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_l \pi_l \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}$, where z_{nk} indicates whether $\mathbf{x}_n$ belongs to the mixture component k and $\gamma(z_{nk})$ represents 'responsibility' that the mixture component k explains the observation $\mathbf{x}_n$. Then, the M step updates the parameters using $\gamma(z_{nk})$. The E and M steps are repeated until the likelihood function converges to a local maximum.

Nevertheless, this EM algorithm could not be applied to the current problem because the maximum likelihood estimation of the Gaussian mixture model is intrinsically an ill-posed problem if any two components can severely overlap ([51]) (Fig. 2.4C). In this case, a Gaussian mixture component can collapse to a single pixel $\mathbf{x}$ and the likelihood function can contain the term $\mathcal{N}(\mathbf{x}|\mathbf{x}, \boldsymbol{\Sigma}) = (2\pi|\boldsymbol{\Sigma}|^{\frac{1}{2}})^{-1}$, which diverges to infinity as $|\boldsymbol{\Sigma}| \to 0$.

Therefore, I developed a novel algorithm that overcomes this limitation. The key idea was to fix the eigenvalues of $\boldsymbol{\Sigma}_k$ since they represent the sizes of the individuals, which can be considered constant during a video sequence (Fig. 2.4D). If the eigenvalues are fixed, a Gaussian mixture component cannot collapse to a single pixel and $|\boldsymbol{\Sigma}|$ cannot approach 0. First, the original EM algorithm described above is applied to the first image frame and the eigenvalues of $\boldsymbol{\Sigma}_k$ are calculated. This requires that all animal individuals do not overlap in the first frame, though it is trivial to choose any frame that fulfills this condition in a video sequence. Then, the adapted EM algorithm that maximizes the likelihood function while fixing the eigenvalues is applied to the first and subsequent frames, using the eigenvalues calculated above as input. Note that the likelihood function does not change even if the eigenvalues are fixed; in other words, only the M step needs to be revised. Since the covariance matrix of a Gaussian distribution is a real symmetric matrix, I can choose the eigenvectors that form an orthonormal set ([51]). Given eigenvalues λ_{ik} and eigenvectors $\mathbf{u}_{ik}$, the covariance matrix is written by

$$\begin{aligned}
\boldsymbol{\Sigma}_k &= \lambda_{1k}\mathbf{u}_{1k}\mathbf{u}_{1k}^{\mathrm{T}} + \lambda_{2k}\mathbf{u}_{2k}\mathbf{u}_{2k}^{\mathrm{T}} \\
&= \begin{pmatrix} \lambda_{1k}\cos^2\theta_k + \lambda_{2k}\sin^2\theta_k & (\lambda_{1k} - \lambda_{2k})\sin\theta_k\cos\theta_k \\ (\lambda_{1k} - \lambda_{2k})\sin\theta_k\cos\theta_k & \lambda_{1k}\sin^2\theta_k + \lambda_{2k}\cos^2\theta_k \end{pmatrix}
\end{aligned}$$

Note that I can set $\mathbf{u}_{1k}$ to $(\cos\theta_k, \sin\theta_k)^{\mathrm{T}}$ and $\mathbf{u}_{2k}$ to $(-\sin\theta_k, \cos\theta_k)^{\mathrm{T}}$ $(0 \leq \theta_k < \pi)$, where $\theta_k \in [0, \pi)$ is the angle of the major axis of the Gaussian component.

The log-likelihood function can also be represented by using θ_k, λ_{1k}, and λ_{2k}. By calculating its partial derivatives with respect to θ_k and setting it to zero, I obtain the following equation:

$$\sum_{n=1}^{N} \gamma(z_{nk}) \left\{ \frac{\lambda_{2k} - \lambda_{1k}}{\lambda_{1k}\lambda_{2k}} \times \left(\frac{1}{2}(a_{1nk}^2 - a_{2nk}^2) \sin 2\theta_k - a_{1nk} a_{2nk} \cos 2\theta_k \right) \right\} = 0$$

where $(a_{1\mathrm{nk}}, a_{2\mathrm{nk}})$ is $(\mathbf{x}_n - \boldsymbol{\mu}_k)^T$. The solution of this equation is given by

$$\text{if } \sum_{n=1}^{N} \gamma(z_{nk})(a_{1nk}^2 - a_{2nk}^2) = 0 \Rightarrow \theta_k = \frac{\pi}{4} \text{ and } \frac{3\pi}{4}$$

$$\text{otherwise } \theta_k' = \frac{1}{2} \arctan\left(\frac{2\sum \gamma(z_{nk}) a_{1nk} a_{2nk}}{\sum \gamma(z_{nk})(a_{1nk}^2 - a_{2nk}^2)} \right)$$

$$\theta_k = \begin{cases} \theta_k' + \frac{\pi}{2} \text{ and } \theta_k' + \pi & (\theta_k' < 0) \\ \theta_k' \text{ and } \theta_k' + \frac{\pi}{2} & (\theta_k' \geq 0) \end{cases}$$

The two possible solutions represent the local maximum and local minimum. By selecting the one whose second order differential is negative, the solution for the local maximum is obtained and passed on to the next iteration of the EM algorithm.

When it comes to real datasets, animal individuals sometimes move too fast and the solutions to the EM algorithm from the previous frame could become inappropriate as the initial parameters. These 'loss of individual' events are detected by calculating the likelihood function for a mixture component k with the initial parameter values μ_k and $\boldsymbol{\Sigma}_k$. If the calculated likelihood is less than a threshold α, a round of K-means++ algorithm is performed by fixing the parameters of all other components, and μ_k and $\boldsymbol{\Sigma}_k$ are updated as described earlier. On the other hand, if no 'loss of individual' events are detected, noise reduction is then conducted where any pixel whose likelihood, according to the initial parameters, lies below a threshold β is regarded as noise and removed. In the current implementation, $\alpha = \beta = 10^{-15}$.

2.2.5 Post-processing Step

As described earlier, the tracking step preserves the identity of each individual across frames in most cases; however, identity-swapping errors may occur at frames that contain occlusion. This step alerts the user to them.

First, for each pixel $\mathbf{x}_n$ in each frame, this step finds $k1, k2 \in \{k | 1 \leq k \leq K\}$ that constitute the largest and second largest values of $\gamma(z_{nk})$, i.e., the top two mixture

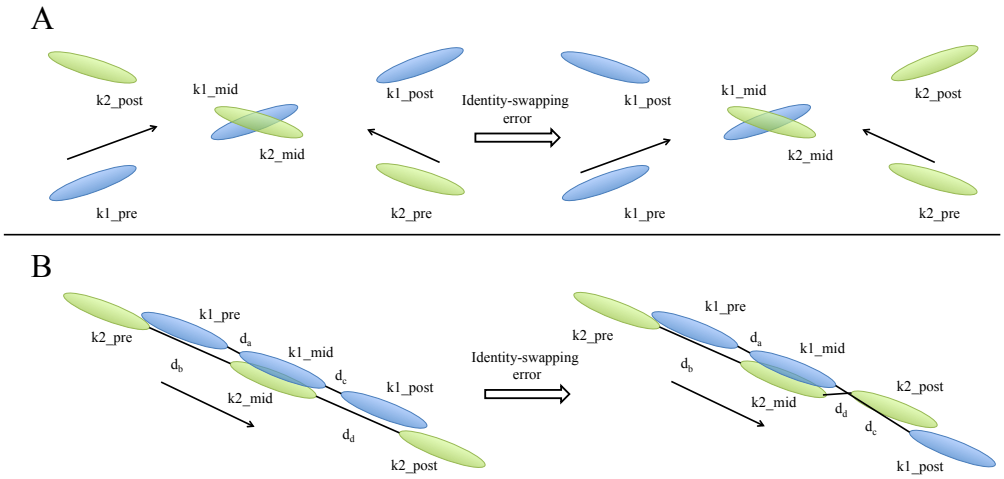


Fig. 2.5 Identity-swapping errors that are corrected at the identity-swapping correction step. (A) A case that sudden changes in directions of movement are detected. (B) A case that sudden changes in speed of movement are detected.

components that best explain $\mathbf{x}_n$. A $\gamma(z_{nk1})$ value less than a threshold $a = 0.7$ indicates that these components get close in that frame. In this case, a combination of the frame number, $k1$, and $k2$ are recorded. A series of successive frames allowing at most one-frame gaps with the same recorded component pair ($k1$, $k2$) are then grouped into an "incident" . Incidents spanning less than a threshold $b = 5$ frames are discarded to exclude potential false positives. In addition, the differences between angles θ_{k1} and θ_{k2} of the two recorded components are calculated for all frames within an incident and, if the minimum difference is larger than a threshold $c = \pi/6$, that incident is discarded. This is because large angle differences result in large Kullback-Leibler divergences between the mixture components that prevent identity-swapping errors. Finally, the remaining incidents are presented to the user as possible cases of identity-swapping errors.

Not every identity-swapping error can be corrected completely automatically. This step aims at correcting errors by detecting unnatural sudden changes in directions or speed of each individual's movement.

Given a user-defined value t_0 (default $t_0 = 10$) and an incident beginning at frame number f_{first} and ending at frame number f_{last}, the values of μ_k for the recorded components $k1$ and $k2$ at frames $(f_{\text{first}} - t_0)$, $\frac{1}{2}(f_{\text{first}} + f_{\text{last}})$, and $(f_{\text{last}} + t_0)$ are extracted. For simplicity, I defined them as μ_{k_pre}, μ_{k_mid}, and μ_{k_post}, respectively. To look for sudden changes in directions, the angle formed by μ_{k1_pre}, μ_{k1_mid}, and μ_{k1_post} and the angle formed by μ_{k2_pre}, μ_{k2_mid}, and μ_{k2_post} are examined. If both angles are smaller than a threshold $d = \pi/2$, the incident is judged as an identity-swapping error and corrected by re-swapping the identities mapped to the two components starting from

the frames of the incident (Fig. 2.5A). To detect sudden changes in speed, I first define the distances $d_{1\mathrm{pre}}$, $d_{1\mathrm{post}}$, $d_{2\mathrm{pre}}$, and $d_{2\mathrm{post}}$ as $|\mu_{k1\_\mathrm{pre}} - \mu_{k1\_\mathrm{mid}}|$, $|\mu_{k1\_\mathrm{mid}} - \mu_{k1\_\mathrm{post}}|$, $|\mu_{k2\_\mathrm{pre}} - \mu_{k2\_\mathrm{mid}}|$, and $|\mu_{k2\_\mathrm{mid}} - \mu_{k2\_\mathrm{post}}|$, respectively. Then, if $d_{\mathrm{score}} = |d_{1\mathrm{pre}} - d_{1\mathrm{post}}| + |d_{2\mathrm{pre}} - d_{2\mathrm{post}}| - |d_{1\mathrm{pre}} - d_{2\mathrm{post}}| - |d_{2\mathrm{pre}} - d_{1\mathrm{post}}|$ is greater than a threshold $e = 20$ pixels, the incident is judged as an identity-swapping error and corrected in the same manner (Fig. 2.5B).

At the tracking step, I introduced $\theta_k \in [0, \pi)$, which represents the angle of the major axis of the Gaussian component representing individual k. The upper limit was π instead of 2π, because the covariance matrices are diagonal and did not discriminate between the head and tail of an individual. At this step, the head directions of the individuals are explicitly determined and θ_k is updated to be in the range $[0, 2\pi)$.

First, because the head direction of the individual k does not abruptly change between successive image frames, frames are grouped if the differences between their θ_k values are less than $\pi/4$ (or greater than $3\pi/4$). Note that, if the individual k does not overlap with any other individuals during the entire video sequence, all frames usually formed a single group. This process is repeated for each individual. For each frame f in the frame group for individual k, the velocity $\mathbf{v}_k(f)$ is obtained as the difference vector between μ_k at frames $f - t_0$ and $f + t_0$. At the frame $f_{\max}$ where $|\mathbf{v}_k(f_{\max})|$ is maximized, the movement of individual k is assumed to be its head direction. Thus, if the difference between the angle of $\mathbf{v}_k(f_{\max})$ and θ_k at $f_{\max}$ is greater than $\pi/2$, to the value of θ_k at $f_{\max}$ is updated to $\theta_k + \pi$. Finally, π is added to θ_k at any frame so that the differences between θ_k from adjacent frames are always less than $\pi/4$ (or greater than $7\pi/4$).

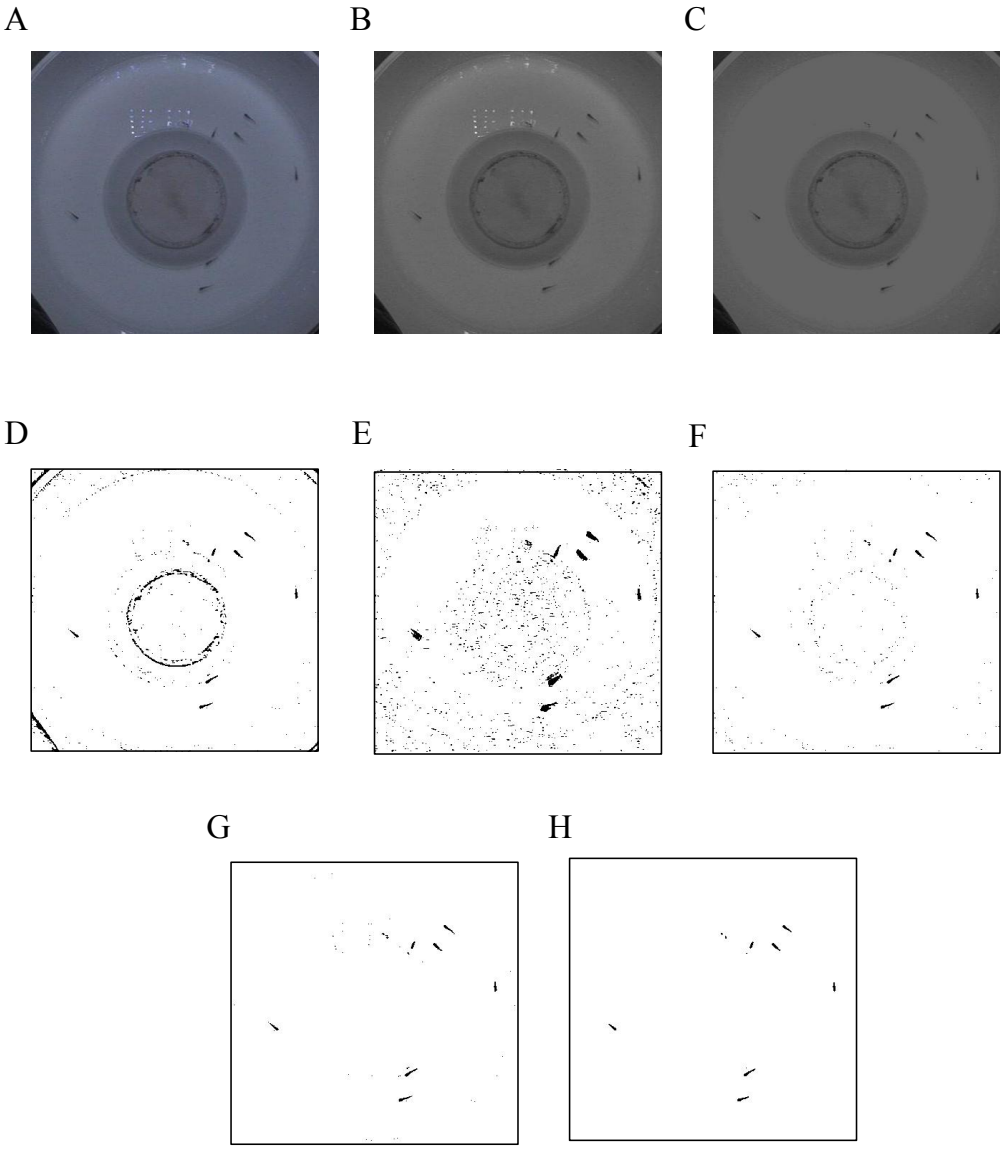


Fig. 2.6 Intermediate states during the preprocessing step. (A) An input raw image frame. (B) After gray-scaling. (C) After brightness-value thresholding. (D) After dynamic threshold binarization. (E) After statistical background subtraction. (F) The product set of (D) and (E). (G) After deleting pixels outside of the movable area. (H) After median-filter application.

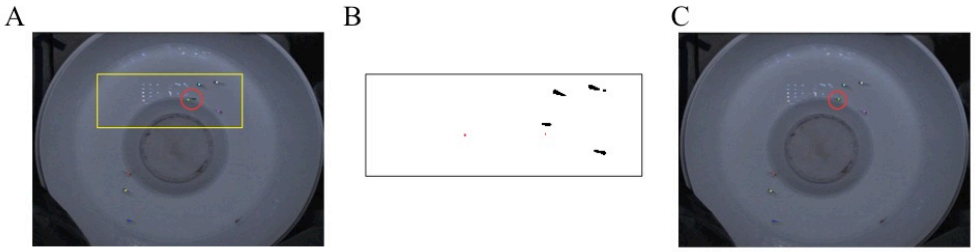


Fig. 2.7 (A) A tracking result without performing the noise reduction. The red circle indicates an individual whose position was inaccurately tracked. (B) The regions surrounded by the yellow rectangle in (A) after the preprocessing step. The red pixels are noise that caused the inaccurate position estimation. (C) The corrected result after performing the noise reduction.

2.3 Results

2.3.1 Application to Medaka Video Sequences

GroupTracker was applied to five ten-minute video sequences that recorded one, two, four, eight, and sixteen individuals. Figure 2.1B shows the final product of the preprocessing step of a raw image frame shown in Figure 2.1A (intermediate states are shown in Fig. 2.6). For most frames, the preprocessing step successfully identified pixels that constitute animal shapes. Any noise pixels that remained were removed by the noise reduction algorithm at the tracking step (Fig. 2.7). Figure 2.1C shows the movement tracks of the mean values of the eight Gaussian components during a one-minute video segment.

The present method requires several user-defined parameters that largely depend on the nature of the video data and the desired applications. Among these parameters, I note that the number of K-means++ trials R should be set sufficiently large to increase the accuracy of the subsequent EM algorithm. For example, setting $R = 1$ resulted in incorrect clusterings for the cases of eight and sixteen medaka individuals. On the other hand, setting $R = 100$ yielded 100% accuracy in every case (Fig. 2.8).

2.3.2 Evaluation of Identity-Swapping Errors

Since it is natural for the user to only manually check the alerted frames for identity-swapping errors, I prioritize sensitivity of the identity-swapping alert and report every frame that might contain the errors. To achieve this, I investigated the sensitivity of the system under various choices of threshold parameters a, b, and c (see Material and Methods). Manual inspection of all identity-swapping errors confirmed that the sensi-

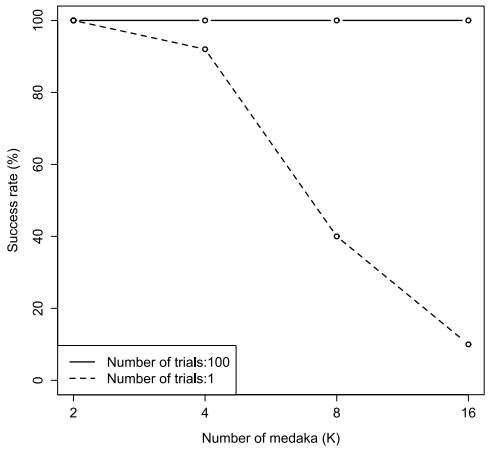


Fig. 2.8 Success rate of dividing pixels to K clusters at correct positions by the K-means++ algorithm. The x-axis represents the number of medaka. The y-axis represents the success rate. The solid line and broken line represent cases that the number of K-means++ trials were 100 and 1, respectively.

tivity was 100% for $a \geq 0.6$ (Fig. 2.9A , $b = 0$ and $c = \pi/2$ are fixed). By definition, the larger a was, the more frames were recorded during the first phase of the identity-swapping alert step (Fig. 2.9B). Nonetheless, the actual number of alerted incidents did not monotonically increase with a and in some instances reached its minimum value around $a = 0.7$ (Fig. 2.9C), probably because some incidents were mistakenly divided into smaller ones when the value of a was too small. Then, I fixed $a = 0.7$, and optimized the values for b and c so that the 100% sensitivity is maintained while minimizing the number of incorrectly reported incidents. I found that setting $b = 5$ frames and $c = \pi/6$ is appropriate because the minimum length of confirmed incidents was 9 frames and the maximum angle difference of the two individuals in an incident was 19° (Fig. 2.10A and 2.10B). With these threshold values, the precision of the identity-swapping alert step was improved from 0.033 to 0.143 while maintaining perfect sensitivity (Fig. 2.10C).

Table 2.1 Ratios of cases that identities were correctly preserved by the system.

	K=2	K=4	K=8	K=16	Total
Without correction step	**1.00**	0.72	**0.93**	0.88	0.88
With correction step	**1.00**	**0.80**	**0.93**	**0.96**	**0.92**

The bold figures indicate the better value in each case.

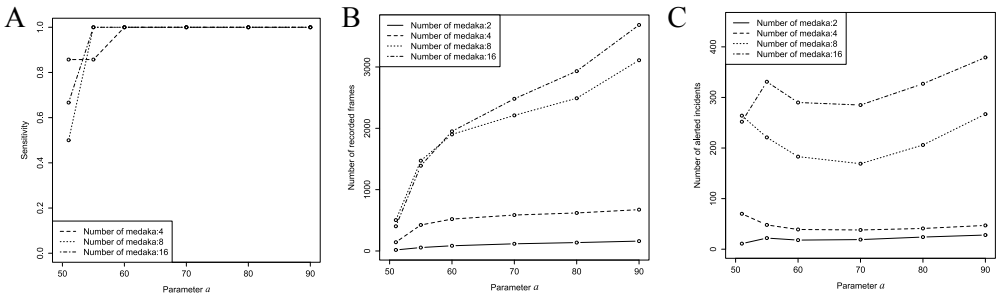


Fig. 2.9 (A) The sensitivity of alerting identity-swapping errors with regard to the parameter a. Note that $b = 0$ and $c = \pi/2$ in this panel. The x-axis represents the parameter a. The y-axis represents the sensitivity. (B) The numbers of recorded frames with regard to the parameter a. (C) The numbers of alerted incidents with regard to the parameter a.

On the other hand, at the identity-swapping correction step, precision becomes fundamental, i.e., false positives should be avoided. I found that setting $d = \pi/2$ and $e = 20$ pixels (see Material and Methods) safely maintains 100% precision. For incidents that did not contain identity-swapping errors, the angles defined by individual's changes in direction are well above $\pi/2$ (data not shown), and the maximum d_{score} was 14 pixels (Fig. 2.10D).

Next, I evaluated the performance of my system regarding the identity-swapping errors by calculating the accuracy of the correction step over all image frames where some fish individuals overlap. First, I extracted all image frames in which individuals are confirmed to be overlapped. In the present dataset, I could manually judge every identity-swapping case without ambiguity. Table 2.1 summarizes the accuracy of the systems with and without the identity-swapping correction step. Although the overall accuracy was already high (0.88) even without any corrections, utilizing the identity-swapping correction step improved it to 0.92. These results show that the system accurately preserves individual identities under occlusion.

2.3.3 Evaluation of Position and Angle Estimation

Next, I evaluated the accuracy of the estimated positions and angles, i.e., head-to-tail directions of fish individuals, by comparing them with the ground truth that was obtained as follows. I selected one frame per five seconds, i.e., 120 frames per one ten-minute video sequence, and manually measured the coordinates of the head, center, and tail of all individuals. The coordinates of the centers were regarded as the ground truth for the positions, and the angles of the difference vectors between the head and tail coordinates were regarded as the ground truth for the angles.

Tables 2.2 and 2.3 show the percentile errors of the estimated positions and angles,

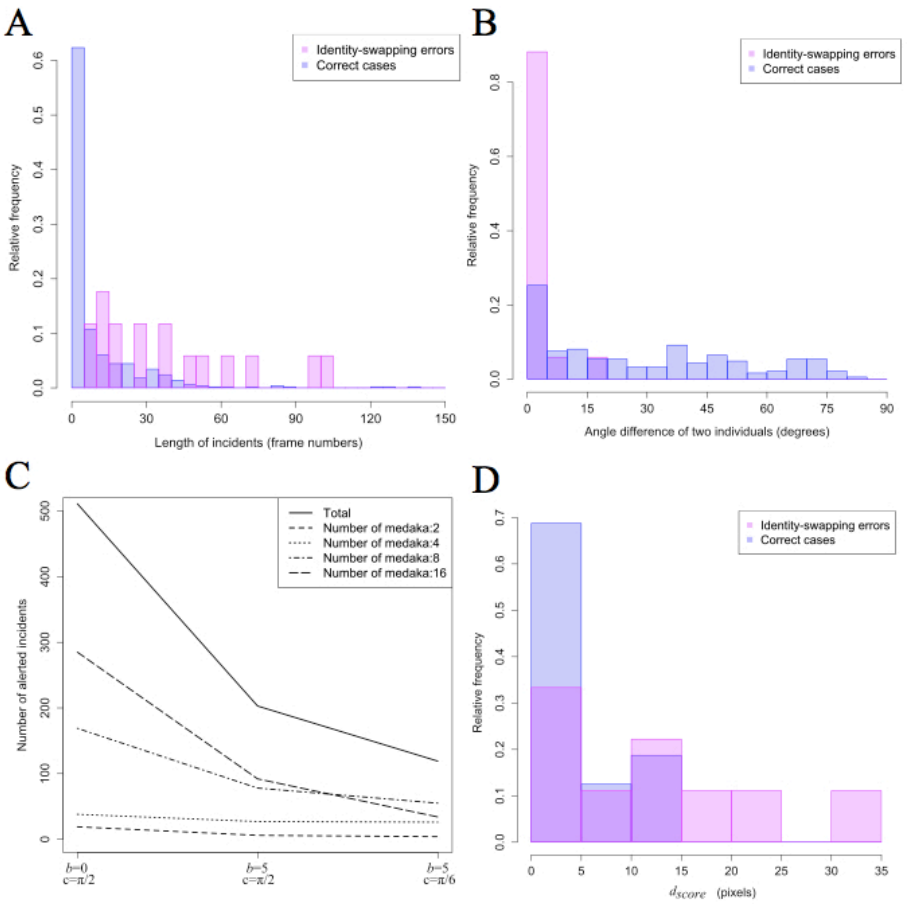


Fig. 2.10 (A) Lengths of incidents that contained identity-swapping errors and those did not. The y-axis represents relative frequencies in any case of two, four, eight, and sixteen. (B) Angle differences between the two individuals. (C) Number of alerted incidents with regard to the parameters b and c. (D) d_{score} of incidents that contained identity-swapped errors and those did not.

respectively. In all case, 90% of the estimation fell within the errors of 3.16 pixels in position and 8.31 degrees in angle, and the number of individuals has little effects on these errors. These results show that the estimated positions and angles agree well with the ground truth and also that this performance scales well with the number of individuals (Visualized in Fig. 2.11).

2.3.4 Evaluation of Running Time

I evaluated the speed of the system by comparing its running time with the time required for manual annotations. The computation was performed on an Intel(R) Core(TM) i5-3320M 2.6 GHz CPU with 4 GB of memory. Figure 2.12 shows the times required for the system, those required for manual annotation, and the relative efficiency. Overall, the system is 250-fold to 1800-fold faster than manual annotation and the efficiency increases

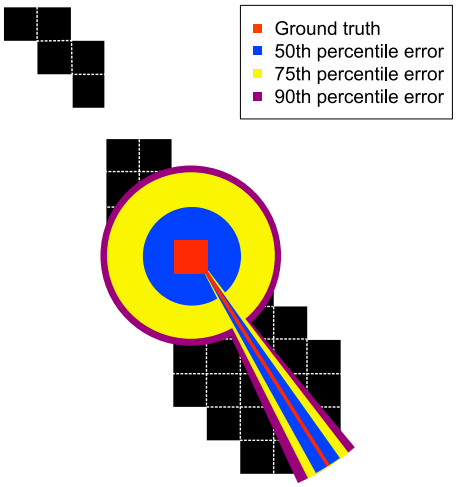


Fig. 2.11 Visualization of percentile errors in the position and angle estimation in the case that the number of individual was one. The black area represents pixels that constituted a medaka shape in an image frame. The red color represents the ground truth, while the blue, yellow, and purple colors represent the ranges of the 50th, 75th, and 90th percentile errors, respectively.

Table 2.2 Percentile errors in the estimation of the positions.

Percentile	K=1	K=2	K=4	K=8	K=16
25th	1.0	1.0	1.0	1.0	1.0
50th	1.0	1.0	1.41	1.41	1.41
70th	2.0	2.0	2.0	2.24	2.24
90th	2.24	2.24	2.83	3.16	3.0

(in pixels)

Table 2.3 Percentile errors in the estimation of the angles.

Percentile	K=1	K=2	K=4	K=8	K=16
25th	1.98	1.75	1.47	1.22	1.39
50th	3.68	3.35	3.37	2.84	2.79
70th	5.38	5.41	5.85	5.01	4.49
90th	7.52	7.99	8.31	7.43	7.65

(in degrees)

with the increasing number of individuals. Even in the case of sixteen individuals, the system required less than 90 minutes to process a ten-minute video sequence, and is thus time-efficient enough for practical uses.

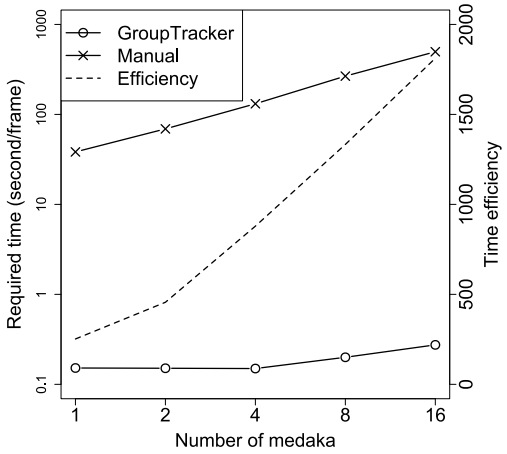


Fig. 2.12 Time required for the system and manual annotation. The x-axis represents the number of medaka individuals. The left y-axis represents time in second per frame in a logarithmic scale (solid lines). The right y-axis represents the time efficiency (dashed line).

2.4 Conclusion

In the present study, I developed a multiple animal tracking system called 'GroupTracker'. Its primary algorithm is based on an adaptation of the EM algorithm for Gaussian mixture model with fixed eigenvalues. This enables the system to accurately track individuals under severe occlusion. Recently, Mr. Rito Taekeuchi and Mr. Osamu Yamanaka have developed UMATracker, which is a GUI system to track multiple animals (http://ymnk13.github.io/UMATracker/). Tracking step of GroupTracker is implemented in this UMATracker software, and thus ethologists can use my algorithm with ease. In addition, as UMATracker system can change tracking algorithm to the other algorithm, software developers can compare their developed algorithm with GroupTracker algorithm on the same condition with ease.

I envision three future improvements and utilizations of GroupTracker. The first is speeding up of the system. Although the system already illustrated reasonable time-efficiency, further improvements may be needed, for example, for real-time tracking or tracking of a large flock of animals. This can be achieved by parallelization. In particular, the preprocessing step is clearly parallelizable because the processing of each frame is independent. The tracking and post-processing steps could also be parallelizable if an input video sequence is divided into segments separated by frames where all individuals are clearly identified. The second is the possibility of adopting more-sophisticated machine

learning techniques for the detection and correction of identity-swapping errors. Techniques such as the support vector machine have been widely utilized in bioinformatics fields ([60, 61]) and could improve the accuracy, sensitivity, and specificity of tracking. Last but not least, the third is the actual utilization of the system to gain novel biological knowledge. I aim at revealing unexplored social network structures ([46, 6]) and behavioral patterns ([39, 41]) behind animal interactions, which would provide insights into the high-order functions of their nervous systems.

Chapter 3

Bioinformatic analysis of postural change patterns for *Caenorhabditis elegans* mutants

3.1 Introduction

In this section, I present an analytic method for understanding *Caenorhabditis elegans* behavior based on unsupervised learning. *C. elegans* has been used as a model organism in many fields of biology because of its simple body plan and neural system. At present, various research resources are available for its molecular biology and neuroscience, for example, the complete genome sequence, a highly curated and integrated database, and the complete neuronal wiring diagram [62, 63, 64]. In addition to these rich resources, we can also utilize various tracking systems for quantifying *C. elegans* behavior automatically [22, 23, 24, 65, 66]. Therefore, this animal is one of the most suitable organisms to elucidate the molecular and neural mechanisms of animal behavior.

Worm posture is a key phenotype for revealing relationships between their behavior and the molecular mechanisms. This is because mutations of genes expressed in neuron changes their posture [67], and also their postural change patterns decide their movement trajectories [68]. Therefore, several *C. elegans* tracking systems can measure not only their locations but also their postures, and the analysis of the obtained large-scale worm postural dataset has been conducted. For example, Brown *et al.* detected frequently repeated postural change patterns of *C. elegans* by unsupervised learning analysis [39]. They revealed that feature vectors calculated from these postural change patterns provide sufficient information for classifying mutants whose responsible genes have related functions. As another example, Schwarz *et al.* revealed that worms show different postural change patterns as a responses to optogenetic stimuli [69].

However, it is not still unclear how postural change patterns of mutants are different from those of wild type (WT) strain. There are several possible behavioral cases. For example, if a mutant takes different posture set from WT, naturally the mutant should

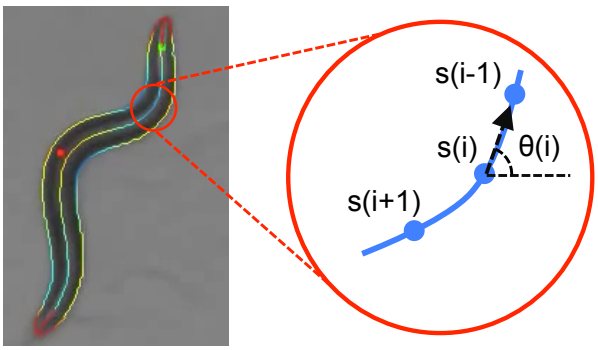


Fig. 3.1 Description of worm postures. Left figure shows a picture of N2 strain's contour and midline highlighted. This picture is a frame from *C. elegans* behavioral database. Right figure represents a zoomed illustration of the midline, and shows calculation method of angles along the midline.

show different postural change patterns from WT. On the other hand, if a mutant takes a same set of posture as WT but the postural change speed is faster than WT, the mutant also shows different postural change patterns from WT. Although these two examples mean clearly different behavioral patterns, but they were putted in the same category, and the classification of these behavioral patterns had not attracted the attention in previous research. By regarding these different behavioral patterns as different phenotypes, we can accurately infer the effect that genetic mutation influences worm behavior.

In this study, I analyzed mutant strains that show abnormal postural change patterns, and revealed the cause in behavioral level. I classified these behavioral reasons into four categories; the usage of different postural set, the frequency change of quiescence behavior, the change of behavioral speed, and taking the novel postural change patterns. I firstly calculated posture occurrence probabilities and posture transition probabilities for 322 *C. elegans* strains using template posture set, which was obtained by binning postural space. Then, I detected some mutant strains that show similar posture occurrence probabilities to N2 but different posture transition probabilities from N2. Finally, by investigating the distribution of postural change speed for each strain, I revealed the cause of the abnormal behavioral pattern for these mutant strains.

3.2 Methods

3.2.1 Data preparation

I downloaded worm behavioral feature dataset from the *C. elegans* behavioral database [70]. This dataset consisted of 9975 individual worms covering 338 strains (21 wild

types and 317 mutants), and each individual worm data consisted of pre-calculated 702 behavioral features such as the velocity and the orientation. The experimental condition is that hermaphroditic worms are freely crawling on the surface of agar plates with food. I focused on only "Eigen Projections" feature, which represents a worm posture by a few value. The "Eigen projections" feature was pre-calculated from the movie data as follows. Firstly, midline of worm shape was obtained by image processing for each image, and 48 angles were measured along the midline (Fig. 3.1 shows the illustration). Next, these angles were normalized so that the mean value of these 48 angles is zero in order to represent these angles independently of the worm's orientation. Then, principal component analysis was conducted against pooled angle data of multiple individual worm data of N2 strains, and some principal components were extracted. In this research, four principal components were extracted because even only these four values can reconstruct the worm posture with high accuracy (92%) [39, 71]. As a result, a worm posture and the time-series change were represented by four principal components and the time series of these four values, respectively. The details of the calculation method were given in the database paper [70]. This eigenvalue representation of animal shapes has been widely used to characterize the dynamics of animal locomotion [42, 72, 73].

I excluded individual worm data that met one of the following three criteria from the analysis; (1) The video length is shorter than 890 seconds or longer than 910 seconds. (2) The percentage of gap frame in all video frames is larger than 40 percentage. (3) The number of individuals belonging to the strain is smaller than 5. The second criterion was adopted because the "Eigen Projections" feature includes gap frames derived from tracking failure, and the large gap percentage may have bad effects on the analysis results. The distribution of gap percentage in all individual worm data is shown by Fig. 3.2A. As a result, I obtained 322 strains dataset (20 wild types and 302 mutants) consisting of 8769 individual worm data. As data pre-processing, all gap frames were linearly interpolated. In addition, frames per second (fps) of all individual worm data were downsampled and unified to 5 fps. This is because the original dataset includes individual worm data with various fps (Fig. 3.2B), and individual worm data with different fps cannot be directly compared with each other.

3.2.2 Template posture detection algorithm

To calculate posture occurrence probabilities and posture transition probabilities for each strain, I firstly obtained template posture set by binning the postural space. Then, all posture data were assigned to any of template posture, and worm postural change patterns were transformed into sequences of template postures. As binning method of postural spaces, I evaluated the performances of two methods that are K-means cluster-

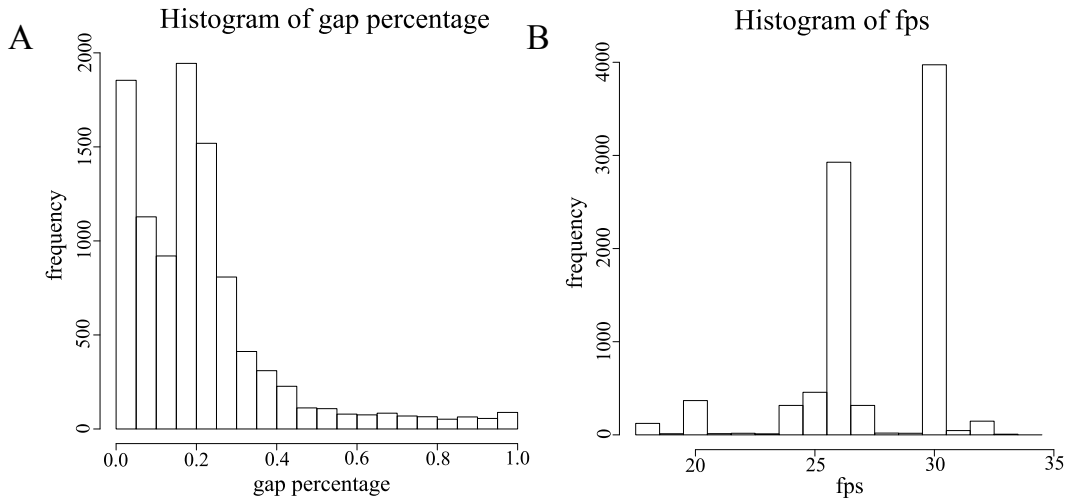


Fig. 3.2 (A) The distribution of gap percentage in the dataset. The x-axis and the y-axis represent the gap percentage and the frequency, respectively. (B) The distribution of fps in the dataset. The x-axis and the y-axis represent the fps and the frequency, respectively.

ing and Gaussian Mixture Model (GMM) [51]. Note that K-means clustering was used by previous research in order to bin the postural space [69].

K-means-based binning method was conducted as follows. First, 1% of postural data was sampled from all postural data in order to speed up parameter estimation. The number of sampled postural data was 385,790. Then, K-means clustering was applied to the pooled postural dataset, and the model parameters were estimated by Lloyd algorithm [58]. The initial parameters were estimated by K-means++ algorithm [59]. I regarded the centroid of each cluster after the convergence as a template posture. The number of K was set to 90, 44, 95, or 459 . 90 is the number that was used by previous research [69], and the other numbers were obtained by GMM-based binning method, which will be described later. After parameter estimation, the cluster assignment of the remaining 99% of postural data was conducted using estimated parameter.

GMM-based binning method was conducted as follows. First, data sampling was conducted like K-means-based binning method. Then, four-dimensional GMM was fitted to the pooled postural dataset, and the model parameters were estimated by Factorized Asymptotic Bayes (FAB) algorithm [74, 75]. FAB algorithm automatically selects the number of mixture component based on Factorized Information Criteria (FIC), which can be applied to the mixture model unlike conventional information criteria such as Bayes Information Criteria. FAB algorithm is different from conventional EM algorithm in only two of the shrinkage step and the calculation formula of E-step. More specifically, after the modified E-step, this algorithm shrinks the components whose mixture ratio is

smaller than a given threshold ϵ. In this analysis, ϵ was set to 0.01, 0.005, or 0.001. The initial parameters were estimated by K-means++ algorithm [58, 59]. I regarded the mean value of each Gaussian distribution after the convergence as a postural motif, and finally obtained 44, 95, and 459 postural motifs when ϵ is 0.01, 0.005, and 0.001, respectively. After the parameter estimation by FAB algorithm, the responsibility calculation of the remaining 99% of postural data was conducted using estimated parameter.

3.2.3 Calculation of posture occurrence probabilities and posture transition probabilities

Posture occurrence probabilities for each individual worm were calculated as follows. When K-means algorithm and GMM algorithm were used as binning method, each posture occurrence frequency of each individual worm were counted as the assigned number to each cluster and the summation of responsibilities of each cluster, respectively. Then, the frequencies were normalized as the posture occurrence probabilities. In addition, posture occurrence probabilities for each strain were defined as the average of posture occurrence probabilities of all individuals belonging to the strain.

Posture transition probabilities from a template posture i to a template posture j for each individual worm were defined as follows:

$$\text{transition probability}(i,j) = \frac{1}{N}\sum_{t} r_{t-1,i} r_{t,j}$$

where $r_{t,i}$ and N represent responsibility of template posture i at frame t on the movie and the number of frame, respectively. Finally, posture transition probabilities for each strain were defined as the average of posture transition probabilities of all individuals belonging to the strain.

3.2.4 Quantification of the difference of postural patterns and postural change patterns between N2 and the other strain

To quantify the difference of postural patterns and postural change patterns between N2 and the other strain, I calculated the Jensen-Shanon Divergence (JSD) of posture occurrence probabilities and posture transition probabilities between N2 and the other strain, and I termed these scores as $\mathrm{JSD}_{\mathrm{oc}}$ and $\mathrm{JSD}_{\mathrm{tr}}$, respectively. In addition, I calculated the difference between $\mathrm{JSD}_{\mathrm{oc}}$ and $\mathrm{JSD}_{\mathrm{tr}}$ for each strain, and termed this score as JSD Difference (JSDD). Note that $\mathrm{JSD}_{\mathrm{oc}}$ of a strain is equal to smaller than $\mathrm{JSD}_{\mathrm{tr}}$ of the strain (See Appendix).

3.2.5 Evaluation criteria of template posture detection algorithm

I assessed the performance of template posture detection algorithm on the basis of intra-strain consistency performance, which evaluates that posture occurrence probabilities of

Table 3.1 Ratios of strains whose posture occurrence probabilities were significantly similar to each other by each binning method.

Binning method	Parameter	Intra-strain performance
K-means	$K = 44$	207/322
	$K = 90$	203/322
	$K = 95$	213/322
	$K = 459$	205/322
GMM	$\epsilon = 0.01$	238/322
	$\epsilon = 0.005$	242/322
	$\epsilon = 0.001$	229/322

individual worms belonging to the same strain are similar to each other. More specifically, intra-strain consistency performance was evaluated as follows. Firstly, I regarded the set of posture occurrence probabilities of all individual worms belonging to the evaluated strain as a positive dataset. Next, I randomly sampled the individual worms belonging to the different strains as many as the positive dataset, and regarded the set of posture occurrence probabilities of sampled strains as a negative dataset. Then, I calculated a positive score and a negative score for an individual worm i as follows:

$$\text{positive score}(i) = \frac{1}{|P| - 1} \sum_{j \in P, j \neq i} d(x_i, x_j)$$

$$\text{negative score}(i) = \frac{1}{|N|} \sum_{j \in N} d(x_i, x_j)$$

where P, N, d, and x represent the positive dataset, the negative dataset, a dissimilarity function and a posture occurrence probability, respectively. As the dissimilarity function, I used JSD. In short, a positive score and a negative score represent an averaged dissimilarity between the positive datum and the other positive data, and the positive datum and negative data, respectively. After the positive and negative scores of all individual worms in the evaluated strain were calculated, I computed P value using the one-sided Wilcoxon-Mann-Whitney test against a set of positive scores and a set of negative scores. This P value computation was conducted against all strains. Benjamini-Hochberg FDR approach was used for multiple testing ($q < 0.05$) [76].

3.3 Results

3.3.1 Performance of template posture detection algorithm

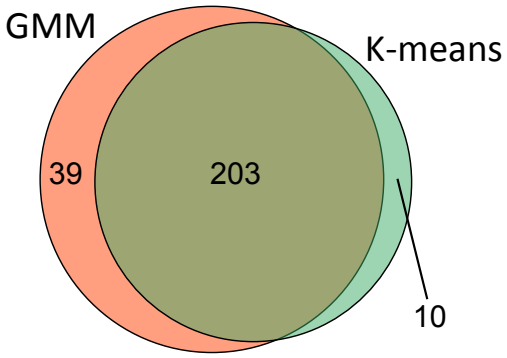


Fig. 3.3 The Venn diagram that represents relationship between intra-strain consistency result of K-means algorithm (K = 95) and that of GMM algorithm ($\epsilon = 0.005$). Red and Green color shows K-means algorithm and GMM algorithm, respectievly.

To obtain a set of template postures, I binned postural space by clustering method. To select more better clustering method, I evaluated intra-strain consistency performances of two clustering methods that are K-means algorithm and GMM algorithm (Table 3.1). This table shows that GMM algorithm achieved better performance than K-means algorithm. The parameter K and ϵ did not have a strong impact on intra-strain consistency performances. Next, I investigated whether strains that show significance by these two methods are different from each other. Fig. 3.3 shows Venn diagram that represents the relationship between intra-strain consistency results of K-means algorithm and that of GMM algorithm. As a result, almost all strains that show significance by K-means algorithm also show significance by GMM algorithm. These results suggested that GMM algorithm is superior method to K-means algorithm as detecting template posture. Therefore, I used GMM algorithm ($\epsilon = 0.005$) in the following analysis.

3.3.2 Analysis of $\mathrm{JSD_{oc}}$, $\mathrm{JSD_{tr}}$, and JSDD for each non-N2 strain

Next, to investigate whether each non-N2 strain takes different postural patterns and postural change patterns from N2 strain, I calculated $\mathrm{JSD_{oc}}$ and $\mathrm{JSD_{tr}}$ for all non-N2 strains. Fig. 3.4 shows the relationship between two JSDs. As overall trends, the relationship was approximately linear, and $\mathrm{JSD_{tr}}$ was small when $\mathrm{JSD_{oc}}$ was small.

In order to estimate whether the reason why mutants show large $\mathrm{JSD_{tr}}$ is "the usage of different postural set", I calculated the JSDD for all non-N2 strains. The large JSDD means that the strain shows abnormal postural transition patterns independent from the difference of the usage of postural set. In this study, in order to focus on the strains that show small $\mathrm{JSD_{oc}}$ but large JSDD, I excluded strains that have larger $\mathrm{JSD_{oc}}$ than 0.1

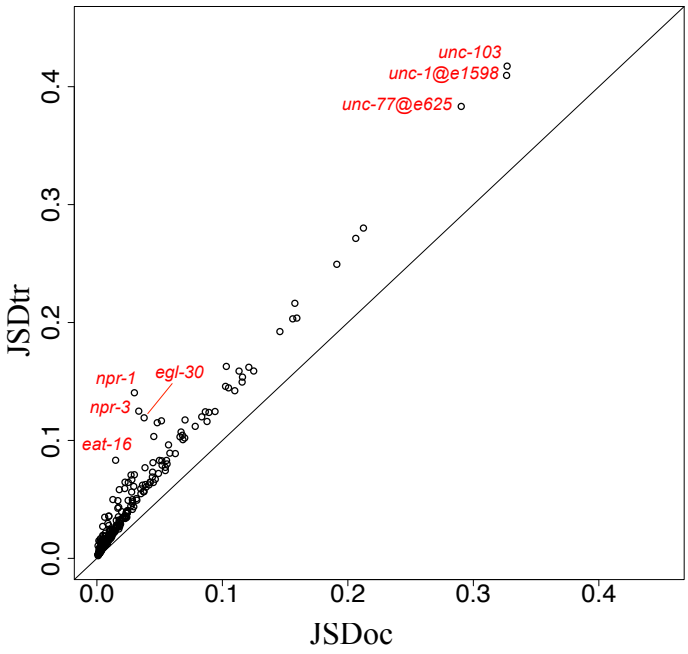


Fig. 3.4 The relationships between JSD_{oc} and JSD_{tr}. The x-axis corresponds to the JSD_{oc}, whereas the y-axis is the JSD_{tr}

Table 3.2 JSD_{oc}, JSD_{tr}, and JSDD of top 5 strains that have large JSDD.

strain	JSD_{oc}	JSD_{tr}	JSDD
npr-1	0.0299	0.1404	0.1106
npr-3	0.0333	0.1249	0.0915
egl-30	0.0376	0.1192	0.0815
eat-16	0.0149	0.0831	0.0683
lon-2	0.0481	0.1150	0.0669

in the following analysis. Fig. 3.5 shows the distribution of JSDD. Although many strain showed small JSDD, there were some strains that have large JSDD (Table 3.2).

Interestingly, both the strain with the largest JSDD (*npr-1* mutant) and the strain with the second largest JSDD (*npr-3* mutant) mutated neuropeptide receptor (npr) gene, but the other npr mutant strains did not show large JSDDs (Table 3.3). In addition, JSD of posture occurrence probabilities and posture transition probabilities between *npr-1* and *npr-3* mutants were 0.0003 and 0.0049, respectively. These differences were quite small. These results suggested that *npr-1* gene and *npr-3* gene have a similar function with each other, and a different function with the other npr genes in behavioral level. I hypothesized that the behavioral similarity between *npr-1* and *npr-3* was caused by

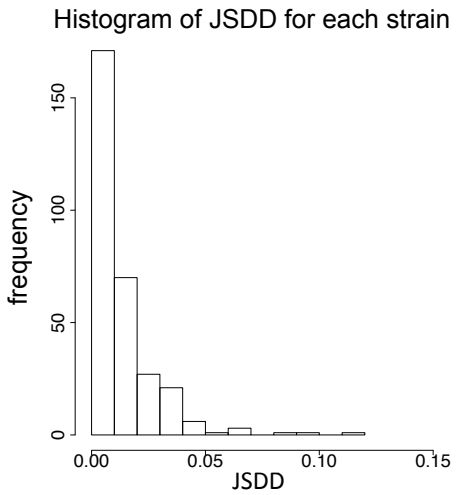


Fig. 3.5 The distribution of JSDD for each strain. The x-axis and the y-axis represent the JSDD and the frequency, respectively.

the sequence similarity of these two sequences, and thus I conducted pairwise alignment analysis using MAFFT version 7.271 with default option (Fig. 3.6A) [77]. This alignment result suggested that the relationship between *npr-1* gene and *npr-3* gene is paralogous. On the other hand, *npr-1* gene and *npr-2* gene alignment results show these two genes are also paralogous genes altough these two gene mutants did not show behavioral similarity (Fig. 3.6B). Therefore, I concluded that the sequence similarity between *npr-1* and *npr-3* cannot explain the behavioral similarity between *npr-1* mutant and *npr-3* mutant. There have been many researches about behavior of *npr-1* mutants, and several *npr-1* functions such as social behavior and feeding have been revealed [78, 79]. In addition, it has been known that the locomotion of *npr-1* mutants is more active than that of N2 strain[80, 81]. On the other hand, few researches focused *npr-3* mutants [82].

Both *egl-30* gene and *eat-16* gene are components of G protein Gαq signaling pathway, and EAT-16 negatively regulates EGL-30 directly [83, 84]. In addition, previous researches reported mutants with loss of *egl-30* function decreased the activity. The mutant alleles of *egl-30* and *eat-16* mutant used in this research were *ep271* and *sa609*, and it is known that these alleles are activation and reduction of function alleles, respectively [85, 84]. Therefore, *egl-30* and *eat-16* mutants in this research should show similar behavioral phenotypes of active locomotion. Actually, JSD of posture occurrence probabilities and posture transition probabilities between *egl-30* and *eat-16* mutants were both small values (0.0201 and 0.0356).

LON-2 is a glypican family of heparan sulfate proteoglycans, and the mutant shows longer body length than N2 [86]. Previous reseaches mentioned that this postures of mutant could not be very captured by N2-derived eigenworms because of the abnormal

Table 3.3 The list of JSDD of npr mutants.

strain	JSDD
npr-1	0.1106
npr-3	0.0915
npr-20	0.0208
npr-9	0.0195
npr-10	0.0175
npr-12	0.0110
npr-11	0.0098
npr-5	0.0062
npr-8	0.0055
npr-7	0.0045
npr-4	0.0035
npr-2	0.0028
npr-13	0.0018

body length [39]. The cause of this large JSDD may be derived from not biological reason but the less eigenworm fit, and thus I did not analyzed *lon-2* strain in the following analysis.

A

```
npr-1           MEVENFTDCQV---YWKVYPDPSQSIYAIVPFLTVYLFLFFLGLFGNVTLIYVTCSHKAL
npr-3           M--EGGRNCVMTVQQWQPEYNDMNQIRAIFSLL--YLLVWVGAIVGNTLVLYVLTFNQVS
                *  *.  :* :    *:   :  :.* **..:*  **:::. .:.**. ::**   ::.

npr-1           LSVQNIFILNLAASDCMMCILSLPITPITNVYKNWYFGNLLCHLIPCIQGISIFVCTFSL
npr-3           LSVRTVFVGCLAGSDLLMCLFSLPITAISIFSRVWVFPAIFCKLIGVFQGGTIFVSSFTL
                ***:.:*:  **.** :**::*****.*: . : * *  ::*:**  :** :***.:*:*

npr-1           GAIALDRYILVVRPHSTPLSQRGAFLTTVLLWILSFVVTLPYAFNMQMIEYTEERICGYF
npr-3           TVIALDRCVLILRPNQEIVNFPRAVFIVFCIWLLGYSLALPVGIYSDIAVYDE--ICGTF
                 .***** :*::**:.  :.   *.: .. :*:*.: ::** .:  ::  * *  *** *

npr-1           CTEKW-----ESAKS--RRAYTMIVMLAQFVVPFAVMAFCYANIVSVLSKRAQTKIRKMV
npr-3           CEENWPDFNPDTGRSGIRRAYGLSVLVLQFGIPALISSICYWMISRVMSDQLARRRGHNI
                * *:*     ::.:*  **** : *:: ** :*  : ::**  *  *:*.:   :  : :

npr-1           ERTSALESSCAFPSHGLEQYENELNEFLDKQEKEKQRVVLQNRRTTSILVTMVVWFGITW
npr-3           RPES-----------------------------ETKLVNRKTRANRMMIVMVVGFVLAW
                .  *                             : ::* :: *:. :::.*** * ::*

npr-1           LPHNVISLIIEYDDTQSFFRLYGRDDYDISYLLNLFTHSIAMSNNVLNPVLYAWLNPSFR
npr-3           MPFNAVNL---YRDLFGISKWYS--------TVFALCHVCAMCSAVLNPIIYSWFNPQFR
                :*.*.:.*   * *  .: : *.         :  : *  **.. ****::*:*:**.**

npr-1           QLVIKTYFGDRRKSDRIINQTSVYKTKIVHDTKHLNGRAKIGGGGSHEALKERELNSCSE
npr-3           QSITTLFKGTDEA--RLIKKKPQSTSKMVSYPTNFS-----------EIRKETEIASTKT
                * : . : *  .   *:*::..  .:*:*  ..::.           *  ** *: * .

npr-1           NLSYHVNGHTRTPTPEVQLNEVSSPEISKLVAEPEELIEFSVNDTLV
npr-3           KITIAENDY-------------------------------RAGDQLL
                :::   *.:                                 ..* *:
```

B

```
npr-1           -------------------------MEVENFTDCQVYWKVYPDPSQSIYAIVPFLTVYLF
npr-2           MLRQSDGTRMLQEMRRKLIQLHSSQMINETEETCDRYIDKHPDMTNEPTVLVTFSLLYLH
                                         *  *.   *: * . :** ::.  .:*.*  :**.

npr-1           LFFLGLFGNVTLIYVTCSHKALLSVQNIFILNLAASDCMMCILSLPITPITNVYKNWYFG
npr-2           IFLLGILGNSAVLYLTMKHRQLQTVQNIFILNLCASNVLMCLTSLPITFITNVYKQWFFS
                :*:**::** :::*:* .*: * :**********.**: :**: ***** ******:*:*.

npr-1           NLLCHLIPCIQGISIFVCTFSLGAIALDRYILVVRPHSTPLSQRGAFLTTVLLWILSFVV
npr-2           SPVCKLIPLVQGASIFVSTFSLSAIALDRYNLVVRPHKQKLSSRSAMMIALLIWVISVVV
                . :*:*** :** ****.****.******* ******.  **.*.*:: ::*:*::*.**

npr-1           TLPYAFNMQMIEYTEERICGYFCTEKWESAKSRRAYTMIVMLAQFVVPFAVMAFCYANIV
npr-2           CMPYGWYMDVEKL--NGLCGEYCSEHWPLAEVRKGYTFLVLITQFLFPFATMAFCYYNIF
                 :**.: *:: :   : :** :*:*:*  *: *:.**::*:::**:.***.***** **.

npr-1           SVLSKRAQTKIRKMVERTSALESS--CAFPSHGLEQYENELNEFLDKQEKEKQRVVLQNR
npr-2           SRLRQRVETKLKKLSERSQLLENSTTCGTTNHIVSINAEAVQNGL--ENKQRLAVLAQQR
                * * :*.:**::*: **:. **.*  *. ..* :.   : ::: *  ::*::  *: *:*

npr-1           RTTSILVTMVVWFGITWLPHNVISLIIEYDDTQSFFRLYGRDDYDISYLLNLFTHSIAMS
npr-2           RTTTILSCMVLLFAFTWLPHNVVTLMIEYDGFFFHSDETSATSTDHTYIVSMTAHLISML
                ***:**  **: *.:*******::*:****.   .    .  . * :*::.: :* *:*

npr-1           NNVLNPVLYAWLNPSFRQLVIKTYFGDRRKSDRI-INQTSVYKTKIVHDTKHLNGRAKIG
npr-2           TNVTNPFLYAWLNPMFKEMLIKTLRGGSKSPKPADIKQTSFIR----------------
                .** **.******* *::::***  *. :...   *:***. :

npr-1           GGGSHEALKERELNSCSENLSYHVNGHTRTPTPEVQLNEVSSPEISKLVAEPEELIEFSV
npr-2           -------------------------------------MPNSGAPSQS-------------
                                                     : : .:*. *

npr-1           NDTLV
npr-2           --SYL
                  : :
```

Fig. 3.6 The pairwise alignment results between (A) *npr-1* and *npr-3*, and (B) *npr-1* and *npr-2*

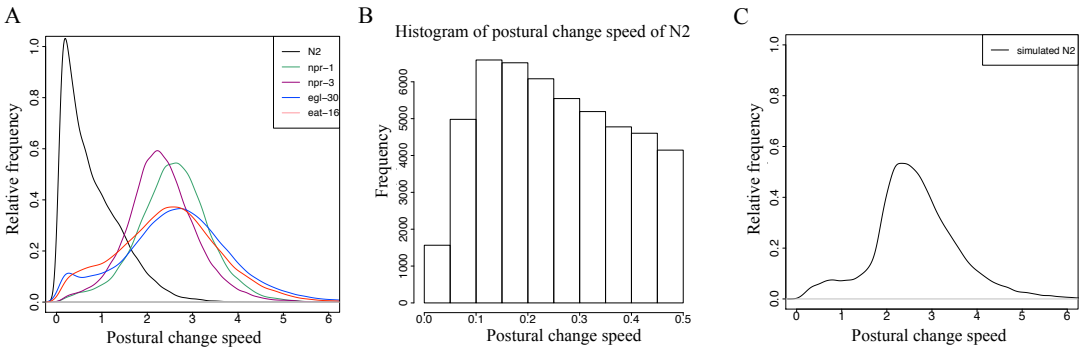


Fig. 3.7 The distribution of postural change speed (A) for N2, *npr-1*, *npr-3*, *egl-30*, and *eat-16*. (B) for simulated N2. The x-axes and the y-axes represent postural change speed and relative frequency, respectively.

3.3.3 Postural change speed analysis of N2 and mutant strains

In this section, I analyzed the reason why *npr-1*, *npr-3*, *egl-30*, and *eat-16* mutants showed large JSDDs. I classified the possible reasons into three behavioral patterns: the frequency change of the quiescence state, the change of behavioral speed, and taking the novel postural change patterns. I explain these behavioral phenotypes in detail below. As an example, we think that N2 takes only five postures "A", "B", "C" , "D", and "E", and a postural change pattern "ABCDEABC...". Here, the occurrence probability of each posture is all 0.2. In addition, the posture transition probabilities of A $\rightarrow$ B, B $\rightarrow$ C, C $\rightarrow$ D, D $\rightarrow$ E, E $\rightarrow$ A are all 0.2, and those of the other transition patterns are all 0.0. In first case, if the frequencies of quiescence behavior increase, the postural change pattern is changed as follows. "ABBCCDEEAABCDDE...". In this case, the posture occurrence probabilities are same as N2 strain but change the posture transition probabilities. In second case, if the behavioral speed is twice faster than N2, the transition pattern is changed as follows. "A(B)C(D)E(A)B(C)D(E)...". In this case, the postural change pattern is not changed actually, but the postures are observed alternately and the posture change patterns is recognized as "ACEBDACE..". Third case is a most simple case that worm takes a novel postural change patterns. For example, if worm shows postural change pattern "ADBCEADBCE...", the posture transition probabilities are completely different from N2. I investigated which behavioral phenotypes could explain large JSDD of *npr-1*, *npr-3*, *egl-30*, and *eat-16*.

Firstly, in order to investigate whether the cause of JSDD were "the frequency change of quiescence behavior" and "the change of behavioral speed" for these mutant strains, I calculated postural change speed for N2 and mutant strains. The postural change speed is defined as the Euclidean distance of four eigenvalues between continuous time points.

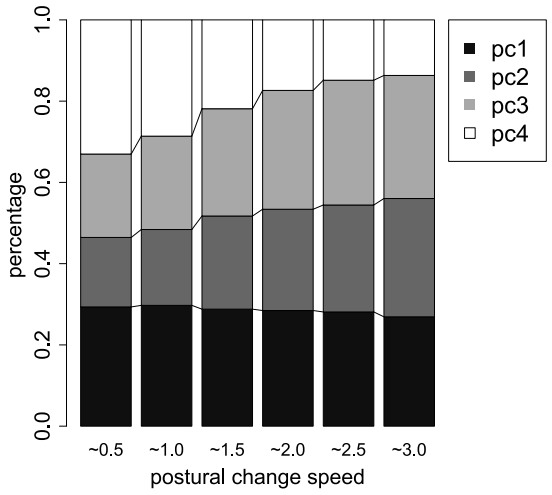


Fig. 3.8 The ratio of change of each principal component to total postural change speed . The x-axis represents the postural change speed.

Table 3.4 JSD_{oc_sim}, JSD_{tr_sim}, and $JSDD_{sim}$ between simulated N2 and mutant strains.

strain	JSD_{oc_sim}	JSD_{tr_sim}	$JSDD_{sim}$
npr-1	0.0135	0.0314	0.0179
npr-3	0.0158	0.0398	0.0240
egl-30	0.0235	0.0494	0.0259
eat-16	0.0150	0.0339	0.0189

Fig. 3.7A shows the distribution of postural change speed for N2, *npr-1*, *npr-3*, *egl-30*, and *eat-16*. The distribution for N2 was unimodal, and the quiescence state and behavior state could not be explicitly classified. In addition, the mode of distribution was slightly greater than 0.0, and the duration of complete quiescence was short (Fig. 3.7B). To reveal the cause of this subtle postural change, I investigated which principal components mainly changed when worm posture changes. As a result, the ratio of PC4, which means the movement of worm head and tail, was large when worms change their postures slowly (Fig. 3.8) [71]. For the other mutant strains, each strain accelerated their postural change speed. In addition, *npr-1* and *npr-3* mutants did not show low speed state, and *egl-30* and *eat-16* took low speed state but the ratios were considerably smaller than that for N2. Therefore, each strain shows both "the frequency change of quiescence behavior" and "the change of behavioral speed" patterns, and these behaviors should influence large JSDD.

Finally, I analyzed whether "taking the novel postural change patterns" was occurred at these mutant strains. In order to exclude the effect of "the frequency change of

quiescence behavior" and "the change of behavioral speed", I generated simulated N2 dataset, and compared the simulated N2 dataset with the mutant strain. Namely, the postural changes speed and the frequency of low speed state of simulated N2 dataset is almost same as mutant strain, but the postural change pattern of simulated N2 does not change from original N2. The simulated N2 dataset was generated as follows. Firstly, I excluded the frame that the postural change speed between previous frame and the frame is smaller than 0.9. Next, to double the postural change speed, I alternately discarded the frame of remained N2 dataset. Fig. 3.7B shows the distribution of postural change speed for simulated N2 dataset. The distribution was similar to those of the mutant strains. Then, I calculated the posture occurrence probabilities and posture transition probabilities for simulated N2 dataset. I calculated JSD of posture occurrence probabilities and posture transition probabilities between simulated N2 and the other strain, respectively. I termed these scores as $\mathrm{JSD}_{\mathrm{oc\_sim}}$, $\mathrm{JSD}_{\mathrm{tr\_sim}}$. In addition, the difference between $\mathrm{JSD}_{\mathrm{oc\_sim}}$ and $\mathrm{JSD}_{\mathrm{tr\_sim}}$ is termed as $\mathrm{JSDD}_{\mathrm{sim}}$. Table 3.4 shows these scores for each mutant strain. Surprisingly, for each mutant strain, $JSDD\_{\mathrm{sim}}$ was not a very large value. This result suggested that the contribution of "taking the novel postural change patterns" to large JSDD was limited.

3.4 Discussion

In this study, I firstly obtained template posture set by Gaussian mixture model, and transformed worm postural change patterns into probabilistic sequences of template postures. Next, by comparing with posture occurrence probabilities of N2 and those of the other strains, I investigated whether the reason why mutants show abnormal postural change patterns is "the usage of different postural set" or not. Then, I revealed several strains (*npr-1*, *npr-3*, *egl-30*, *eat-16*) that shows the similar posture occurrence probabilities to N2 as but different posture transition probabilities from N2. Finally, by comparing postural change speeds of these mutants with that of N2, I revealed that these strains show both "the frequency change of quiescence behavior" and "the change of behavioral speed", but do not very take "the novel postural change patterns".

I revealed intra-strain consistency performance of K-means algorithm is lower than that of GMM algorithm. The reason may be that GMM probabilistically assigns each postural data to each cluster while the assignment of K-means algorithm is deterministic. As postural dataset consists of postural change trajectory of each worm and these trajectories are continuous in four-dimensional postural space, postural dataset do not perfectly divided in multiple clusters and there should be many postural data in the middle point of several clusters. In order to appropriately handle these data whose be-

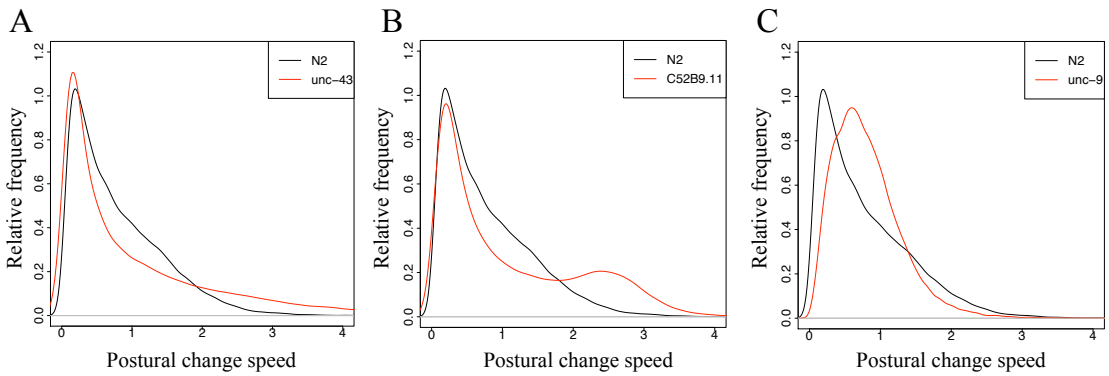


Fig. 3.9 The distribution of postural change speed for N2 and (A) *unc-43*, (B) *C52B9.11*, and (C) *unc-43*.The x-axes and the y-axes represent postural change speed and relative frequency, respectively.

longing is ambiguous, probabilistic assignment should be more suitable than deterministic assignment.

Based on movement trajectory analysis, Gallagher *et al.* discovered that worms take three basic behavior corresponding to roaming, dwelling, and quiescence but there are many intermediate states in worm behavior [87]. In this research, I revealed that the distribution of postural change speed is unimodal, and quiescence state and behavior state cannot be explicitly divided in N2 strain. This result partially supports Gallagher's discussion. On the other hand, I discovered that worm takes subtle movement behavior than complete quiescence behavior, and N2 moves their heads and tails in this state. Head and tail movements do not largely change the location of worm centroid, and thus this movements should be regarded as quiescence behavior in Gallagher's research. However, this movement is important behavior when worm conducts navigation behavior [88], and thus this subtle movement should be discriminated from complete quiescence behavior [89, 90].

All *npr-1*, *npr-3*, *egl-30*, and *eat-16* mutants have caused two strange behavioral patterns that are "the frequency change of quiescence behavior" and "the change of behavioral speed". To investigate whether these two behavioral changes can be caused independently, I checked the distribution of postural change speed for the other mutants, and discovered some characteristic mutants. For example, *unc-43* and *C52B9.11* showed low-speed state like N2 but the behavioral speed was faster than N2 (Fig. 3.9A and B). In contrast, for *unc-9*, the behavioral speed is similar to N2 strain but decreases the duration of low-speed state (Fig. 3.9C). These examples suggest that "the frequency change of quiescence behavior" and "the change of behavioral speed" are independent phenomena.

Recent advances in sequencing technologies have discovered many new genes from genomic data, but these gene functions are unknown in most cases. Therefore, the function prediction of these genes is one of the most important research topic in bioinformatics. Large scale phenotypic analysis was the one of the solution, and many functions of unknown function genes were revealed by this method [91, 92]. As example of worm genes, Yu *et al.* revealed novel gene components in G-protein Gαq signaling pathway by analyzing behavior of 4,400 animals of 239 strains [93]. In this research, I revealed *npr-1* gene and *npr-3* gene have a similar function with each other and a different function with the other *npr* genes at behavioral level. Interestingly, while there has been many researches about behavior of *npr-1* mutants and several *npr-1*-related neural mechanisms have been revealed, few researches focused *npr-3* mutants and thus *npr-3*-related neural mechanisms were almost unknown [78, 79]. My analysis may suggest that the some of the known *npr-1*-related neural mechanisms are also related to *npr-3* gene. This case also demonstrates that large-scale phenotypic analysis may be useful to infer experimentally unverified neural mechanisms. The functions of many *C. elegans* genes were still unknown, and *C. elegans* behavioral database includes many behavioral data of these gene mutants [70]. To reveal functions of genes with unknown function, the development of analytic method of phenotypic information is an essential task.

3.5 Appendix

In this section, I prove that JSD of posture occurrence probabilities between two strains is equal to smaller than JSD of posture transition probabilities between same two strains. I may prove that Kullback-Leibler divergence of posture occurrence probabilities between two strains is equal to smaller than that of posture transition probabilities between same two strains.

I defined posture occurrence probabilities of strain a and b, and posture transition probabilities of strain a and b as p_{oc}, q_{oc}, p_{tr}, and q_{tr}, respectively. $p_{oc}(i)$ and $q_{oc}(i)$ represents probability that strain a and b takes a posture i, respectively. In addition, $p_{tr}(i,j)$ and $q_{tr}(i,j)$ represents probability that strain a and b change postures from posture i and posture j, respectively. Here, $\sum_i p_{oc}(i) = 1$, $\sum_i q_{oc}(i) = 1$, $\sum_j p_{tr}(i,j) = p(i)$, $\sum_j q_{tr}(i,j) = q(i)$ is satisfied.

The proof is as follows:

$$\sum_i \sum_j p_{tr}(i,j) \log \frac{q_{tr}(i,j)}{p_{tr}(i,j)} - \sum_i p_{oc}(i) \log \frac{q_{oc}(i)}{p_{oc}(i)}$$

$$
\begin{aligned}
&= \sum_i \sum_j p_{tr}(i,j) \log \frac{q_{tr}(i,j)}{p_{tr}(i,j)} \frac{q_{oc}(i)}{q_{oc}(i)} \\
&\leq \log \sum_i \sum_j q_{tr}(i,j) \frac{q_{oc}(i)}{q_{oc}(i)} \\
&= \log \sum_i p(i) \\
&= 0
\end{aligned}
$$

Chapter 4

CapR: revealing structural specificities of RNA-binding protein target recognition using CLIP-seq data

4.1 Introduction

In this section, I discuss analysis of genetic basis for animal behavior based on large-scale sequence data. Specifically, I focus on RNA-binding proteins (RBPs) target recognition because RBPs deeply relate to neurodegenerative disorders causing abnormal behavior[94, 95]. RBPs play integral roles in various post-transcriptional regulatory processes, including the splicing, processing, localization, degradation and translation of RNA molecules [96]. RBPs typically contain a limited set of RNA-binding domains, such as the RNA recognition motif and K homology domain, and they must bind to specific RNA molecules to function. RBP–RNA interactions and their specificities are important for understanding the complex gene regulatory networks and the mechanisms of diseases.

Recent advances in 'ribonomic' technologies, such as cross-linking immunoprecipitation high-throughput sequencing (CLIP-seq, also referred to as HITS-CLIP) [97], individual-nucleotide resolution CLIP (iCLIP) [98], and photoactivatable-ribonucleoside-enhanced CLIP (PAR-CLIP) [99], have enabled the study of RBP–RNA interactions, both on a genomic scale and at high resolution.The use of microarrays in the classical RNA-binding protein immunoprecipitation microarray (RIP-chip) method [100] prevented the precise identification of binding sites.In contrast, CLIP-seq methods bond an RBP and RNAs covalently by ultraviolet cross-linking, collect them by immunoprecipitation and directly sequence the RBP-bound sites of the RNAs. Using these technologies, researchers can identify sequential RNA motifs that are over-represented around the binding sites of each RBP using bioinformatics methods similar to those used for analyzing transcription-factor binding DNA motifs [101]. Such sequential motifs are often very short (up to ten bases), and there are many unbound sites that have the same motif. Thus, sequential motifs alone cannot explain the specificity of RBP–RNA interactions.

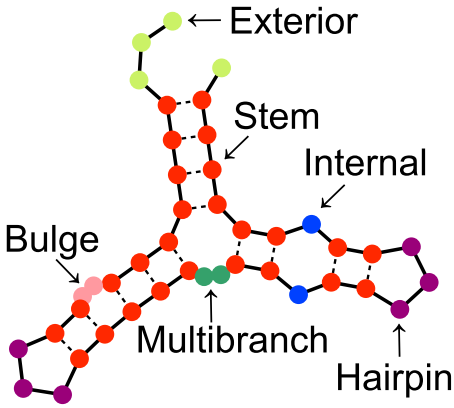


Fig. 4.1 Visual representation of the six structural contexts. The six structural contexts are represented by six colors: stems (red), exterior loops (light green), hairpin loops (purple), bulge loops (pink), internal loops (blue) and multibranch loops (green). The unstructured context is the union of the exterior and multibranch loops. These colors are used throughout the paper.

RBPs bind to their target RNA molecules by recognizing specific RNA sequences and their structures. Several studies have addressed this issue by calculating the accessibility of RNA regions around the RBP-binding sites [102]. Here, the accessibility of an RNA region is defined by the probability that the region exhibits a single-stranded conformation. Theoretically, the accessibility can be efficiently and exactly calculated using an energy model of RNA secondary structures [103, 104]. Double-helical RNAs usually form the A-form helical structure, whose major grooves are too narrow to be accessed by RBPs [105], and Li *et al.* showed that the accessibilities tend to be high around the RBP-bound motif sites by analyzing RIP-chip data [102]. However, it is not sufficient to consider accessibility alone in analyzing the structure-specific target recognition by RBPs. For example, Vts1p, which is a yeast RBP regulating mRNA stability, binds to its target CNGG sequential motif when it is located within hairpin loops but not when it is located in single-stranded regions or other structures [106, 107]. The human FET family of proteins, whose mutations are associated with amyotrophic lateral sclerosis, bind to its target sequential $\mathrm{UAN}_n\mathrm{Y}$ motif within hairpin loops [108]. Computational methods for calculating the secondary structural contexts of RNA molecules, such as bulge loops, hairpin loops and stems, are required to uncover the characteristics of the RNA structures that are recognized by the RBPs *in vivo*.

In the present study, I developed an efficient algorithm that calculates the probabilities that each RNA base position is located within each secondary structural context. Six

contexts of RNA secondary structures were taken into account, according to the well-established Turner energy model of RNAs [109]. These structures included stems (S), hairpin loops (H), bulge loops (B), internal loops (I), multibranch loops (M) and exterior loops (E) (see Fig. 4.1). I defined a *structural profile* of an RNA base as a set of six probabilities that the base belongs to each context. At present, Sfold [110] is the only software that can calculate a structural profile. Sfold cannot be readily applied to tens of thousands RNA fragments because it uses a statistical sampling method that requires huge sample sizes and computational costs, particularly when analyzing long RNAs or mRNAs. I implemented my efficient algorithm as software named 'CapR' , which can compute the structural profiles for tens of thousands of long RNAs within a reasonable time by enumerating all the possible secondary structures of the RNAs.

4.2 Results

4.2.1 Methods overview

I have developed a new algorithm that calculates the structural profiles of any RNA sequence based on the Turner energy model with time complexity $O(NW^2)$ [109]. Here, N is the input sequence length and W is the *maximal span*, which is a given parameter of the maximal length between the bases that form base pairs. The parameter W was introduced because considering very long interactions does not improve the accuracy of the secondary structure predictions but does increase the computational costs [111].

Let x be an RNA sequence of length N and σ be a possible secondary structure on x without pseudoknots. I refer to a base in x as *stem* if it forms a base pair with another base, and represent it using the character S. Single-stranded bases are categorized into five structural contexts, namely, *bulge loop* (represented by B), *exterior loop* (E), *hairpin loop* (H), *internal loop* (I) and *multibranch loop* (M), which are defined as follows. In a secondary structure representation, RNA bases are vertices of polygons whose edges are the RNA backbone or hydrogen bonds, which are shown as solid or dotted lines, respectively, in Fig. 4.1. The exterior loop context is given to single-stranded bases if they do not form polygons. The hairpin loop context is given to single-stranded bases if they form a polygon that has a single hydrogen bond. The bulge and internal loop contexts are given to single-stranded bases if they form a polygon that has two hydrogen bonds, which are connected by a single backbone edge for bulge loops and which are not connected by a single backbone edge for internal loops. Finally, the multibranch loop context is given to single-stranded bases if they form a polygon that has more than two hydrogen bonds. Note that for a given secondary structure σ, any base of x is unambiguously classified as one of the six structural contexts. Additionally, I define

unstructured (U) to represent collectively the exterior and multibranch loop contexts.

I assume that the probability distribution of the secondary structures follows the Boltzmann distribution with respect to the Turner energy model [109]. The probability $p(i, \delta)$ that a base at position i has the structural context $\delta \in \{B, E, H, I, M, S\}$ is given by

$$p(i, \delta) = \frac{1}{Z(x)} \sum_{\sigma \in \Omega(i,\delta)} \exp\left(-\Delta G(\sigma, x)/RT\right)$$
$$Z(x) = \sum_{\sigma \in \Omega_0} \exp\left(-\Delta G(\sigma, x)/RT\right)$$

where $\Delta G(\sigma, x)$ is the difference of the Gibbs energies of the given structure σ and the structure σ_0 that contains no base pairs, R is the gas constant and T is the temperature (I used $T = 310.15$ K in this study). Ω_0 is the set of all the possible secondary structures of x, and $\Omega(i, \delta)$ is the set of all the possible secondary structures in which the base at position i is in the structural context δ. Then, the structural profile of i is defined as the probabilities of the structural contexts $\{p(i, \delta) | \delta \in \{B, E, H, I, M, S\}\}$. Note that the structural profile satisfies the probability condition $\sum_\delta p(i, \delta) = 1$.

My algorithm efficiently calculates structural profiles by referring to the Rfold model, which is a variant of the stochastic context-free grammar (SCFG) that calculates all the RNA secondary structures without redundancy [112]. In formal language theory, the RNA secondary structures without pseudoknots are modeled by SCFG [113]. While the state transition rules of the Rfold model contain seven non-terminal symbols, my algorithm associated them with the six structural contexts. The details of the algorithm, which is a variant of the inside-outside algorithm of SCFG, are given in the Materials and methods section.

4.2.2 Influence of the maximal span and the GC content on the structural profile calculations

Before I investigated the structure-specific target recognition by RBPs, I evaluated the performance of CapR. Because I introduced the maximal span W, I needed to investigate an appropriate range for this parameter. Because GC content is known to affect the RNA secondary structures, its effect was also analyzed.

To investigate the dependence on the maximal span W, I applied CapR to 1,000 random RNA sequences of 2,000 nucleotides with a fixed GC content (GC = 0.5). Fig. 4.2A shows how the proportions of the calculated structural profiles depend on W. As expected, if W is small, the predictions are dominated by exterior loops because few bases form base pairs under this condition. Whereas the probabilities for bulge loops, hairpin loops, internal loops and stems are relatively stable for $W \geq 100$, the exterior loop probabilities monotonically decrease and the multibranch loop probabilities monotoni-

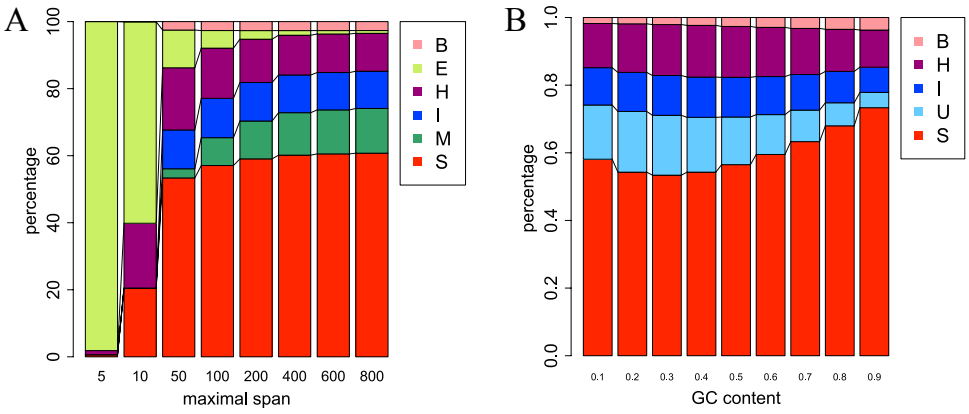


Fig. 4.2 (A) Dependence of the structural profiles on the maximal span W. The x-axis represents the maximal span W. The y-axis represents the averaged $p(i, \delta)$ over all the nucleotides. (B) Dependence of the structural profiles on the GC content. The x-axis represents the GC content. The y-axis represents the averaged $p_\delta(i)$ over all the nucleotides. The unstructured context is represented by light blue. B, bulge loop; E, exterior loop; H, hairpin loop; I, internal loop; M, multibranch loop; S, stem; U, unstructured.

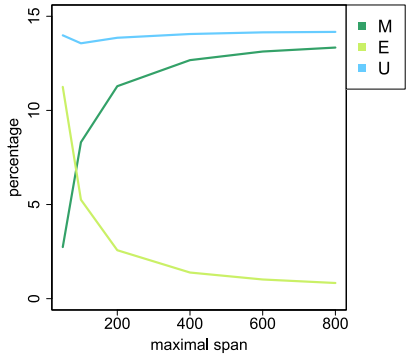


Fig. 4.3 Dependence of the exterior loop, multibranch loop, and unstructured context on the maximal span W. The x-axis represents the maximal span W. The y-axis represents the averaged $p(i, \delta)$ over all the nucleotides.

cally increase with increasing W. This is because at large W new base pairs form in exterior loops and exterior loops turn into multibranch loops. On the other hand, the probabilities of the unstructured context, which collectively represents the exterior and multibranch loop contexts, are insensitive to W (Figure 4.3). Therefore, the unstructured context can be adopted instead of the external and multibranch loop contexts to avoid the influence of the parameter W, if a discrimination of the two contexts is not critical.

Table 4.1 The AUC score of each structural context.

Software	Bulge	Exterior	Hairpin	Internal	Multibranch	Stem
CapR	0.847	0.866	0.890	0.765	0.852	0.805
Sfold	0.842	0.817	0.890	0.769	0.853	0.804

Although Kiryu *et al.* revealed the dependence of the accessibilities on the GC content [104], the dependence of structural profiles on the GC content has not been investigated. I investigated the dependence on the GC content by applying CapR to 1,000 random RNA sequences of 2,000 nucleotides with a fixed maximal span ($W = 100$). Fig. 4.2B shows how the proportions of the computed structural profiles depend on the GC content. The stem probability is high and the unstructured probability is low with a high GC content, probably because the energy of the G-C pairs is larger than that of the A-U pairs and palindromic sequences are more likely to occur in the high-GC background. This result suggests that users should carefully interpret the results when analyzing RNAs with biased GC content.

4.2.3 Performance of CapR

I evaluated the speed of CapR by comparing its computational run-time with that of Sfold. The input sequences were generated randomly with equal probabilities of A, C, G and U. For Sfold, the number of sampled structures was set to its default value (1,000). The computation was performed on an AMD Opteron 6276 2.3 GHz with 1 GB memory. Fig. 4.4A shows the computational run-times, which depended on the maximal span W and sequence lengths. In all cases, CapR was much faster than Sfold. Sfold could not run for $N \geq 4000$ while CapR did for $N = 10000$. These results show that CapR can compute structural profiles for long RNAs within a reasonable time.

Next, I evaluated the accuracy of the structural profiles computed by CapR using 8,775 RNA genes that have experimentally validated secondary structure annotations in the Rfam database [114]. I set $W = 800$ to allow for stem-forming of the base pairs with the longest distance observed in the Rfam dataset. To estimate the accuracy of the structural profiles, I calculated the area under the receiver operating characteristic curve (AUROC) for each structural context. Briefly, the AUROC is high if the probability $p(i, \delta)$ for the structural context δ annotated in Rfam is high.

Table 4.1 and Fig. 4.4B show the AUROC values and the receiver operating characteristic curves, respectively. The AUROC value for each structural context was larger than 0.75, indicating that the computed structural profiles were very consistent with the Rfam annotation. For example, the structural profile of transfer RNAs (tRNAs), whose secondary structures are well characterized, is shown in Fig. 4.4C. Each line represents

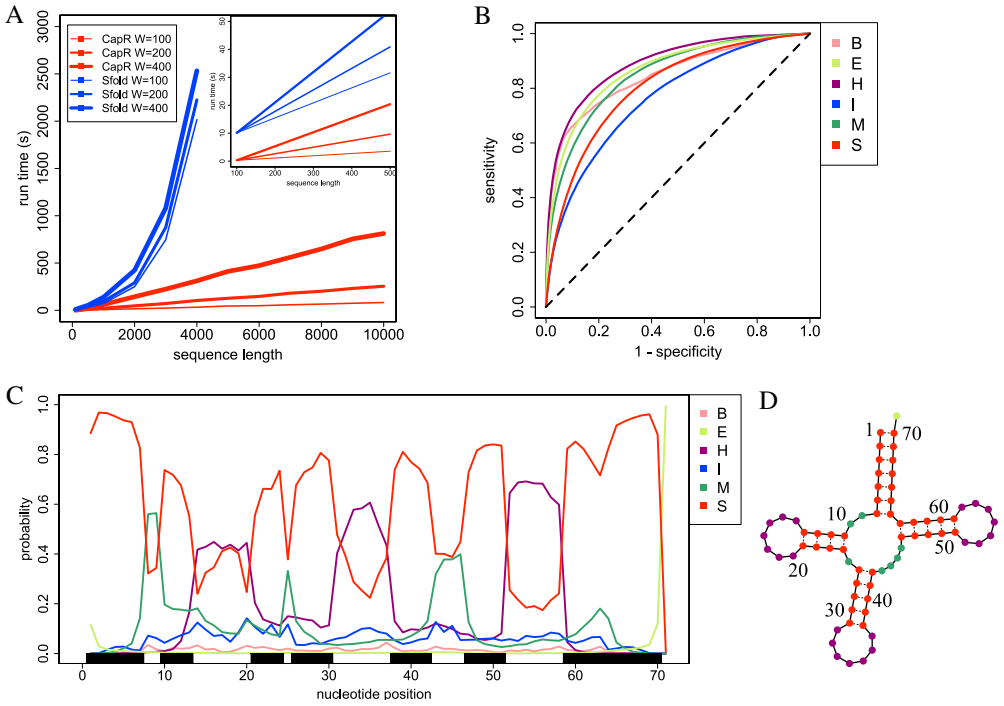


Fig. 4.4 (A) Computational run-times for different values of maximal span W and sequence length N. The x-axis represents the sequence length N. The y-axis represents the computational run-time. (B) The receiver operating characteristic curve for each loop context. The x-axis represents 1-specificity and the y-axis represents the sensitivity. The specificity and sensitivity are defined as $\mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$ and $\mathrm{TN}/(\mathrm{TN}+\mathrm{FP})$, respectively. (C) The structural profiles of tRNAs. The x-axis represents the nucleotide positions from $5'$ to $3'$. The y-axis represents averaged probabilities that each base belongs to each structural context across all tRNA genes in the Rfam dataset [114]. The black boxes represent the nucleotides annotated as stem in Rfam. (D) tRNA cloverleaf structure annotated in Rfam. B, bulge loop; E, exterior loop; H, hairpin loop; I, internal loop; M, multibranch loop; S, stem.

averaged probabilities that each base belongs to each structural context across all tRNA genes in the Rfam dataset. Probabilities of the stem, hairpin loop, multibranch loop and exterior loop contexts were high at the corresponding parts of the tRNA cloverleaf structure (Fig. 4.4D). Calculated structural profiles are interpreted by considering that stem probabilities tend to be overestimated by the Turner energy model. In the tRNA example, the calculated stem probabilities were slightly higher than the multibranch loop probabilities at positions 25, 43 and 44, which are annotated as multibranch loops in Rfam.

Finally, the same analysis was conducted using Sfold, and the accuracies of the structural profiles predicted by CapR and Sfold were compared. The accuracies of CapR were

comparable to those of Sfold (Table 4.1).

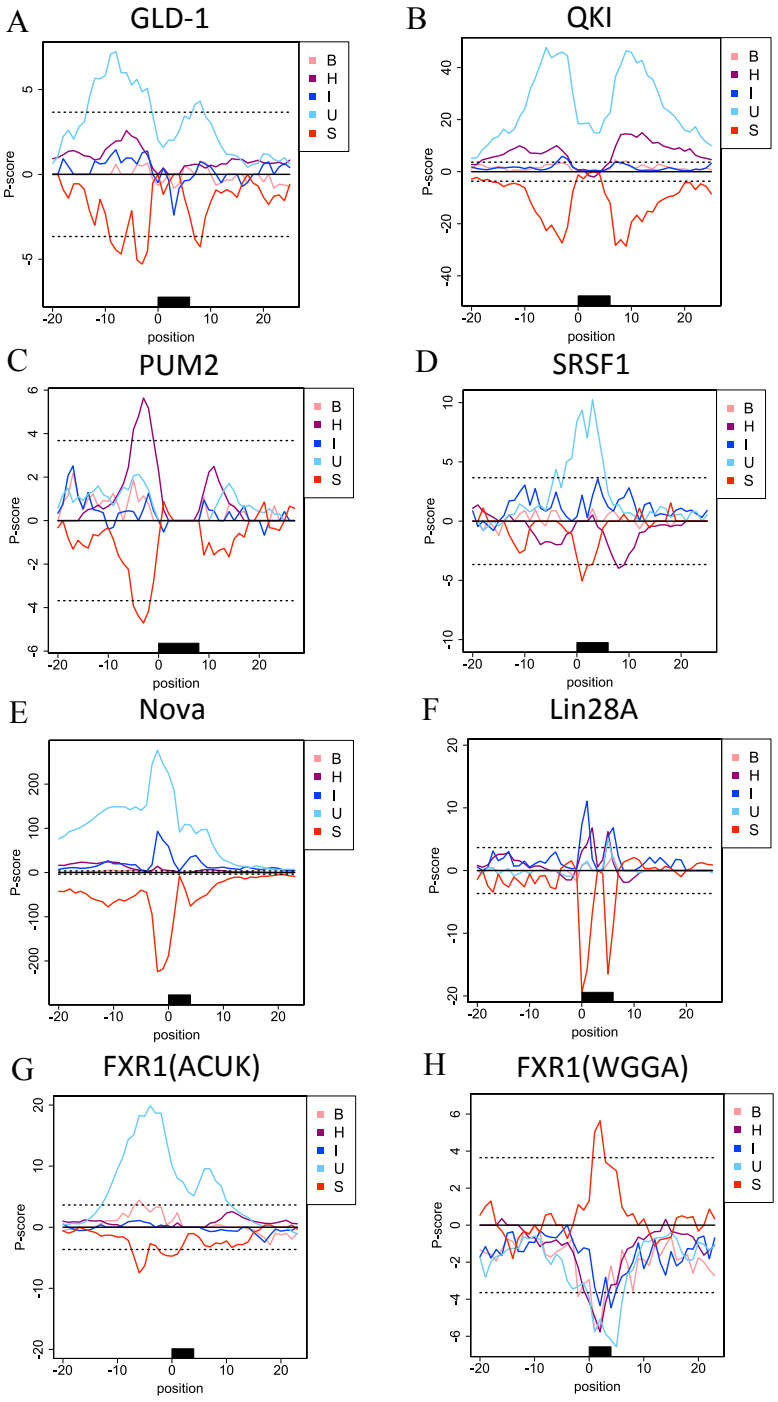
4.2.4 Datasets and methods used in the CLIP-seq data analysis

Because it was shown that CapR is accurate in calculating structural profiles of RNA molecules, I applied it to several CLIP-seq datasets to reveal the structural specificities of RBP–RNA interactions. For the subsequent analyses, I downloaded CLIP-seq data of RBP-bound RNAs from the doRina database [115], and selected ten RBPs: GLD-1 (nematode), QKI (human), Pum2 (human), SRSF1 (human), Nova (mouse), Lin28A (mouse), FXR1 (human), FXR2 (human), FMR1_7 (human) and FMR1_1 (human) [116, 99, 117, 118, 119, 120] (refer to Materials and methods for the criteria for the data selection). FMR1_7 and FMR1_1 are two splicing isoforms of FMR1. RBPs with two known sequential motifs (FXR1, FXR2, FMR1_7 and FMR1_1) were analyzed separately for each of the motifs. Hereafter, these cases are represented by the protein names with their sequential motifs: FXR1(ACUK), FXR1(WGGA), FXR2(ACUK), FXR2(WGGA), FMR1_7(ACUK), FMR1_7(WGGA), FMR1_1(ACUK) and FMR1_1(WGGA).

I created one positive dataset and two negative datasets for each of these 14 cases. The positive dataset was a collection of transcribed sequences of ±2000 nucleotides around each RBP-bound site. The RBP-bound sites were defined as sites of sequential motifs within the CLIP-seq peak regions. The two negative datasets are referred to as the unbound and shuffled datasets. The unbound dataset was a collection of transcribed sequences of ±2000 nucleotides around a sequential motif site that was in the same transcriptional unit and within ±1000 nucleotides of any RBP-bound site, but was not an RBP-bound site. In short, this dataset represents the sequential motif sites that are transcribed but unbound by the RBP. The shuffled dataset was generated by randomly shuffling each of the upstream and downstream sequences of each RBP-bound site by preserving nucleotide di-nucleotide frequencies for every sequence in the positive dataset. Thus it represents the sequential motif sites flanked by sequences with preserved sequence compositions. The details of the datasets are described in the Materials and methods section.

I calculated the structural profiles of the positive, unbound and shuffled datasets for each of the RBPs ($W = 200$). Then, to evaluate the structural contexts that are significant in the positive dataset statistically, I defined a P score as follows. First, I calculated a P value using the one-sided Wilcoxon–Mann–Whitney test for each side for each position. Second, I selected the smaller P value of the two hypotheses and transformed it into $-\log_{10} P$, which I designated the P score. Third, if a P score was calculated under the hypothesis that each context probability of the positive dataset was smaller than that of the negative dataset, I changed the sign of the P score. For

example, a large positive P score indicates that the probability of that structural context is significantly larger in the positive dataset. Finally, the two P scores calculated for the two negative datasets were compared for each position, and the smaller P score was taken (if one P score was positive and the other was negative, I used 0 instead of the two P scores). Note that the Bonferroni correction was used for multiple testing. To avoid the effects of the artificial value selection for the parameter W, I used the unstructured context instead of the exterior and multibranch loop contexts in the following analysis.



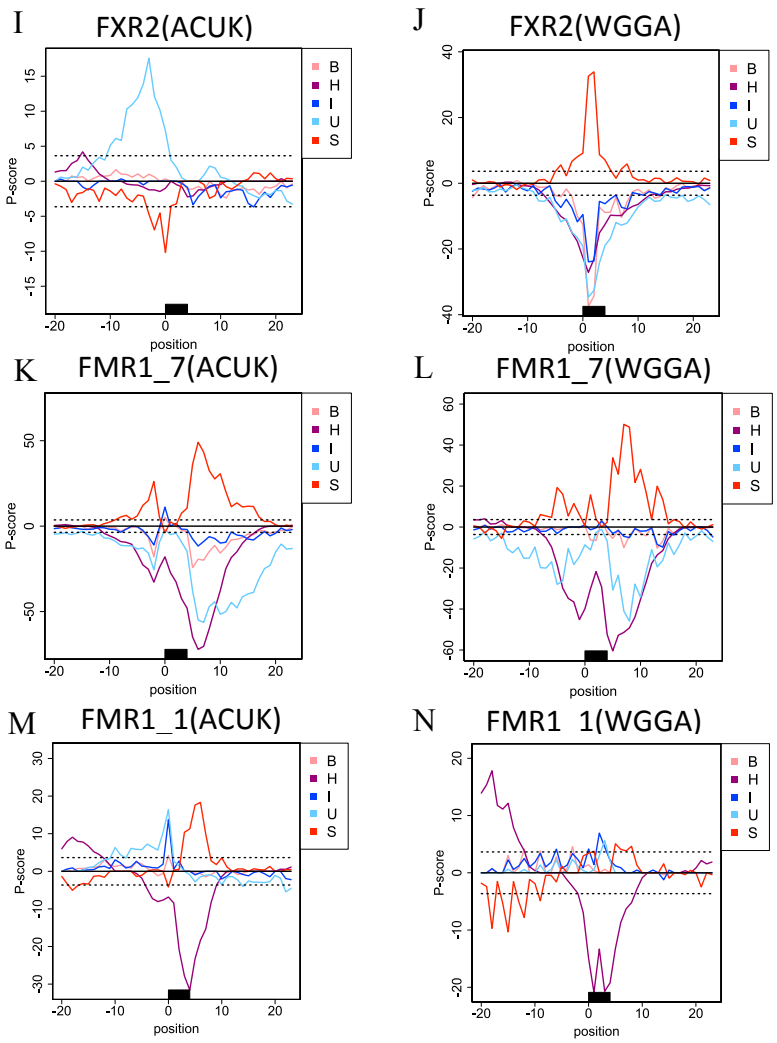


Fig. 4.5 The distribution of the P-scores for each RBP. The x-axis represents nucleotide positions, and the y-axis represents P-score of ± 20 bases around the sequential motif site. The black box represents the sequential motif site. The dotted lines are the corrected significance level of Bonferroni correction ($\alpha = 0.05$).

4.2.5 Specific RNA structural contexts recognized by RNA-binding proteins

I investigated the preferred RNA structural contexts for each RBP and revealed that most RBPs prefer a specific structural context (Figure 4.5). My method was robust regarding the selection of the negative datasets, because selecting the larger P scores did not affect the results overall. Among the 14 cases analyzed, six cases showed a preference for the unstructured context (GLD-1, QKI, SRSF1, Nova, FXR1(ACUK) and FXR2(ACUK)). Except for Nova, the RBP-bound sites tended to form the unstructured context, but did not show preferences for the bulge, internal or hairpin loop contexts (Fig. 4.5A, B, D, E, G, I). It should be noted that these results could not be obtained by analyzing the accessibility alone, which does not discriminate between these non-stem contexts.

Pum2 showed a preference for the hairpin loop context (Fig. 4.5C). To my knowledge, this is the first report of the structural preference for the hairpin loop context by Pum2, which is known to be involved in germ cell development [121]. Lin28A showed preferences for the hairpin and internal loop contexts (Fig. 4.5F). Lin28A is known to inhibit the maturation of let-7 miRNAs and the translation of mRNAs that are destined for the endoplasmic reticulum [119]. The specificity of Lin28A to the hairpin loop context is consistent with the previous study [119]. In addition, my result is the first to suggest that Lin28A prefers the internal loop context in mRNA binding, and Lin28A has been reported to bind to the internal loop of let-7 miRNAs [119].

FXR1(WGGA), FXR2(WGGA) and FMR1_7(WGGA) showed preferences for the stem context (Fig. 4.5H, J, L), although RBPs were considered to be unlikely to be bound to the stem regions of RNAs as already mentioned. These three RBPs (and FMR1_1) are members of the FMRP family and are known to be responsible for the fragile X syndrome. Darnell *et al.* showed that FMRP-bound WGGA sites tend to form a G-quadruplex, which is composed of guanine-rich sequences forming a four-stranded RNA structure [122]. I suppose that the preference for the stem contexts could reflect the tendency that these family members recognize the G-quadruplex; however, this should be investigated further as currently my energy model and grammar cannot deal with G-quadruplexes.

FMR1_7(ACUK) showed preferences for the internal and bulge loop contexts (Fig. 4.5K). To my knowledge, this is the first report of the structural specificities of FMR1. In contrast, FXR2(ACUK), where FXR2 is a homolog of FMR1, preferred neither the internal nor bulge loop context (Fig. 4.5I). FMR1_7 has an exon insertion in its K homology domain that recognizes the ACUK sequential motifs [120]. This insertion appears to underlie the differences in the structural specificity between FMR1_7(ACUK) and FXR2(ACUK).

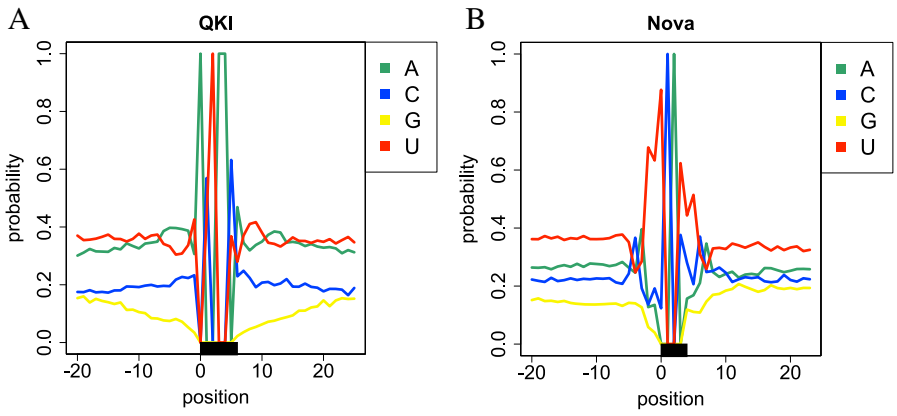


Fig. 4.6 The nucleotide compositions of ±20 bases around the RBP-bound sites for (A) QKI and (B) Nova. The x-axis represents the nucleotide position and the y-axis is the proportion of each nucleotide. The black box represents the sequential motif site.

4.2.6 Positional preferences in the RNA structure recognition by RNA-binding proteins

The present understanding of the structural specificities of RBP–RNA interactions overlooks structures of the flanking sequences of RBP-bound sites. Therefore, I investigated the secondary structures not only of the RBP-bound sites but also of their flanking sequences. In fact, the positions with the highest P scores were not within the RBP-bound sites in some RBPs. QKI (Fig. 4.5B), SRSF1 (Fig. 4.5D) and Nova (Fig. 4.5E) preferred the unstructured context. High P scores were observed within the RBP-bound sites for SF2ASF, whereas they were observed in the flanking and upstream sequences for QKI and Nova, respectively. These results suggest that RBPs also recognize specific structures existing outside of sequential motif sites, and CapR can uncover these positional preferences from ribonomic datasets.

Fig. 4.6A,B shows the nucleotide compositions around the RBP-bound sites of QKI and Nova. The flanking sequences of QKI-bound sites were guanine poor, whereas those of Nova-bound sites were uracil rich. Because sequences with low GC content tend to form an unstructured context, the aforementioned positional preferences could be generated by the biased nucleotide compositions. To address this possibility, I investigated the relations between the nucleotide compositions and structural specificities in the flanking sequences. I generated partially shuffled datasets by randomly shuffling sequences outside of the ±5 or 10 nucleotides of the RBP-bound sites with preserving di-nucleotide frequencies, and compared their structural profiles with those of the positive datasets using the Wilcoxon–Mann–Whitney test. Then, the P scores for the shuffled and partially

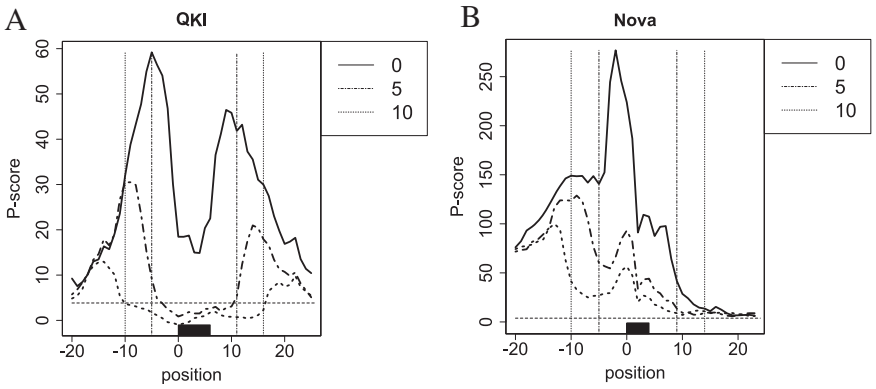


Fig. 4.7 The shuffled, the partially shuffled (±5) and the partially shuffled (±10) datasets are represented by 0, 5 and 10, respectively. The x-axis represents the nucleotide position and the y-axis is the P score of (A) QKI and (B) Nova. The black boxes are the RBP-bound sites, and the horizontal dotted lines the corrected significance levels of the Bonferroni correction. The vertical dotted lines indicate the ±5 or 10 nucleotides of RBP-bound sites.

shuffled datasets were compared (Fig. 4.7A,B). For QKI, whereas the shuffled dataset had positional preferences in the flanking sequences, the partially shuffled datasets had no significant preferences. This means that the structural specificities of QKI could be generated by the biased nucleotide compositions in the flanking sequences. For Nova, the partially shuffled datasets still had significant P scores upstream of the RBP-bound sites. Therefore, the nucleotide compositions in the flanking sequences alone cannot generate the positional specificities of Nova, that is, sequences in distant regions could also contribute to the position-specific RNA binding of Nova.

4.3 Discussion

In this study, I developed an efficient algorithm that calculates the structural profiles of RNAs, and implemented it as CapR. It is the fastest software that can be applied to tens of thousands of long RNAs.

Using CapR, I investigated structural specificities of RBP target recognition using several CLIP-seq datasets. My analysis revealed that most RBPs prefer specific structural contexts and some RBPs show positional preferences in their structural recognition. These findings could provide insights into the mechanisms of diseases involving RBPs. FMR1_7, where FMR1 is a causative gene of the fragile X syndrome, was revealed to bind specifically to internal and bulge loops. The observed structural specificity raises

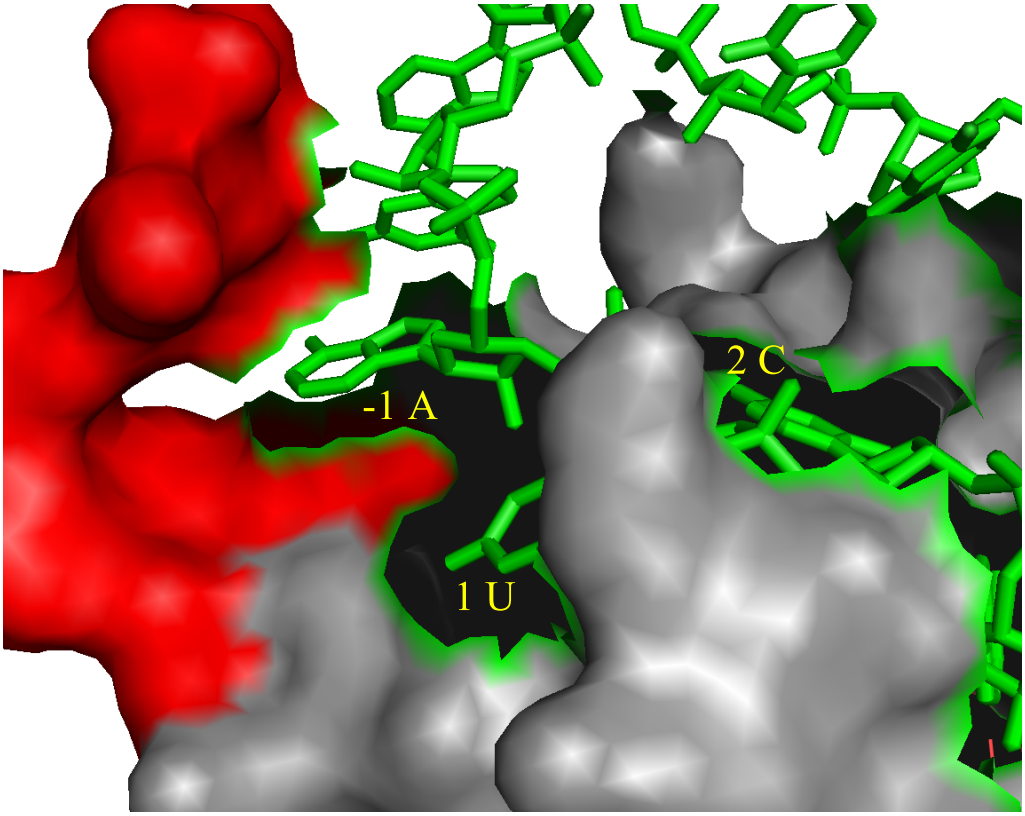


Fig. 4.8 This figure was generated using Pymol. The ten amino acids of the C-terminal tail are shown in red. RNA is represented by green sticks. The positions and the nucleotides are shown in yellow. Position 1 is the start position of the sequential motif.

the possibility that disruption of the internal or bulge loop structures within the target sites of FMR1_7 may cause this disease. On the other hand, the structural specificities of Nova were revealed to be affected by the sequences of distant regions. This means that a mutation of a nucleotide distant from the RBP-bound sites can cause changes to the secondary structures around the RBP-bound sites. Because some disease-associated single nucleotide polymorphisms in non-coding regions are reported to affect RNA secondary structures [123, 124], CapR could also contribute to exploring disease mechanisms behind such polymorphisms.

It has been shown that the secondary structures around the target sites of small interfering RNAs (siRNAs) and miRNAs influence their activities [125, 126]. Kiryu *et al.* showed that the activity of an siRNA depends on the accessibility of the 3′ end of the siRNA target site, and Marin *et al.* showed that the 3′ end of an miRNA target site is more accessible than the other positions [104, 127]. As supported by the X-ray crystal structure of the guide-strand-containing Argonaute [128], these positional tendencies in the accessibility can reflect the kinetic aspects of the siRNA and miRNA binding

mechanisms. I hypothesize that the positional preferences of RBPs discovered in this study also reflect the kinetic aspects of the RBP–RNA interactions. For example, Nova had a positional preference for upstream of the sequential motif site in the unstructured context recognition. In fact, the co-crystal structure of human Nova with the target RNA (PDBID: 1EC6) [129] showed that the area upstream of the sequential motif site interacts with the C-terminal amino acids of Nova [130] (see Fig. 4.8; note that the CLIP-seq data were for a highly similar ortholog, mouse Nova). In addition, the deletion of these C-terminal amino acids inhibits the RNA binding function of Nova [131]. Therefore, the positional preference does likely reflect the kinetic aspects of the RNA binding function of Nova. I argue that this example demonstrates the potential power of ribonomic analysis.

Three future perspectives are envisioned based on the present study. The first perspective is to estimate the sequential and structural specificities simultaneously. Throughout this study, I focused on the RBPs with known and well-defined sequential motifs. Nonetheless, for several RBPs, no such sequential motifs have been identified (for example, FET binds to a highly flexible $\mathrm{UAN}_n\mathrm{Y}$ motif within the hairpin context [108]). To examine the binding specificities of these RBPs, CapR needs to be extended. The second perspective is prediction of RBP-bound sites. Li *et al.* showed that prediction of RBP-bound RNAs *in vivo* was improved by a motif-finding algorithm that considers accessibility [102]. Thus, consideration of structural profiles may also improve the prediction of RBP-bound sites *in vivo*, although I did not directly show this in the present study. Further investigation is necessary for evaluating whether discrimination of RBP-binding sites from a background sequence would be improved using the structural specificities of RBP target recognition. Other factors or subcellular localizations also need to be considered. The third perspective is application of CapR to functional RNAs. For example, the kissing hairpin, which is a hairpin–hairpin interaction that stabilizes RNA structures [132], may be predicted accurately using CapR because CapR enables the calculation of the hairpin loop probabilities. Another target would be small nucleolar RNAs (snoRNAs), where the detection algorithms still have room for improvement [133]. Because snoRNAs are characterized by specific internal loops, they may also be predicted accurately by taking advantage of the accurate calculation of internal loop probabilities by CapR.

4.4 Materials and methods

4.4.1 Rfold model

The state transition rules of the Rfold model are given by

$$\mathrm{Outer} \longrightarrow \epsilon | \mathrm{Outer} \cdot a | \mathrm{Outer} \cdot \mathrm{Stem}$$

$$
\begin{aligned}
\text{Stem} &\longrightarrow b_< \cdot \text{Stem} \cdot b_> | b_< \cdot \text{StemEnd} \cdot b_> \\
\text{StemEnd} &\longrightarrow s_n | s_m \cdot \text{Stem} \cdot s_n (m+n>0) | \text{Multi} \\
\text{Multi} &\longrightarrow a \cdot \text{Multi} | \text{MultiBif} \\
\text{MultiBif} &\longrightarrow \text{Multi1} \cdot \text{Multi2} \\
\text{Multi1} &\longrightarrow \text{MultiBif} | \text{Multi2} \\
\text{Multi2} &\longrightarrow \text{Multi2} \cdot a | \text{Stem}
\end{aligned}
$$

where ϵ represents the null terminal symbol, a is an unpaired nucleotide character, s_k is an unpaired base string of length k and $(b_<, b_>)$ is a base pair. There are seven non-terminal symbols: Outer, Stem, StemEnd, Multi, MultiBif, Multi1 and Multi2. Outer emits exterior bases. Stem emits all the base pairs. StemEnd represents the end of each stem from which a hairpin loop (StemEnd $\longrightarrow s_n$), and internal and bulge loop (StemEnd $\longrightarrow s_m \cdot \text{Stem} \cdot s_n (m+n>0)$), or a multibranch loop (StemEnd $\longrightarrow$ Multi) is emitted. Multi represents a complete multibranch loop. Multi1, Multi2 and MultiBif represent parts of a multibranch loop structure that contains one or more, exactly one, and two or more base pairs in the loop, respectively. Based on this grammar, the structural profiles are calculated by using a variant of the inside and outside algorithm for SCFG. First, I give an illustrative example to show how to calculate the internal loop probabilities from the inside and outside variables $\alpha_s(i,j)$ and $\beta_s(i,j)$ $(i,j=0,\ldots,N$, $s \in \{\text{Outer,Stem,StemEnd,Multi,MultiBif,Multi1,Multi2}\})$. In the subsequent section, I completely describe how to calculate structural profiles.

4.4.2 Algorithm for calculating internal loop probabilities

When a base at position i has an internal loop context, the base i is caught in two base pairs, (j, k) and (p, q) where $j \leq p \leq q \leq k$ (Fig 4.9). Then, the outside structure of base pair (j, k) and the inside structure of base pair (p, q) may take arbitrary structures. The sums of Boltzmann weights of all patterns of the outside structure of base pair (j, k) and the inside structure of base pair (p, q) are represented by outside variable $\beta_{StemEnd}(j,k-1)$ and inside variable $\alpha_{Stem}(p-1,q)$, respectively. Therefore, Boltzmann weights that the base i is caught in two base pairs (j, k) and (p, q) are obtained by the multiplication of $\beta_{StemEnd}(j,k-1)$, the score for transition $StemEnd(j,k-1) \rightarrow Stem(p-1,q)$, and $\alpha_S tem(p,q)$. Here, I sum these Boltzmann weights for all combinations of base pairs (j, k) and (p, q). Finally, I obtain $p(i,I)$ by dividing the sum by the partition function.

The calculation formulas are given by:

$$
\begin{aligned}
w(i,I) &= w_{\text{InternalLeft}}(i,I) + w_{\text{InternalRight}}(i,I) \\
w_{\text{InternalLeft}}(i,I) &= \sum_{j=\max(1,i-W)}^{i} \sum_{k=i+1}^{\min(n,j+W)} \sum_{p=i+1}^{\min(j+C+1,k-1)} \sum_{q=\max(p+4,k-C-p+j-1)}^{k} \\
&\qquad \beta_{\text{StemEnd}}(j,k-1) \cdot \alpha_{\text{Stem}}(p-1,q) \cdot t(\text{StemEnd} \rightarrow (\text{Interior}) \rightarrow \text{Stem})
\end{aligned}
$$

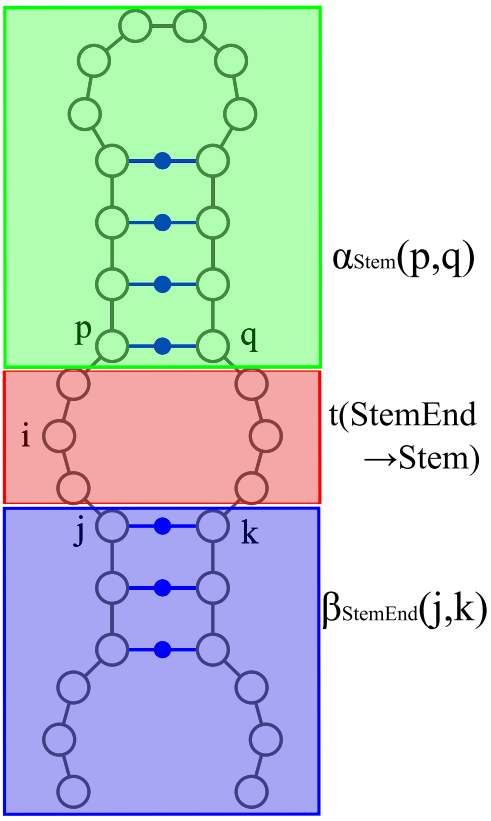


Fig. 4.9 This figure shows the transition patterns that emit an internal loop. This figure was generated by modifying the output of VARNA [134]

$$w_{\text{InternalRight}}(i,I) = \sum_{j=\max(1,i-W)}^{i} \sum_{k=i+1}^{\min(n,j+W)} \sum_{p=j+1}^{\min(j+C+1,i-1)} \sum_{q=\max(p+4,k-C-p+j-1)}^{i} \beta_{\text{StemEnd}}(j,k-1) \cdot \alpha_{\text{Stem}}(p-1,q) \cdot t(\text{StemEnd} \to (\text{Interior}) \to \text{Stem})$$

$$p(i,I) = w(i,I)/Z(x)$$

where $t(s \to s')$ is the score for transition $s \to s'$ and C is the maximal length of the internal and bulge loops. Many software programs, including RNAfold[135], adopt this parameter. In this study, following the default setting of RNAfold, I set $C = 30$.

4.4.3 Algorithms for calculating the structural profile

To calculate the inside and outside variables, a variant of the inside-outside algorithm corresponding to the Rfold model was developed. The inside algorithm is described as follows:

$$\alpha_{\text{Stem}}(i,j) = \sum \begin{cases} \alpha_{\text{Stem}}(i+1,j-1) \cdot t(\text{Stem} \to \text{Stem}) \\ \alpha_{\text{Stem}}(i+1,j-1) \cdot t(\text{Stem} \to \text{StemEnd}) \end{cases}$$

$$\alpha_{\text{Multibif}}(i,j) = \sum \begin{cases} \alpha_{\text{Multi1}}(i,k) \cdot \alpha_{\text{Multi2}}(k,j) \cdot t(\text{MultiBif} \to \text{Multi1} \cdot \text{Multi2}) \\ \text{for } i < k < j \end{cases}$$

$$\alpha_{\text{Multi2}}(i,j) = \sum \begin{cases} \alpha_{\text{Stem}}(i,j) \cdot t(\text{Multi2} \to \text{Stem}) \\ \alpha_{\text{Multi2}}(i,j-1) \cdot t(\text{Multi2} \to \text{Multi2}) \end{cases}$$

$$\alpha_{\text{Multi1}}(i,j) = \sum \begin{cases} \alpha_{\text{Multi2}}(i,j) \cdot t(\text{Multi1} \to \text{Multi2}) \\ \alpha_{\text{MultiBif}}(i,j) \cdot t(\text{Multi1} \to \text{MultiBif}) \end{cases}$$

$$\alpha_{\text{Multi}}(i,j) = \sum \begin{cases} \alpha_{\text{Multi}}(i+1,j) \cdot t(\text{Multi} \to \text{Multi}) \\ \alpha_{\text{MultiBif}}(i,j) \cdot t(\text{Multi} \to \text{MultiBif}) \end{cases}$$

$$\alpha_{\text{StemEnd}}(i,j) = \sum \begin{cases} t(\text{StemEnd} \to (\text{Hairpin})) \\ \alpha_{\text{Stem}}(i',j') \cdot t(\text{StemEnd} \to (\text{Interior}) \to \text{Stem}) \\ \text{for } i \le i' \le j' \le j, 0 < (j-j') + (i'-i) \le C \\ \alpha_{\text{Multi}}(i,j) \cdot t(\text{StemEnd} \to \text{Multi}) \end{cases}$$

$$\alpha_{\text{Outer}}(i) = \sum \begin{cases} 1 \text{ if } j = 0 \\ \alpha_{\text{Outer}}(i-1) \cdot t(\text{Outer} \to \text{Outer}) \\ \alpha_{\text{Outer}}(k) \cdot \alpha_{Stem}(k,i) \cdot t(\text{Outer} \to \text{Outer} \cdot \text{Stem}) \\ \text{for } (i-W) < k < i \end{cases}$$

The outside algorithm is described as follows:

$$\beta_{\text{Outer}}(i) = \sum \begin{cases} 1 \text{ if } i = N \\ \beta_{\text{Outer}}(i+1) \cdot t(\text{Outer} \to \text{Outer}) \\ \alpha_{Stem}(i,k) \cdot \beta_{\text{Outer}}(k) \cdot t(\text{Outer} \to \text{Outer} \cdot \text{Stem}) \\ \text{for } i < k < i+W \end{cases}$$

$$\beta_{\text{StemEnd}}(i,j) = \beta_{\text{Stem}}(i-1,j+1) \cdot t(\text{Stem} \to \text{StemEnd})$$

$$\beta_{\text{Multi}}(i,j) = \sum \begin{cases} \beta_{\text{StemEnd}}(i,j) \cdot t(\text{StemEnd} \to \text{Multi}) \\ \beta_{\text{Multi}}(i-1,j) \cdot t(\text{Multi} \to \text{Multi}) \end{cases}$$

$$\beta_{\text{Multi1}}(i,j) = \sum \begin{cases} \beta_{\text{MultiBif}}(i,k) \cdot \alpha_{\text{Multi2}}(j,k) \cdot t(\text{MultiBif} \to \text{Multi1} \cdot \text{Multi2}) \\ \text{for } j < k < (i+W) \end{cases}$$

$$\beta_{\text{Multi2}}(i,j) = \sum \begin{cases} \beta_{\text{Multi2}}(i,j+1) \cdot t(\text{Multi2} \to \text{Multi2}) \\ \beta_{\text{Multi1}}(i,j) \cdot t(\text{Multi1} \to \text{Multi2}) \\ \beta_{\text{MultiBif}}(k,j) \cdot \alpha_{\text{Multi1}}(k,i) \cdot t(\text{MultiBif} \to \text{Multi1} \cdot \text{Multi2}) \\ \text{for } (j-W) < k < i \end{cases}$$

$$\beta_{\text{MultiBif}}(i,j) = \sum \begin{cases} \beta_{\text{Multi1}}(i,j) \cdot t(\text{Multi1} \to \text{MultiBif}) \\ \beta_{\text{Multi}}(i,j) \cdot t(\text{Multi} \to \text{MultiBif}) \end{cases}$$

$$\beta_{\text{Stem}}(i,j) = \sum \begin{cases} \alpha_{\text{Outer}}(i) \cdot \beta_{\text{Outer}}(j) \cdot t(\text{Outer} \to \text{Outer} \cdot \text{Stem}) \\ \beta_{\text{StemEnd}}(i',j') \cdot t(\text{StemEnd} \to (\text{Interior}) \to \text{Stem}) \\ \text{for } i' \le i < j \le j', 0 < (i-i') + (j-j') \le C \\ \beta_{\text{Multi2}}(i,j) \cdot t(\text{Multi2} \to \text{Stem}) \\ \beta_{\text{Stem}}(i-1,j+1) \cdot t(\text{Stem} \to \text{Stem}) \end{cases}$$

The original computational complexity of both algorithms is $O(NW^3)$; because I adopted the parameter C, it becomes $O(NW^2)$ as described below.

I calculate the structural profiles from the inside and outside variables computed by the inside-outside algorithm. The calculation formula is described as follows:

$$Z = \alpha_O(N)$$

$$p(i,B) = \frac{1}{Z} \left(\sum_{j=\max(1,i-W)}^{i} \sum_{k=i+1}^{\min(n,j+W)} \sum_{p=i+1}^{\min(j+C+1,k-1)} \right.$$

$$
\begin{aligned}
&\beta_{\mathrm{SE}}(j,k-1)\cdot\alpha_{\mathrm{S}}(p-1,k-1)\cdot t(\mathrm{SE}\to(\mathrm{Interior})\to\mathrm{S})\\
&+\sum_{j=\max(1,i-W)}^{i}\ \sum_{k=i+1}^{\min(n,j+W)}\ \sum_{q=\max(j+4,k-C-1)}^{i}\\
&\qquad\beta_{\mathrm{SE}}(j,k-1)\cdot\alpha_{\mathrm{S}}(j,q)\cdot t(\mathrm{SE}\to(\mathrm{Interior})\to\mathrm{S})\Bigg)
\end{aligned}
$$

$$
p(i,E)=\frac{1}{Z}\big(\alpha_{\mathrm{O}}(i-1)\cdot\beta_{\mathrm{O}}(i)\cdot t(\mathrm{O}\to\mathrm{O})\big)
$$

$$
p(i,H)=\frac{1}{Z}\sum_{j=\max(1,i-W)}^{i-1}\ \sum_{k=i+1}^{k=\min(n,i+W)}\beta_{\mathrm{SE}}(j,k-1)\cdot t(\mathrm{SE}\to(\mathrm{Hairpin}))
$$

$$
\begin{aligned}
p(i,I)=\frac{1}{Z}\Bigg(&\sum_{j=\max(1,i-W)}^{i}\ \sum_{k=i+1}^{\min(n,j+W)}\ \sum_{p=i+1}^{\min(j+C+1,k-1)}\ \sum_{q=\max(p+4,k-C-p+j-1)}^{k}\\
&\beta_{\mathrm{SE}}(j,k-1)\cdot\alpha_{\mathrm{S}}(p-1,q)\cdot t(\mathrm{SE}\to(\mathrm{Interior})\to\mathrm{S})\\
&+\sum_{j=\max(1,i-W)}^{i}\ \sum_{k=i+1}^{\min(n,j+W)}\ \sum_{p=j+1}^{\min(j+C+1,i-1)}\ \sum_{q=\max(p+4,k-C-p+j-1)}^{i}\\
&\beta_{\mathrm{SE}}(j,k-1)\cdot\alpha_{\mathrm{S}}(p-1,q)\cdot t(\mathrm{SE}\to(\mathrm{Interior})\to\mathrm{S})\Bigg)
\end{aligned}
$$

$$
p(i,M)=\frac{1}{Z}\begin{cases}\sum_{k=i}^{\min(i+W,n)}\beta_{\mathrm{M}}(i-1,k)\cdot\alpha_{\mathrm{M}}(i,k)\cdot t(\mathrm{M}\to\mathrm{M})\\ \sum_{k=\max(0,i-W)}^{i}\beta_{\mathrm{M2}}(i,k)\cdot\alpha_{\mathrm{M2}}(k,i-1)\cdot t(\mathrm{M2}\to\mathrm{M2})\end{cases}
$$

$$
p(i,S)=\frac{1}{Z}\sum_{j=\max(0,i-W)}^{\min(n,i+W)}\begin{cases}\beta_{\mathrm{S}}(i-1,j)\cdot\alpha_{\mathrm{SE}}(i,j-1)\cdot t(\mathrm{S}\to\mathrm{SE})\\ \beta_{\mathrm{S}}(i-1,j)\cdot\alpha_{\mathrm{S}}(i,j-1)\cdot t(\mathrm{S}\to\mathrm{S})\end{cases}
$$

Here, O is the outer state, S is the stem state, SE is the stem-end state, M is the multi state and M2 is the multi2 state in the Rfold model.

4.4.4 Implementation

I implemented the algorithms in C++ as a program named CapR. CapR exhaustively computes the structural profile $\{p(i,\delta)\}$ for a given RNA sequence with $O(NW^2)$ time and $O(NW)$ memory. I used a portion of the source code from the Vienna RNA package [135].

4.4.5 Data preparation and analysis

To evaluate the accuracy of the structural profiles calculated by CapR, I used 188 structural RNA families in the Rfam 10.0 seed dataset [114]. They are provided as 188 structural alignments with experimentally validated pseudoknot-free structures. By ex-

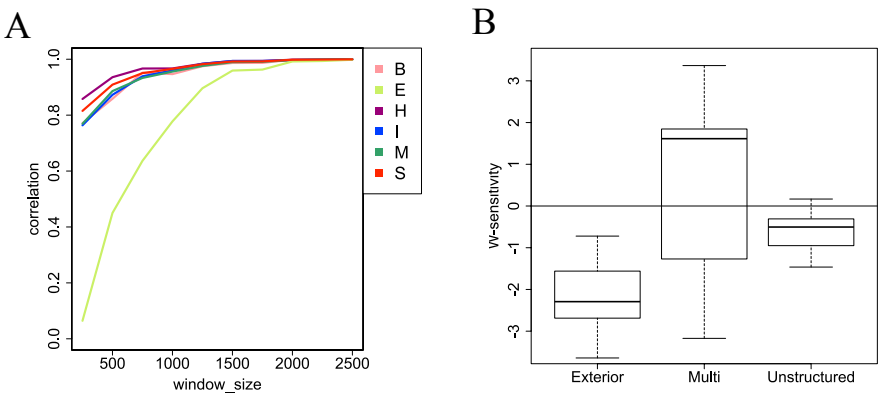


Fig. 4.10 (A) The dependence of structural profiles on the truncated length. The x-axis represents the truncated length. The y-axis represents the Pearson correlation coefficient between the structural profiles of the original sequence and those of the truncated sequences. (B) The W-sensitivities of exterior loop, multibranch loop, and unstructured contexts for CLIP-seq datasets. The y-axis represents the W-sensitivity. The low W-sensitivity means that the highest P-score at $W = 30$ is larger than that at $W = 400$, and vice versa. When W-sensitivity (δ) equals zero, the structural context δ is completely insensitive to the maximal span.

cluding alignment columns with a gap proportion of $\geq$0.5, I obtained 8,775 sequences and 1,039,537 nucleotides.

In the present study, I focused on RBP target recognition. In this application, it should be ineffective to consider transcribed sequences that are too long because regions that are too distant are unlikely to affect the secondary structures around the RBP-bound sites, although my algorithm itself can be applied to long RNAs. Therefore, I investigated how much distance I should take into account. I prepared 100 random RNA sequences 10,100 nucleotides long and truncated them so that the lengths of the flanking sequences of the central 100 bases became $l = 250, 500, ..., 2500$. Then, I calculated the structural profiles of the central 100 bases for each l, and calculated the Pearson correlation coefficient between the structural profiles of the original sequence and those of the truncated sequences. Figure 4.10A shows that the Pearson correlation coefficients were more than 0.99 for $l \geq 2000$. Therefore, I considered 2,000 nucleotides upstream and downstream of the RBP-bound sites in this study.

To investigate the structural characteristics of RNAs around the RBP-binding sites, I downloaded CLIP-seq datasets from the doRina database [115]. I excluded from the analysis CLIP-seq datasets that met one of the following three criteria: (1) well-defined sequential motifs not presented in the original paper of the dataset, (2) datasets for mutant RBPs and (3) the average number of RBP-bound sites (that is the sequential motif-matched sites within the CLIP-seq peak regions defined in doRina) is less than

Table 4.2 Basic statistics of the CLIP-seq datasets

RBP	motif	species(assembly)	number of motif	average number
GLD-1	AYUAAY	*C.elegans*(ce6)	385	1.17
QKI	AYUAAY	*H.sapiens*(hg18)	3054	1.26
Pum2	UGUANAUA	*H.sapiens*(hg18)	1327	1.054
SF2ASF	GAAGAA	*H.sapiens*(hg18)	2721	1.2521
Nova	YCAY	*M.musclus*(mm9)	24019	1.345
Lin28A	AAGNNG	*M.musclus*(mm9))	28642	1.1164
FXR1	ACUK or WGGA	*H.sapiens*(hg18)	2634	1.15
FXR2	ACUK or WGGA	*H.sapiens*(hg18)	12886	1.2112
FMR1_7	ACUK or WGGA	*H.sapiens*(hg18)	46826	1.43478
FMR1_1	ACUK or WGGA	*H.sapiens*(hg18)	93678	1.616

Table 4.3 The numbers of two known sequential motifs for the CLIP-seq data set of the FMRP family

RBP	ACUK	WGGA	Total
FXR1	2435	199	2634
FXR2	9829	3057	12886
FMR1_7	19159	27667	46826
FMR1_1	46364	47314	93678

two. The third criterion was adopted because many RBP-bound sites include false positives. As a result, I selected ten RBPs: GLD-1 (nematode), QKI (human), Pum2 (human), SRSF1 (human), Nova (mouse), Lin28A (mouse), FXR1 (human), FXR2 (human), FMR1_7 (human) and FMR1_1 (human) [116, 99, 117, 118, 119, 120]. When the peak regions spanned just one or two bases, I sought sequential motif-matched sites within ± 10 nucleotides around the peak regions. If no motif-matched sites were found, such peak regions were excluded from the analysis. Then, I extracted $\pm 2,000$ nucleotide sequences around the RBP-bound sites to create the positive datasets. If there existed multiple RBP-bound sites in the same peak region, I averaged the structural profiles around those sites and used them as a single observation. For each gene in RefSeq [136], the transcribed sequence was defined by the genomic region between the most upstream $5'$ position and the most downstream $3'$ position of its mRNA isoforms. To generate the shuffled and partially shuffled datasets, I used the uShuffle software to preserve the di-nucleotide frequencies of the original sequences [137]. The data sizes and other basic statistics of the CLIP-seq datasets are summarized in Table 4.2 and 4.3. In the present study, because the distributions of the structural profiles did not follow a normal distri-

bution, I used the non-parametric Wilcoxon–Mann–Whitney test.

I also examined how the choice of the maximal span W influences the results. I compared the highest P scores of the exterior and multibranch loops with different W because these two loops are sensitive to W. I calculated the ratios of the W sensitivity (δ) of the highest P scores among all positions for each loop δ calculated at $W = 400$ and 30:

$$W - \text{sensitivity}(\delta) = \frac{\text{Highest } P \text{ score for } \delta \text{ at } W = 400}{\text{Highest } P \text{ score for } \delta \text{ at } W = 30}$$

Figure 4.10B is a box plot of the W sensitivity of the exterior loop, multibranch loop and unstructured contexts for all the RBP datasets. The highest P scores of the exterior and multibranch loops were sensitive to W, whereas the highest P score of unstructured context was insensitive to W.

Chapter 5

Conclusion

In conclusion of this thesis, I describe summaries of the researches presented in this thesis and discuss the future works on the basis of these researches. In this thesis, I documented three bioinformatics researches for understanding animal behavior: 1) Development of tracking software for solving occlusion problem; 2) Novel analytic method of worm posture for interpreting relationship between posture and gene; 3) Analysis of secondary structure around target sites of RNA binding proteins.

In chapter 2, I presented GroupTracker, a multiple animal tracking system that accurately tracks individuals even under severe occlusion. As maximum likelihood estimation of Gaussian mixture model whose components can severely overlap is theoretically an ill-posed problem, I devised an Expectation-Maximization scheme with additional constraints on the eigenvalues of the covariance matrix of the mixture components. My system was shown to accurately track multiple medaka (*Oryzias latipes*) which freely swim around in three dimensions and frequently overlap each other. As an accurate multiple animal tracking system, GroupTracker will contribute to revealing unexplored structures and patterns behind animal interactions.

In chapter 3, I showed bioinformatics analysis of postural change patterns of *C.elegans* mutants. I firstly obtained template posture set by Gaussian mixture model, and transformed worm postural change patterns into probabilistic sequences of template postures. Next, by comparing with posture occurrence probabilities of N2 and those of the other strains, I investigated whether the reason why mutants show abnormal postural change patterns is "the usage of different postural set" or not. Then, I revealed several strains (*npr-1*, *npr-3*, *egl-30*, *eat-16*) that shows the similar posture occurrence probabilities to N2 as but different posture transition probabilities from N2. Finally, by comparing postural change speeds of these mutants with that of N2, I revealed that these strains show both "the frequency change of quiescence behavior" and "the change of behavioral speed", but do not very take "the novel postural change patterns".

In chapter 4, I developed a highly efficient algorithm that calculates the probabilities that each RNA base position is located within each secondary structural context for tens of thousands of RNA fragments. The algorithm was implemented as software named

CapR and was applied to the CLIP-seq data of various RBPs. My algorithm demonstrated that several RBPs bind to their target RNA molecules under specific structural contexts. For example, FMR1, which is an RBP responsible for the fragile X syndrome, was found to bind specifically to the internal and bulge loops of RNA. Another example is Nova, a neuron-specific RBP related to a paraneoplastic neurologic disorder, which showed positional preference in the structural contexts of binding targets. Secondary structures are known to be essential for the molecular functions of RNA. As large-scale, high-throughput approaches are becoming more popular in studying RNAs and RBPs, our algorithm will contribute to the systematic understanding of RNA functions and structure-specific RBP-RNA interactions.

Several bioinformatics methods for understanding animal behavior have been developed, but the availability of software for computational ethology is not yet sufficient. Although many tracking systems have been developed, the target species are still limited to model organisms in most cases. In addition, because almost all tracking systems require well-arranged video recording conditions, these systems cannot be applied to animals in outdoor environments. Furthermore, there is insufficient research on methods for the analysis of tracking data. In order to quantify and analyze the diverse behavior of various animals computationally, the development of appropriate software is highly necessary. The field of computational ethology is therefore only beginning to emerge.

It is expected that the integration of computational ethology and advanced technologies in genetics and neuroscience will provide novel insights into the molecular and neural mechanisms of animal behavior. Examples of such measuring and engineering technologies include high-throughput sequencing, genome editing, whole-brain imaging, and optogenetics [138, 139, 140, 141, 142]. In a pioneering study, Vogelstein *et al.* described the relationship between behavior and neurons in *Drosophila* larvae on a large scale by combining optogenetics with large-scale movie data analysis [40]. In addition, applications of computational ethology in biomedical researches are emerging. For example, behavioral analysis of model organism with neurological disorder and drug screening based on automatically quantified behavior are being performed [143, 144].

The 21st century has witnessed the generation and accumulation of large-scale omics data shared in open databases, enabling researchers to make novel discoveries using new techniques and analytical methods. In other words, the sharing and disclosure of scientific data not only prevent the dead storage of data but also promote the development of novel analytical methods. At present, there is an abundance of well-curated databases for biological molecular data, but open databases for the storage of information on animal behavior are few in number [129, 136]. In order to facilitate the study of bioinformatics methods for understanding animal behavior, development and maintenance of

well-curated behavioral databases are urgently needed. I expect that the development of these bioinformatic researches will be key to understanding the fascinating world of animal behavior.

Reference

[1] Aristotle. History of animals. The fourth century BC.

[2] Alex Gomez-Marin, Joseph J Paton, Adam R Kampff, Rui M Costa, and Zachary F Mainen. Big behavioral data: psychology, ethology and the foundations of neuroscience. *Nature Neuroscience*, 17:1455–1462, 2014.

[3] David J Anderson and Pietro Perona. Toward a science of computational ethology. *Neuron*, 84:18–31, 2014.

[4] Christian Rutz and Graeme C Hays. New frontiers in biologging science. *Biology Letters*, 5:289–292, 2009.

[5] Yasuhiko Naito, Daniel P Costa, Taiki Adachi, Patrick W Robinson, Melinda Fowler, and Akinori Takahashi. Unravelling the mysteries of a mesopelagic diet: a large apex predator specializes on small prey. *Functional Ecology*, 27:710–717, 2013.

[6] Máté Nagy, Gábor Vásárhelyi, Benjamin Pettit, Isabella Roberts-Mariani, Tamás Vicsek, and Dora Biro. Context-dependent hierarchies in pigeons. *Proceedings of the National Academy of Sciences*, 110:13049–13054, 2013.

[7] Anthony I Dell, John A Bender, Kristin Branson, Iain D Couzin, Gonzalo G de Polavieja, Lucas PJJ Noldus, Alfonso Pérez-Escudero, Pietro Perona, Andrew D Straw, Martin Wikelski, and *et al.* Automated image-based tracking and its application in ecology. *Trends in Ecology and Evolution*, 29:417–428, 2014.

[8] Hanchuan Peng. Bioimage informatics: a new area of engineering biology. *Bioinformatics*, 24:1827–1836, 2008.

[9] Hanchuan Peng, Alex Bateman, Alfonso Valencia, and Jonathan D Wren. Bioimage informatics: a new category in bioinformatics. *Bioinformatics*, 28:1057–1057, 2012.

[10] Robert F Murphy. A new era in bioimage informatics. *Bioinformatics*, 30:1353–1353, 2014.

[11] Nicolas Chenouard, Ihor Smal, Fabrice De Chaumont, Martin Maška, Ivo F Sbalzarini, Yuanhao Gong, Janick Cardinale, Craig Carthel, Stefano Coraluppi, Mark Winter, and *et al.* Objective comparison of particle tracking methods. *Nature Methods*, 11:281–289, 2014.

[12] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*, volume 3. Prentice Hall, Upper Saddle River, NJ, 2007.

[13] Jasper C Simon and Michael H Dickinson. A new chamber for studying the behavior

of *Drosophila*. *PLOS ONE*, 5:e8793, 2010.

[14] Yair Shemesh, Yehezkel Sztainberg, Oren Forkosh, Tamar Shlapobersky, Alon Chen, and Elad Schneidman. High-order social interactions in groups of mice. *eLife*, 2:e00759, 2013.

[15] Shay Ohayon, Ofer Avni, Adam L Taylor, Pietro Perona, and SE Roian Egnor. Automated multi-day tracking of marked mice for the analysis of social behaviour. *Journal of Neuroscience Methods*, 219:10–19, 2013.

[16] Aharon Weissbrod, Alexander Shapiro, Genadiy Vasserman, Liat Edry, Molly Dayan, Assif Yitzhaky, Libi Hertzberg, Ofer Feinerman, and Tali Kimchi. Automated long-term tracking and social behavioural phenotyping of animal colonies within a semi-natural environment. *Nature Communications*, 4:2018, 2013.

[17] Andrew D Straw, Kristin Branson, Titus R Neumann, and Michael H Dickinson. Multi-camera real-time three-dimensional tracking of multiple flying animals. *Journal of The Royal Society Interface*, 8:395–409, 2010.

[18] Julia Freund, Andreas M Brandmaier, Lars Lewejohann, Imke Kirste, Mareike Kritzler, Antonio Krüger, Norbert Sachser, Ulman Lindenberger, and Gerd Kempermann. Emergence of individuality in genetically identical mice. *Science*, 340:756–759, 2013.

[19] Haruka Imada, Masahito Hoki, Yuji Suehiro, Teruhiro Okuyama, Daisuke Kurabayashi, Atsuko Shimada, Kiyoshi Naruse, Hiroyuki Takeda, Takeo Kubo, and Hideaki Takeuchi. Coordinated and cohesive movement of two small conspecific fish induced by eliciting a simultaneous optomotor response. *PLOS ONE*, 5:e11248, 2010.

[20] Johann Delcourt, Christophe Becco, Nicolas Vandewalle, and Pascal Poncin. A video multitracking system for quantification of individual behavior in a large fish shoal: advantages and limits. *Behavior Research Methods*, 41:228–235, 2009.

[21] Alfonso Pérez-Escudero, Julián Vicente-Page, Robert C Hinz, Sara Arganda, and Gonzalo G de Polavieja. idTracker: tracking individuals in a group by automatic identification of unmarked animals. *Nature Methods*, 11:743–748, 2014.

[22] Joong-Hwan Baek, Pamela Cosman, Zhaoyang Feng, Jay Silver, and William R Schafer. Using machine vision to analyze and classify *Caenorhabditis elegans* behavioral phenotypes quantitatively. *Journal of Neuroscience Methods*, 118:9–21, 2002.

[23] Daniel Ramot, Brandon E Johnson, Tommie L Berry Jr, Lucinda Carnell, and Miriam B Goodman. The Parallel Worm Tracker: a platform for measuring average speed and drug-induced paralysis in nematodes. *PLOS ONE*, 3:e2208, 2008.

[24] Nicholas A Swierczek, Andrew C Giles, Catharine H Rankin, and Rex A Kerr.

High-throughput behavioral analysis in *C. elegans*. *Nature Methods*, 8:592–598, 2011.

[25] Fabrice de Chaumont, Renata Dos-Santos Coura, Pierre Serreau, Arnaud Cressant, Jonathan Chabout, Sylvie Granon, and Jean-Christophe Olivo-Marin. Computerized video analysis of social interactions in mice. *Nature Methods*, 9:410–417, 2012.

[26] Luca Giancardo, Diego Sona, Huiping Huang, Sara Sannino, Francesca Managò, Diego Scheggia, Francesco Papaleo, and Vittorio Murino. Automatic visual tracking and social behaviour analysis with multiple mice. *PLOS ONE*, 8:e74557, 2013.

[27] Heiko Dankert, Liming Wang, Eric D Hoopfer, David J Anderson, and Pietro Perona. Automated monitoring and analysis of social behavior in *Drosophila*. *Nature Methods*, 6:297–303, 2009.

[28] Kristin Branson, Alice A Robie, John Bender, Pietro Perona, and Michael H Dickinson. High-throughput ethomics in large groups of *Drosophila*. *Nature Methods*, 6:451–457, 2009.

[29] Zia Khan, Tucker Balch, and Frank Dellaert. MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:1805–1819, 2005.

[30] Satoru Kato, Takashi Nakagawa, Masato Ohkawa, Kenichiro Muramoto, Osamu Oyama, Akihito Watanabe, Hiroshi Nakashima, Tetsu Nemoto, and Kayo Sugitani. A computer image processing system for quantification of zebrafish behavior. *Journal of Neuroscience Methods*, 134:1–7, 2004.

[31] Olivier Mirat, Jenna R Sternberg, Kristen E Severi, and Claire Wyart. Zebrazoom: an automated program for high-throughput behavioral analysis and categorization. *Frontiers in Neural Circuits*, 7:107, 2013.

[32] Daniel E Bath, John R Stowers, Dorothea Hörmann, Andreas Poehlmann, Barry J Dickson, and Andrew D Straw. FlyMAD: rapid thermogenetic control of neuronal activity in freely walking *Drosophila*. *Nature Methods*, 11:756–762, 2014.

[33] Joan Savall, Eric Tatt Wei Ho, Cheng Huang, Jessica R Maxey, and Mark J Schnitzer. Dexterous robotic manipulation of alert adult *Drosophila* for high-content experimentation. *Nature Methods*, 12:657–660, 2015.

[34] Hueihan Jhuang, Estibaliz Garrote, Xinlin Yu, Vinita Khilnani, Tomaso Poggio, Andrew D Steele, and Thomas Serre. Automated home-cage behavioural phenotyping of mice. *Nature Communications*, 1:68, 2010.

[35] Ulrich Stern, Ruo He, and Chung-Hui Yang. Analyzing animal behavior via classifying each video frame using convolutional neural networks. *Scientific Reports*, 5:14351, 2015.

[36] Mayank Kabra, Alice A Robie, Marta Rivera-Alba, Steven Branson, and Kristin

Branson. JAABA: interactive machine learning for automatic annotation of animal behavior. *Nature Methods*, 10:64–67, 2013.

[37] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[38] Leo Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

[39] André EX Brown, Eviatar I Yemini, Laura J Grundy, Tadas Jucikas, and William R Schafer. A dictionary of behavioral motifs reveals clusters of genes affecting *Caenorhabditis elegans* locomotion. *Proceedings of the National Academy of Sciences*, 110:791–796, 2013.

[40] Joshua T Vogelstein, Youngser Park, Tomoko Ohyama, Rex A Kerr, James W Truman, Carey E Priebe, and Marta Zlatic. Discovery of brainwide neural-behavioral maps via multiscale unsupervised structure learning. *Science*, 344:386–392, 2014.

[41] Gordon J Berman, Daniel M Choi, William Bialek, and Joshua W Shaevitz. Mapping the stereotyped behaviour of freely moving fruit flies. *Journal of The Royal Society Interface*, 11:20140672, 2014.

[42] Balázs Szigeti, Ajinkya Deogade, and Barbara Webb. Searching for motifs in the behaviour of larval *Drosophila melanogaster* and *Caenorhabditis elegans* reveals continuity between behavioural states. *Journal of The Royal Society Interface*, 12:20150899, 2015.

[43] Darren P Croft, Jens Krause, and Richard James. Social networks in the guppy (*Poecilia reticulata*). *Proceedings of the Royal Society of London B: Biological Sciences*, 271:S516–S519, 2004.

[44] David Lusseau. The emergent properties of a dolphin social network. *Proceedings of the Royal Society of London B: Biological Sciences*, 270:S186–S188, 2003.

[45] Danielle P Mersch, Alessandro Crespi, and Laurent Keller. Tracking individuals shows spatial fidelity is a key regulator of ant social organization. *Science*, 340:1090–1093, 2013.

[46] Máté Nagy, Zsuzsa Ákos, Dora Biro, and Tamás Vicsek. Hierarchical group dynamics in pigeon flocks. *Nature*, 464:890–893, 2010.

[47] Tsukasa Fukunaga, Haruka Ozaki, Goro Terai, Kiyoshi Asai, Wataru Iwasaki, and Hisanori Kiryu. CapR: revealing structural specificities of RNA-binding protein target recognition using CLIP-seq data. *Genome Biology*, 15:R16, 2014.

[48] Tsukasa Fukunaga, Shoko Kubota, Shoji Oda, and Wataru Iwasaki. GroupTracker: video tracking system for multiple animals under severe occlusion. *Computational Biology and Chemistry*, 57:39–45, 2015.

[49] Yoshiaki Ono and Tatsumi Uematsu. Mating ethogram in *Oryzias latipes*. *Journal of the Faculty of Science Hokkaido University Series VI. Zoology*, 13:197–202, 1957.

[50] Hung-Yin Tsai and Yen-Wen Huang. Image tracking study on courtship behavior of *Drosophila*. *PLOS ONE*, 7:e34784, 2012.

[51] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 1. Springer, New York, 2006.

[52] Johann Delcourt, Mathieu Denoël, Marc Ylieff, and Pascal Poncin. Video multitracking of fish behaviour: a synthesis and future perspectives. *Fish and Fisheries*, 14:186–204, 2013.

[53] John J Magnuson. An analysis of aggressive behavior, growth, and competition for food and space in medaka (*Oryzias latipes* (*Pisces*, *Cyprinodontidae*)). *Canadian Journal of Zoology*, 40:313–363, 1962.

[54] Teruhiro Okuyama, Saori Yokoi, Hideki Abe, Yasuko Isoe, Yuji Suehiro, Haruka Imada, Minoru Tanaka, Takashi Kawasaki, Shunsuke Yuba, Yoshihito Taniguchi, and *et al.* A neural mechanism underlying mating preferences for familiar individuals in medaka fish. *Science*, 343:91–94, 2014.

[55] Saori Yokoi, Teruhiro Okuyama, Yasuhiro Kamei, Kiyoshi Naruse, Yoshihito Taniguchi, Satoshi Ansai, Masato Kinoshita, Larry J Young, Nobuaki Takemori, Takeo Kubo, and *et al.* An essential role of the arginine vasotocin system in mate-guarding behaviors in triadic relationships of medaka fish (*Oryzias latipes*). *PLOS Genetics*, 11:e1005009, 2015.

[56] Ralf H Anken and Franck Bourrat. *Brain atlas of the medaka fish: Oryzias latipes*. Editions Quae, Versailles, 1998.

[57] Masahiro Kasahara, Kiyoshi Naruse, Shin Sasaki, Yoichiro Nakatani, Wei Qu, Budrul Ahsan, Tomoyuki Yamada, Yukinobu Nagayasu, Koichiro Doi, Yasuhiro Kasai, and *et al.* The medaka draft genome and insights into vertebrate genome evolution. *Nature*, 447:714–719, 2007.

[58] Stuart Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28:129–137, 1982.

[59] David Arthur and Sergei Vassilvitskii. K-means++: the advantages of careful seeding. *ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035, 2007.

[60] Michael PS Brown, William Noble Grundy, David Lin, Nello Cristianini, Charles Walsh Sugnet, Terrence S Furey, Manuel Ares, and David Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proceedings of the National Academy of Sciences*, 97:262–267, 2000.

[61] Sujun Hua and Zhirong Sun. Support vector machine approach for protein subcellular localization prediction. *Bioinformatics*, 17:721–728, 2001.

[62] The *C. elegans* Sequencing Consortium. Genome sequence of the nematode *C.*

elegans: A platform for investigating biology. *Science*, 282:2012–2018, 1998.

[63] Todd W Harris, Joachim Baran, Tamberlyn Bieri, Abigail Cabunoc, Juancarlos Chan, Wen J Chen, Paul Davis, James Done, Christian Grove, Kevin Howe, and *et al.* Wormbase 2014: new views of curated biology. *Nucleic Acids Research*, 42:D789–D793, 2014.

[64] JG White, E Southgate, JN Thomson, and S Brenner. The structure of the nervous system of the nematode *Caenorhabditis elegans*: the mind of a worm. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 314:1–340, 1986.

[65] Christopher J Cronin, Jane E Mendel, Saleem Mukhtar, Young-Mee Kim, Robert C Stirbl, Jehoshua Bruck, and Paul W Sternberg. An automated system for measuring parameters of nematode sinusoidal movement. *BMC Genetics*, 6:5, 2005.

[66] Stanislav Nagy, Charles Wright, Nora Tramm, Nicholas Labello, Stanislav Burov, and David Biron. A longitudinal study of *Caenorhabditis elegans* larvae reveals a novel locomotion switch, regulated by gαs signaling. *eLife*, 2:e00782, 2013.

[67] Julijana Gjorgjieva, David Biron, and Gal Haspel. Neurobiology of *Caenorhabditis elegans* locomotion: where do we stand? *BioScience*, 64:476–486, 2014.

[68] Greg J Stephens, Bethany Johnson-Kerner, William Bialek, and William S Ryu. From modes to movement in the behavior of *Caenorhabditis elegans*. *PLOS ONE*, 5(11):e13914, 2010.

[69] Roland F Schwarz, Robyn Branicky, Laura J Grundy, William R Schafer, and André EX Brown. Changes in postural syntax characterize sensory modulation and natural variation of *C. elegans* locomotion. *PLOS Computational Biology*, 11:e1004322, 2015.

[70] Eviatar Yemini, Tadas Jucikas, Laura J Grundy, André EX Brown, and William R Schafer. A database of *Caenorhabditis elegans* behavioral phenotypes. *Nature Methods*, 10:877–879, 2013.

[71] Greg J Stephens, Bethany Johnson-Kerner, William Bialek, and William S Ryu. Dimensionality and dynamics in the behavior of *C. elegans*. *PLOS Computational Biology*, 4:e1000028, 2008.

[72] Kiran Girdhar, Martin Gruebele, and Yann R Chemla. The behavioral space of zebrafish locomotion and its neural network analog. *PLOS ONE*, 10:e0128668, 2015.

[73] Stanislav Nagy, Marc Goessling, Yali Amit, and David Biron. A generative statistical algorithm for automatic detection of complex postures. *PLOS Computational Biology*, 11:e1004517, 2015.

[74] Ryohei Fujimaki and Satoshi Morinaga. Factorized asymptotic bayesian inference for mixture modeling. *International Conference on Artificial Intelligence and*

Statistics, pages 400–408, 2012.

[75] Michiaki Hamada, Yukiteru Ono, Ryohei Fujimaki, and Kiyoshi Asai. Learning chromatin states with factorized information criteria. *Bioinformatics*, 31:2426–33, 2015.

[76] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 289–300, 1995.

[77] Kazutaka Katoh and Daron M Standley. Mafft multiple sequence alignment software version 7: improvements in performance and usability. *Molecular Biology and Evolution*, 30:772–780, 2013.

[78] Mario De Bono and Cornelia I Bargmann. Natural variation in a neuropeptide Y receptor homolog modifies social behavior and food response in *C. elegans*. *Cell*, 94:679–689, 1998.

[79] Juliet C Coates and Mario de Bono. Antagonistic pathways in neurons exposed to body fluid regulate social feeding in *Caenorhabditis elegans*. *Nature*, 419:925–929, 2002.

[80] Seungwon Choi, Marios Chatzigeorgiou, Kelsey P Taylor, William R Schafer, and Joshua M Kaplan. Analysis of NPR-1 reveals a circuit mechanism for behavioral quiescence in *C. elegans*. *Neuron*, 78:869–880, 2013.

[81] Benny HH Cheung, Merav Cohen, Candida Rogers, Onder Albayram, and Mario de Bono. Experience-dependent modulation of *C. elegans* behavior by ambient oxygen. *Current Biology*, 15:905–917, 2005.

[82] Teresa M Kubiak, Martha J Larsen, Marjorie R Zantello, Jerry W Bowman, Susan C Nulf, and David E Lowery. Functional annotation of the putative orphan *Caenorhabditis elegans* G-protein-coupled receptor C10C6.2 as a FLP15 peptide receptor. *Journal of Biological Chemistry*, 278:42115–42120, 2003.

[83] Lorna Brundage, Leon Avery, Arieh Katz, Ung-Jin Kim, Jane E Mendel, Paul W Sternberg, and Melvin I Simon. Mutations in a *C. elegans*s Gqα gene disrupt movement, egg laying, and viability. *Neuron*, 16:999–1009, 1996.

[84] Yvonne M Hajdu-Cronin, Wen J Chen, Georgia Patikoglou, Michael R Koelle, and Paul W Sternberg. Antagonism between Goα and Gqα in *Caenorhabditis elegans*: the RGS protein EAT-16 is necessary for Goα signaling and regulates Gqα activity. *Genes & Development*, 13:1780–1793, 1999.

[85] Kevin Fitzgerald, Svetlana Tertyshnikova, Lisa Moore, Lynn Bjerke, Ben Burley, Jian Cao, Pamela Carroll, Robert Choy, Steve Doberstein, Yves Dubaquie, et al. Chemical genetics reveals an RGS/G-protein role in the action of a compound. *PLoS Genetics*, 2:e57, 2006.

[86] Tina L Gumienny, Lesley T MacNeil, Huang Wang, Mario de Bono, Jeffrey L Wrana, and Richard W Padgett. Glypican LON-2 is a conserved negative regulator of bmp-like signaling in *Caenorhabditis elegans*. *Current Biology*, 17:159–164, 2007.

[87] Thomas Gallagher, Theresa Bjorness, Robert Greene, Young-Jai You, and Leon Avery. The geometry of locomotive behavioral states in *C. elegans*. *PLOS ONE*, 8:e59865, 2013.

[88] Jesse M Gray, Joseph J Hill, and Cornelia I Bargmann. A circuit for navigation in *Caenorhabditis elegans*. *Proceedings of the National Academy of Sciences*, 102:3184–3191, 2005.

[89] David M Raizen, John E Zimmerman, Matthew H Maycock, Uyen D Ta, Young-jai You, Meera V Sundaram, and Allan I Pack. Lethargus is a *Caenorhabditis elegans* sleep-like state. *Nature*, 451:569–572, 2008.

[90] Young-jai You, Jeongho Kim, David M Raizen, and Leon Avery. Insulin, cGMP, and TGF-β signals regulate food intake and quiescence in *C. elegans*: a model for satiety. *Cell metabolism*, 7:249–257, 2008.

[91] Yoshikazu Ohya, Jun Sese, Masashi Yukawa, Fumi Sano, Yoichiro Nakatani, Taro L Saito, Ayaka Saka, Tomoyuki Fukuda, Satoru Ishihara, Satomi Oka, et al. High-dimensional and large-scale phenotyping of yeast mutants. *Proceedings of the National Academy of Sciences*, 102:19015–19020, 2005.

[92] David Houle, Diddahally R Govindaraju, and Stig Omholt. Phenomics: the next challenge. *Nature Reviews Genetics*, 11:855–866, 2010.

[93] Hui Yu, Boanerges Aleman-Meza, Shahla Gharib, Marta K Labocha, Christopher J Cronin, Paul W Sternberg, and Weiwei Zhong. Systematic profiling of *Caenorhabditis elegans* locomotive behaviors reveals additional components in G-protein Gαq signaling. *Proceedings of the National Academy of Sciences*, 110:11940–11945, 2013.

[94] Kiven E Lukong, Kai-wei Chang, Edouard W Khandjian, and Stéphane Richard. RNA -binding proteins in human genetic disease. *Trends in Genetics*, 24:416–425, 2008.

[95] Kiran Musunuru. Cell-specific RNA -binding proteins in human disease. *Trends in Cardiovascular Medicine*, 13:188–195, 2003.

[96] Jack D Keene. RNA regulons: coordination of post-transcriptional events. *Nature Reviews Genetics*, 8:533–543, 2007.

[97] Donny D Licatalosi, Aldo Mele, John J Fak, Jernej Ule, Melis Kayikci, Sung Wook Chi, Tyson A Clark, Anthony C Schweitzer, John E Blume, Xuning Wang, and *et al.* HITS-CLIP yields genome-wide insights into brain alternative RNA processing. *Nature*, 456:464–469, 2008.

[98] Julian König, Kathi Zarnack, Gregor Rot, Tomaž Curk, Melis Kayikci, Blaž Zupan,

Daniel J Turner, Nicholas M Luscombe, and Jernej Ule. iCLIP reveals the function of hnRNP particles in splicing at individual nucleotide resolution. *Nature Structural and Molecular Biology*, 17:909–915, 2010.

[99] Markus Hafner, Markus Landthaler, Lukas Burger, Mohsen Khorshid, Jean Hausser, Philipp Berninger, Andrea Rothballer, Manuel Ascano, Anna-Carina Jungkamp, Mathias Munschauer, and *et al.* Transcriptome-wide identification of RNA-binding protein and microRNA target sites by PAR-CLIP. *Cell*, 141:129–141, 2010.

[100] Jack D Keene, Jordan M Komisarow, and Matthew B Friedersdorf. RIP-Chip: the isolation and identification of mRNAs, microRNAs and protein components of ribonucleoprotein complexes from cell extracts. *Nature Protocols*, 1:302, 2006.

[101] Timothy L Bailey, Nadya Williams, Chris Misleh, and Wilfred W Li. MEME: discovering and analyzing DNA and protein sequence motifs. *Nucleic Acids Research*, 34:W369–W373, 2006.

[102] Xiao Li, Gerald Quon, Howard D Lipshitz, and Quaid Morris. Predicting *in vivo* binding sites of RNA-binding proteins using mRNA secondary structure. *RNA*, 16:1096–1107, 2010.

[103] Stephan H Bernhart, Ulrike Mückstein, and Ivo L Hofacker. RNA accessibility in cubic time. *Algorithms for Molecular Biology*, 6:3, 2011.

[104] Hisanori Kiryu, Goro Terai, Osamu Imamura, Hiroyuki Yoneyama, Kenji Suzuki, and Kiyoshi Asai. A detailed investigation of accessibilities around target sites of siRNAs and miRNAs. *Bioinformatics*, 27:1788–1797, 2011.

[105] David E Draper. Themes in RNA-protein recognition. *Journal of Molecular Biology*, 293:255–270, 1999.

[106] Tzvi Aviv, Zhen Lin, Giora Ben-Ari, Craig A Smibert, and Frank Sicheri. Sequence-specific recognition of RNA hairpins by the SAM domain of Vts1p. *Nature Structural and Molecular Biology*, 13:168–176, 2006.

[107] Florian C Oberstrass, Albert Lee, Richard Stefl, Michael Janis, Guillaume Chanfreau, and Frédéric HT Allain. Shape-specific recognition in the structure of the Vts1p SAM domain with RNA. *Nature Structural and Molecular Biology*, 13:160–167, 2006.

[108] Jessica I Hoell, Erik Larsson, Simon Runge, Jeffrey D Nusbaum, Sujitha Duggimpudi, Thalia A Farazi, Markus Hafner, Arndt Borkhardt, Chris Sander, and Thomas Tuschl. RNA targets of wild-type and mutant FET family proteins. *Nature Structural and Molecular Biology*, 18:1428–1431, 2011.

[109] David H Mathews, Jeffrey Sabina, Michael Zuker, and Douglas H Turner. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA

secondary structure. *Journal of Molecular Biology*, 288:911–940, 1999.

[110] Ye Ding and Charles E Lawrence. A statistical sampling algorithm for RNA secondary structure prediction. *Nucleic Acids Research*, 31:7280–7301, 2003.

[111] Kishore J Doshi, Jamie J Cannone, Christian W Cobaugh, and Robin R Gutell. Evaluation of the suitability of free-energy minimization using nearest-neighbor energy parameters for RNA secondary structure prediction. *BMC Bioinformatics*, 5:105, 2004.

[112] Hisanori Kiryu, Taishin Kin, and Kiyoshi Asai. Rfold: an exact algorithm for computing local base pairing probabilities. *Bioinformatics*, 24:367–373, 2008.

[113] Sean R Eddy and Richard Durbin. RNA sequence analysis using covariance models. *Nucleic Acids Research*, 22:2079–2088, 1994.

[114] Paul P Gardner, Jennifer Daub, John Tate, Benjamin L Moore, Isabelle H Osuch, Sam Griffiths-Jones, Robert D Finn, Eric P Nawrocki, Diana L Kolbe, Sean R Eddy, and *et al.* Rfam: Wikipedia, clans and the "decimal" release. *Nucleic Acids Research*, 39:D141–D145, 2011.

[115] Gerd Anders, Sebastian D Mackowiak, Marvin Jens, Jonas Maaskola, Andreas Kuntzagk, Nikolaus Rajewsky, Markus Landthaler, and Christoph Dieterich. doRiNA: a database of RNA interactions in post-transcriptional regulation. *Nucleic Acids Research*, 40:D180–D186, 2011.

[116] Anna-Carina Jungkamp, Marlon Stoeckius, Desirea Mecenas, Dominic Grün, Guido Mastrobuoni, Stefan Kempa, and Nikolaus Rajewsky. *In vivo* and transcriptome-wide identification of RNA binding protein target sites. *Molecular Cell*, 44:828–840, 2011.

[117] Jeremy R Sanford, Xin Wang, Matthew Mort, Natalia VanDuyn, David N Cooper, Sean D Mooney, Howard J Edenberg, and Yunlong Liu. Splicing factor SFRS1 recognizes a functionally diverse landscape of RNA transcripts. *Genome Research*, 19:381–394, 2009.

[118] Chaolin Zhang and Robert B Darnell. Mapping in vivo protein-RNA interactions at single-nucleotide resolution from HITS-CLIP data. *Nature Biotechnology*, 29:607–614, 2011.

[119] Jun Cho, Hyeshik Chang, S Chul Kwon, Baekgyu Kim, Yoosik Kim, Junho Choe, Minju Ha, Yoon Ki Kim, and V Narry Kim. LIN28A is a suppressor of ER-associated translation in embryonic stem cells. *Cell*, 151:765–777, 2012.

[120] Manuel Ascano, Neelanjan Mukherjee, Pradeep Bandaru, Jason B Miller, Jeffrey D Nusbaum, David L Corcoran, Christine Langlois, Mathias Munschauer, Scott Dewell, Markus Hafner, and *et al.* FMRP targets distinct mRNA sequence elements to regulate protein expression. *Nature*, 492:382–386, 2012.

[121] Frederick L Moore, Jadwiga Jaruzelska, Mark S Fox, Jun Urano, Meri T Firpo, Paul J Turek, David M Dorfman, and Renee A Reijo Pera. Human Pumilio-2 is expressed in embryonic stem cells and germ cells and interacts with DAZ (Deleted in AZoospermia) and DAZ-like proteins. *Proceedings of the National Academy of Sciences*, 100:538–543, 2003.

[122] Jennifer C Darnell, Kirk B Jensen, Peng Jin, Victoria Brown, Stephen T Warren, and Robert B Darnell. Fragile X mental retardation protein targets G quartet mRNAs important for neuronal function. *Cell*, 107:489–499, 2001.

[123] Matthew Halvorsen, Joshua S Martin, Sam Broadaway, and Alain Laederach. Disease-associated mutations that alter the RNAstructural ensemble. *PLOS Genetics*, 6:e1001074, 2010.

[124] Raheleh Salari, Chava Kimchi-Sarfaty, Michael M Gottesman, and Teresa M Przytycka. Sensitive measurement of single-nucleotide polymorphism-induced changes of RNA conformation: application to disease studies. *Nucleic Acids Research*, 41:44–53, 2013.

[125] Michael Kertesz, Nicola Iovino, Ulrich Unnerstall, Ulrike Gaul, and Eran Segal. The role of site accessibility in microRNA target recognition. *Nature Genetics*, 39:1278–1284, 2007.

[126] Yu Shao, Chi Yu Chan, Anil Maliyekkel, Charles E Lawrence, Igor B Roninson, and Ye Ding. Effect of target secondary structure on RNAi efficiency. *RNA*, 13:1631–1640, 2007.

[127] Ray M Marín, Franziska Voellmy, Thibaud von Erlach, and Jiří Vaníček. Analysis of the accessibility of CLIP bound sites reveals that nucleation of the miRNA: mRNA pairing occurs preferentially at the 3′ -end of the seed match. *RNA*, 18:1760–1770, 2012.

[128] Yanli Wang, Gang Sheng, Stefan Juranek, Thomas Tuschl, and Dinshaw J Patel. Structure of the guide-strand-containing Argonaute silencing complex. *Nature*, 456:209–213, 2008.

[129] Peter W Rose, Chunxiao Bi, Wolfgang F Bluhm, Cole H Christie, Dimitris Dimitropoulos, Shuchismita Dutta, Rachel K Green, David S Goodsell, Andreas Prlić, Martha Quesada, and *et al.* The RCSB Protein Data Bank: new resources for research and education. *Nucleic Acids Research*, 41:D475–D482, 2013.

[130] Hal A Lewis, Kiran Musunuru, Kirk B Jensen, Carme Edo, Hua Chen, Robert B Darnell, and Stephen K Burley. Sequence-specific RNA binding by a Nova KH domain: implications for paraneoplastic disease and the fragile X syndrome. *Cell*, 100:323–332, 2000.

[131] Kirk B Jensen, Kiran Musunuru, Hal A Lewis, Stephen K Burley, and Robert B

Darnell. The tetranucleotide UCAY directs the specific recognition of RNA by the Nova K-homology 3 domain. *Proceedings of the National Academy of Sciences*, 97:5740–5745, 2000.

[132] Pan TX Li, Carlos Bustamante, and Ignacio Tinoco. Unusual mechanical stability of a minimal RNA kissing complex. *Proceedings of the National Academy of Sciences*, 103:15847–15852, 2006.

[133] Jana Hertel, Ivo L Hofacker, and Peter F Stadler. SnoReport: computational identification of snoRNAs with unknown targets. *Bioinformatics*, 24:158–164, 2008.

[134] Kévin Darty, Alain Denise, and Yann Ponty. VARNA: Interactive drawing and editing of the RNA secondary structure. *Bioinformatics*, 25:1974, 2009.

[135] Ronny Lorenz, Stephan HF Bernhart, Christian Hoener Zu Siederdissen, Hakim Tafer, Christoph Flamm, Peter F Stadler, Ivo L Hofacker, and *et al.* Vienna RNA package 2.0. *Algorithms for Molecular Biology*, 6:26, 2011.

[136] Kim D Pruitt, Tatiana Tatusova, Garth R Brown, and Donna R Maglott. NCBI reference sequences (RefSeq): current status, new features and genome annotation policy. *Nucleic acids research*, 40:D130–D135, 2012.

[137] Minghui Jiang, James Anderson, Joel Gillespie, and Martin Mayne. uShuffle: a useful tool for shuffling biological sequences while preserving the k-let counts. *BMC Bioinformatics*, 9:192, 2008.

[138] Ophir Shalem, Neville E Sanjana, Ella Hartenian, Xi Shi, David A Scott, Tarjei S Mikkelsen, Dirk Heckl, Benjamin L Ebert, David E Root, John G Doench, and *et al.* Genome-scale CRISPR-Cas9 knockout screening in human cells. *Science*, 343:84–87, 2014.

[139] Michael L Metzker. Sequencing technologies—the next generation. *Nature reviews genetics*, 11:31–46, 2010.

[140] Kwanghun Chung, Jenelle Wallace, Sung-Yon Kim, Sandhiya Kalyanasundaram, Aaron S Andalman, Thomas J Davidson, Julie J Mirzabekov, Kelly A Zalocusky, Joanna Mattis, Aleksandra K Denisin, and *et al.* Structural and molecular interrogation of intact biological systems. *Nature*, 497:332–337, 2013.

[141] Etsuo A Susaki, Kazuki Tainaka, Dimitri Perrin, Fumiaki Kishino, Takehiro Tawara, Tomonobu M Watanabe, Chihiro Yokoyama, Hirotaka Onoe, Megumi Eguchi, Shun Yamaguchi, and *et al.* Whole-brain imaging with single-cell resolution using chemical cocktails and computational analysis. *Cell*, 157:726–739, 2014.

[142] Karl Deisseroth, Guoping Feng, Ania K Majewska, Gero Miesenböck, Alice Ting, and Mark J Schnitzer. Next-generation optical technologies for illuminating genetically targeted brain circuits. *The Journal of Neuroscience*, 26:10380–10386,

2006.

[143] Andrew D Steele, Walker S Jackson, Oliver D King, and Susan Lindquist. The power of automated high-resolution behavior analysis revealed by its application to mouse models of Huntington's and prion diseases. *Proceedings of the National Academy of Sciences*, 104:1983–1988, 2007.

[144] Jason Rihel, David A Prober, Anthony Arvanites, Kelvin Lam, Steven Zimmerman, Sumin Jang, Stephen J Haggarty, David Kokel, Lee L Rubin, Randall T Peterson, and *et al.* Zebrafish behavioral profiling links drugs to biological targets and rest/wake regulation. *Science*, 327:348–351, 2010.