Multimed Tools Appl
DOI 10.1007/s11042-016-4045-3

Robust tracking of fish schools using CNN for head identification

Shuo Hong Wang[1] · Jing Wen Zhao[1] · Yan Qiu Chen[1]

Received: 3 February 2016 / Revised: 29 August 2016 / Accepted: 30 September 2016

Abstract Tracking individuals in a fish school with video cameras is one of the most effective ways to quantitatively investigate their behavior which is of great value for biological research. However, tracking large numbers of fish with complex non-rigid deformation, similar appearance and frequent mutual occlusions is a challenge task. In this paper we propose an effective tracking method that can reliably track a large number of fish throughout the entire duration. The first step of the proposed method is to detect fish heads using a scale-space method. Data association across frames is achieved via identifying the head image pattern of each individual fish in each frame, which is accomplished by a convolutional neural network (CNN) specially tailored to suit this task. Then the prediction of the motion state and the recognition result by CNN are combined to associate detections across frames. The proposed method was tested on 5 video clips having different number of fish respectively. Experiment results show that the correctness of their identities is not affected by frequent occlusions. The proposed method outperforms two state-of-the-art fish tracking methods in terms of 7 performance metrics.

Keywords Multi-object tracking · Tracking by identification · Fish school · Convolutional neural network

Thanks to National Natural Science Foundation of China, Grant No.61175036 for funding.

Electronic supplementary material The online version of this article (doi: 10.1007/s11042-016-4045-3) contains supplementary material, which is available to authorized users.

✉ Yan Qiu Chen
chenyq@fudan.edu.cn

Shuo Hong Wang
sh_wang@fudan.edu.cn

Jing Wen Zhao
jingwenzhao13@fudan.edu.cn

[1] School of Computer Science, Shanghai Key Laboratory of Intelligent Information Processing, Fudan University, Shanghai, China

Published online: 22 November 2016

1 Introduction

Visually tracking each fish in a large school is the most effective way for quantitatively investigating their behavior patterns and underlying rules, which is of high scientific value for biological and robotic researches [9, 11, 14, 24–27, 33]. Since many fish behavior research experiments use shallow water tank and the fish typically swim around the same horizontal plane, top-view video and 2D tracking can obtain sufficiently informative motion data for behavior research. One sample of captured images is shown in Fig. 1a. Reliably tracking a large fish group in 2D space and keeping individuals' identities throughout the entire period remains a challenging task due to appearance similarity, frequent occlusions and highly non-rigid body deformation.

Being automatic and reliable are the basic requirements for software that strives to be truly helpful for biological research. Several computer software such as ANY-maze® and EthoVision® (2.3 and more recent versions) was developed to track multiple animal individuals and has been widely used in literatures on biological research [9, 32]. These systems are however only capable of tracking a few targets, and they require professional experiment setup. Delcourt et al.'s system can track as many as 100 fish simultaneously but is not suitable for long period tracking [9]. Qian et al. developed a multiple fish tracking system based on a scale-space Determinant of Hessian (DoH) fish head detector and Kalman filter [30]. However, data association in those methods is highly dependent on detection results and motion continuity, and the discriminative information in head images is not fully exploited. Difficulties arise when severe occlusions happen, leading to problematic situation: Individuals may be assigned wrong identities and these errors would propagate throughout the rest of the video. They are therefore not appropriate for researches that require reliably tracking each individual in a group throughout the experiment duration.

Several existing tracking systems combine the detection and tracking stages together, thus faulty detections can be removed to some extent. Kalal et al. [19] proposed the framework Tracking-Learning-Detection (TLD) to track single object, which is divided into tracking, learning and detection 3 subtasks. The tracker follows the target across frames; the detector localizes the object in each frame; and the detector is corrected and updated online by P-N learning. Andriluka et al. [1] proposed to use prior knowledge on possible

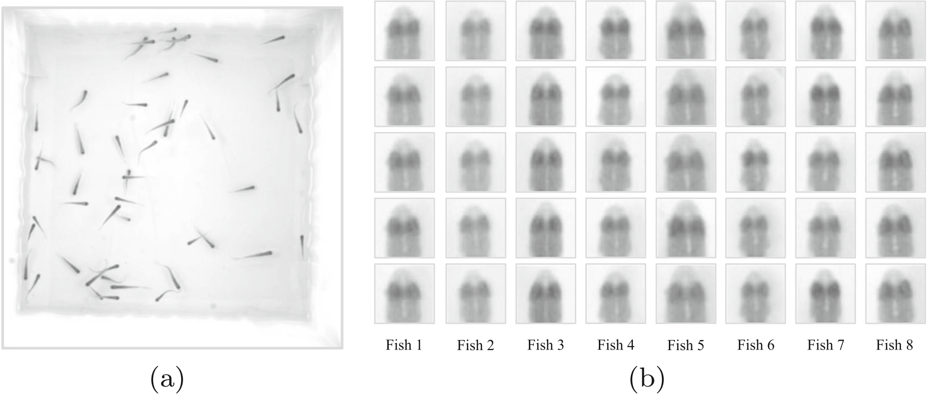


Fig. 1 **a** One frame in a video clip capturing 49 swimming fish; **b** Head images of 8 fish at 5 different frames

articulations and temporal coherency to associate the detection of each individual across frames. The performance of the method depends on the motion model and object specific codebooks constructed by clustering local features. Guo et al. [16] modeled the tracked target's structure with a set of key-point manifolds organized as a graph. Tracking is then done by detecting key-points on each coming frame and matching them with the manifold models. By extracting, learning the features of the tracked targets and then identifying each detected target in each frame, tracking by recognition strategy enables a tracking system to preserve the individuals' identities even after trajectory interruption due to occlusion. This motivates collective behavior researchers to apply image recognition techniques based on handcrafted or learned image features to identify each individual in a fish group during tracking. Visible implant elastomer (VIE) tags [4] which are widely used in biological research are used to mark individuals in a fish school for tracking, such as the work by Delcourt et al. [10]. However, the number of tracked fish is largely limited in order to guarantee sufficient discriminative information for recognition. What's worse, these artificial tags may influence fish behavior. Alfonso et al. proposed an intensity distribution feature and used it to identify the tracked objects in each frame [28]. The identity of each object can be retrieved after trajectory fragmentation, but the discriminative power of such feature is insufficient for identifying similar individuals in a large group. Moreover, for a swimming fish, the intensity distribution of its tail part is not robust due to erratic motion.

Fish head has relatively more stable shape and color distribution compared to other parts of its body [5], making it possible to keep track of fish individuals by tracking their head part [30]. The head images of the same fish keep relatively stable during tracking period, and head images of different fish individuals exhibit otherness. Sample images of 8 different fish at 5 different frames are shown in Fig. 1b. This motivates us to develop a method to learn the discriminative information for identifying fish individuals. However, conventional low-level, handcrafted image features have difficulties in discriminating such subtle differences.

Deep learning methods have shown strong power of recognizing visual patterns. The essence of deep learning based recognition methods is a data driven feature extraction mechanism, therefore the extracted features are consistent with the goal of classification. Convolutional neural network (CNN) is a popular deep learning architecture made up of alternate convolutional and sub-sampling layers to produce a discriminative appearance model and has proved to be effective in a wide range of computer vision applications such as segmentation [7, 15], detection [34, 35], object recognition [18] and classification [20] owing to its ability of extracting sophisticated and high-level image features. For these reasons, CNN has already been implemented in object tracking systems [6, 12, 39]. Most of such systems track a single target, in which CNN is employed to recognize the tracked target from the background. In the proposed method, CNN is adopted to identify the detected targets and the identification results are combined with the predicted motion state to accomplish data association across frames.

Our main contributions are:

1. We solve the data association problem by combining CNN identification results and motion continuity, so that the identities of the tracked targets can be correctly kept throughout the entire duration.
2. We propose a novel weighting strategy to transform the head image patch into a weighted one, which is able to automatically place emphasis on the foreground fish head regions while ignoring irrelevant background pixels, thus offering clean data for CNN training.

3. We propose an online CNN update strategy by maintaining a sample pool, which makes the tracker adaptive to illumination changes.
4. An iterative tracking strategy is proposed which further enhances the tracking performance. Experiment results show that its reliability surpasses the state-of-the-art methods.

2 The proposed method

The goal of a tracking system is to recursively estimate a target's motion state given an observation sequence. Let the state vector for one individual at time t be $X_t = (x_t, y_t, \theta_t)^T$, in which (x_t, y_t) is the coordinate of fish head center, and θ_t is the orientation of fish head. The tracking problem can be solved recursively by state prediction and update at each time step. In multiple object tracking systems, a data association step after prediction is essential, in which detection is associated to the trackers across frames. Typical data association methods include dynamic programming [3], multiple hypotheses tracking (MHT) [31], markov chain monte carlo [38], min-cost flow [29], *etc*.

When swimming in water, fish rarely roll its body. Fish body consists of two connected parts: a rigid head part and a deformable, belt-like tail part. So, fish head can be viewed as undergoing 3-DOF rigid transform during swimming. The color distribution and shape of fish head are thus more robust than those of other parts of its body [5]. Hence, we develop an effective method to track individuals in a fish school by tracking the head region of each fish. The sample head images of 8 fish individuals in 5 different frames are shown in Fig. 1b. It can be seen that the appearance of fish head keeps relatively stable during the tracking period while the appearances of different fish individuals vary. However, these variations are subtle and difficult to be described by conventional feature extraction methods. CNN has shown superior ability of recognizing objects by automatically extracting their sophisticated, high-level features [17, 18, 21, 34]. Hence, in the proposed method, fish head tracking is done by associating detections across frames based on combination of predicted motion state and identification results of CNN.

For each incoming frame, two steps of processing are carried out, namely detection and tracking. The workflow of the two stages is shown in Fig. 2. In the detection stage, scale-space Determinant of Hessian (DoH) is applied to extract head region of each fish, then input image patches for CNN are generated after sample verification. The identification result of CNN is combined with the prediction of state to update the state of each target. During tracking, a sample pool is maintained to store the samples for online update of CNN. The update is performed when the pool is full.

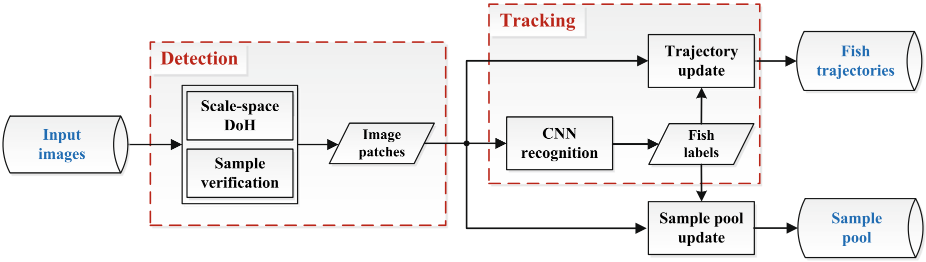


Fig. 2 Workflow of the proposed method

2.1 Fish head detection

To efficiently extract the head region of each fish that resembles a blob, scale-space Determinant of Hessian (DoH) blob detector is applied [30].

Assume (x, y, s) is one point in scale-space calculated by convolving the image with Gaussian kernel function with different scales. The Hessian matrix at this point is:

$$\mathcal{H}(x, y, s) = \begin{bmatrix} L_{xx} & L_{xy} \\ L_{xy} & L_{yy} \end{bmatrix} \tag{1}$$

in which L_{xx}, L_{xy}, L_{yy} are the results of convolution with second order derivative of Gaussian at scale s. DoH is calculated as:

$$\Delta\mathcal{H}(x, y, s) = \left(L_{xx}L_{yy} - L_{xy}^2\right) \times s^4 \tag{2}$$

The blobs in the image which may be the fish heads are located at the extreme points of DoH response in position and scale-space, i.e.,

$$(x_o, y_o, s_o) = \arg\min_{local}(\Delta\mathcal{H}(x, y, s)) \tag{3}$$

After filtering the detection results based on head scale, color and area, the position and orientation of each detected fish head are obtained, denoted as $Z_t = (x_t, y_t, \theta_t)^T$. An example of fitted ellipse on a fish head can be seen in Fig. 3a. For use in CNN, the images of fish head should be aligned to the same position and orientation. In the proposed method, the head image patches are rotated to the up-right position based on the orientation obtained by scale-space DoH. Then the rotated head image patch is cropped to $l_p \times l_p$ image patch, denoted as I, as shown in Fig. 3c, and we choose $l_p = 48$.

To eliminate minor inaccuracy of head orientation calculated by scale-space DoH, simulated annealing is applied to refine the orientation of fish head and rectify the fish head to

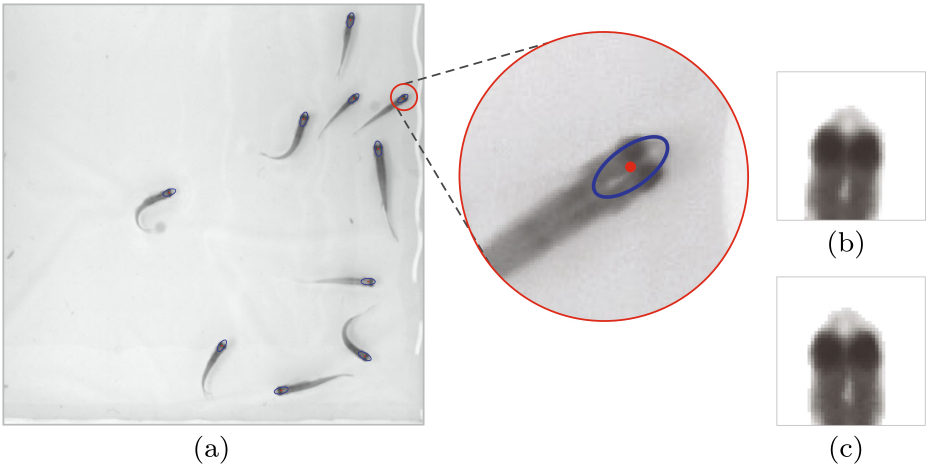


Fig. 3 Fish head detection and CNN preprocessing. **a** Fish head detection result by scale-space DoH; **b** Normalized head image patch before further rectification by simulated annealing; **c** Rectified head image

the up-right position. Considering that the left and right half of fish head are laterally symmetric about their body axis [13], the object function is defined as sum of absolute intensity difference of the corresponding pixels in left and right half of the normalized head image patch:

$$\tau = \arg\min_{\theta} \sum_{i=1}^{l_p} \sum_{j=1}^{l_p/2} |I(i, l_p + 1 - j) - I(i, j)| \tag{4}$$

where $I(i, j)$ denotes pixel intensity of the image patch. The normalized head image patch before and after refinement is shown in Fig. 3b and c respectively.

2.2 Sample verification

To eliminate possible detection error, it is necessary to check if an image patch is a real up-right fish head image. Histogram of Oriented Gradient (HOG) is an appearance and shape descriptor that shows good performance in many object detection applications because of being invariant to geometrical deformation and optical distortion. Besides, HOG is often combined with the support vector machine (SVM) classifier to accomplish detection or recognition tasks [8]. For the above considerations, we verify if the detected fish head is a real fish head at correct position with a pre-trained HoG based SVM classifier. We extract HoG feature of fish head image patch as positive samples and randomly extract HoG feature of other image patches as negative samples. Then, an SVM classifier is trained based on these samples to judge if an image patch is a fish head at up-right position.

2.3 Tracking by CNN identification

2.3.1 Weighted head images

The fish head images extracted in detection step need to be normalized before being input into CNN. When occlusion occurs, if the front part of the fish head region is not occluded, the fish head point can still be correctly detected by the DoH based detector, but it will be difficult to segment the overlapped fish and extract the clean head image of the target. As the case shown in Fig. 4b, the occluded fish head that is not well segmented will result in incorrect CNN identification result. Inspired by the observation that fish heads have similar shape, we propose a strategy to solve the problem by emphasizing the fish head region and ignoring the background region. The strategy is realized by using a weighted head image as CNN input, the procedure is described as below:

(a) Compute the mean image of training samples of fish head images and change the mean image into a binary one using the thresholding method. The resultant binary image is denoted as I_b, where $I_b(x, y) = 1$ denotes all fish head pixels and $I_b(x, y) = 0$ indicates background pixels. The weighting value image I_w is calculated as:

$$I_w(x, y) = \mathcal{N}(d(x, y); 0, \Sigma) \tag{5}$$

where $d(x, y)$ is the minimum Euclidean distance between pixel (x, y) and (x', y'), for all pixels (x', y') that satisfy $I_b(x', y') = 1$. The visualization of I_w is shown in Fig. 4a.

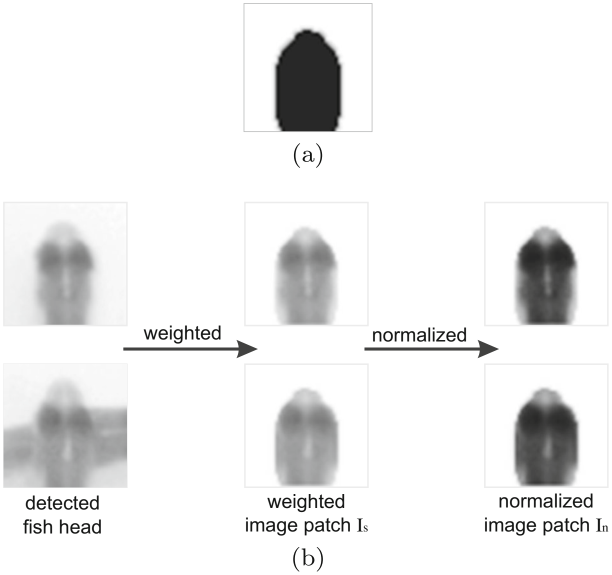


Fig. 4 **a** Visualization of the weight values; (Colors of the images are inverted for clarity, value of *black pixels* equals to 1 *while white* ones is 0.); **b** Steps to compute normalized head images

(b) The weighted image sample I_s is thus calculated by:

$$I_s(x, y) = 1 - I_w(x, y) \cdot [1 - I(x, y)], \quad for\,all\,(x, y) \in I \tag{6}$$

in which $I(x, y)$ is the background subtracted head image patch, but the background is sometimes not clean due to occlusion and changing illumination.

The weighted image I_s is then normalized by subtracting mean pixel value and dividing standard deviation, resulting in a normalized image I_n. The procedure of computing normalized image samples is shown in Fig. 4b. Two samples are shown in the figure. The sample in the second row is an occlusion case, where another fish is right under the target fish. If image patch after normalization without weighting strategy is directly used as CNN input, identification error may occur. After applying the weighting strategy, clean head image of a single fish denoted as I_n is obtained.

2.3.2 CNN architecture

The CNN used in the proposed system consists of 3 convolutional layers C1, C2, C3, 3 subsampling layers S1, S2, S3 and a full connection layer F1. The detailed architecture is shown in Fig. 5. The size of input image sample of the CNN is 48 × 48. The kernel size of C1, C2, C3 is 5 × 5, 5 × 5, 4 × 4 with number of feature maps equals to 6, 12, 24 respectively. The number of feature maps of S1, S2, S3 is 6, 12 and 24. The size of output image of S3 is 3 × 3. Then output of all feature maps in S3 is reshaped and concatenated into a vector whose size is 216 × 1. At last a softmax function is employed to calculate the $n_{fish} \times 1$ output vector of CNN, denoted as O^i (n_{fish} is the number of tracked fish). Each element O^i_l, $(l = 1, ..., n_{fish})$ represents the probability that the fish head sample i belongs to fish l. The identification result of each sample i is $r^i = \arg\max_l(O^i_l)$ with probability $p(r^i) = O^i_{r^i}$. Figure 6 shows the output of C1 layer of one sample image.

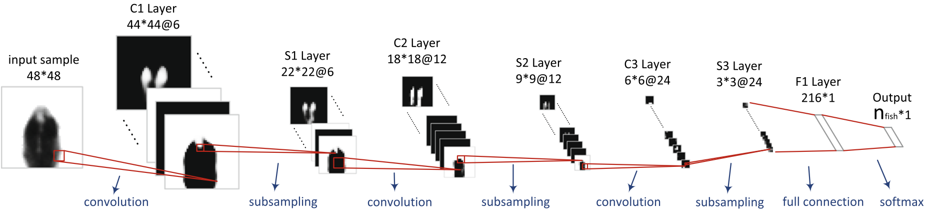


Fig. 5 CNN architecture

2.3.3 CNN training

To reduce the time cost of training stage, we employ the mini-batch stochastic gradient descent strategy [20]. The value of loss function is summed upon each mini-batch and then backpropagated from output layer to input layer.

To reduce manual work on sample selection, we apply a semi-automatic strategy. Firstly, fish heads in each sample image are detected and tracked using the tracking system proposed in [30]. Then, the correctness of tracking results is manually checked and the incorrect results will be removed from the sample image set. For each fish individual, we choose 300 head image patches as samples.

2.3.4 Detection association and state update

CNN has shown outstanding performance in various image recognition applications, its recognition result in most times can be used as the basis for motion state estimation. However, CNN may fail to correctly identify the detected fish in a few cases. If data association is solely dependent on CNN, it will lead to tracking errors such as target lost and ID switches. To eliminate such errors, we combine the predicted state based on motion characteristics and CNN identification results to associate detections across frames and update each target's state.

When the frame rate of the camera is set to a relatively high value ($\geqslant$100fps), displacement of each individual between two consecutive frames is small and nearly uniform. From the Bayesian perspective, if a system is linear and dynamic and the posterior density at each time step is Gaussian, then the state and observation sequence can be formulated as:

$$\begin{aligned} X_t &= FX_{t-1} + \omega_{t-1} \\ Z_t &= HX_t + \nu_t \end{aligned} \tag{7}$$

in which F and H are state transition and observation matrix of the target and are both defined as identity matrix in the proposed method. ω and ν are zero-mean Gaussian noise of state and observation respectively.

The state prediction stage aims to estimate the prior pdf of state at time t, which satisfies:

$$p(X_t|Z_{1:t-1}) = \mathcal{N}(X_t; X_{t|t-1}, \Sigma_{t|t-1}) \tag{8}$$

Fig. 6 Sample output of C1 layer

in which $X_{t|t-1}$ and $\Sigma_{t|t-1}$ are mean and covariance of predicted state X at time t. Inspired by Kalman filter [2], mean of predicted state $X_{t|t-1}$ can be calculated by:

$$X_{t|t-1} = FX_{t-1|t-1} + \omega_{t-1} \tag{9}$$

Before state update, a matching score denoted as $s\left(Z_t^i, X_t^{r^i}\right)$ is calculated to evaluate the probability of a detection i being matched with a target r^i at time t. r^i is the identity of detection i recognized by CNN. The score is calculated by the combination of two terms, (i.e., CNN identification term and detection term), formulated as:

$$s\left(Z_t^i, X_t^{r^i}\right) = \alpha p(r^i) + \beta p_{\mathcal{N}}\left(Z_t^i - X_{t|t-1}^{r^i}\right) \tag{10}$$

- CNN Identification Term

  The term $p(r^i)$ measures the probability the detection i belongs to target i calculated by CNN, in which r^i is the identification result for detection i by CNN.
- Detection Term

  The term $p_{\mathcal{N}}(Z_t^i - X_{t|t-1}^{r^i})$ measures the distance between detection Z_t^i and predicted state of target $X_{t|t-1}^{r^i}$, which is evaluated under a Normal distribution with mean 0 and covariance Σ^{r^i}, written as:

$$p_{\mathcal{N}}\left(Z_t^i - X_{t|t-1}^{r^i}\right) \sim \mathcal{N}\left(Z_t^i - X_{t|t-1}^{r^i}; 0, \Sigma^{r^i}\right) \tag{11}$$

If $s\left(Z_t^i, X_t^{r^i}\right) > \varepsilon_p$, the identification result of detection i is judged as a correct one, then the state of the corresponding target r^i will be updated as:

$$X_{t|t}^{r^i} = Z_t^i \tag{12}$$

For the identification results that are judged as incorrect and those targets that are not detected mainly caused by occlusion, an SVM classifier is employed to verify if the image patch determined by the predicted state $X_{t|t-1}^{r^i}$ is a real fish head image (detail of sample verification is introduced in Section 2.2). If so, the target's state is updated as the predicted value, written as:

$$X_{t|t}^{r^i} = X_{t|t-1}^{r^i} \tag{13}$$

Otherwise, the target's state will not be updated and the trajectory will be interrupted. A relinking step after tracking is proposed to solve trajectory fragmentation, which is introduced in Section 2.4.

2.3.5 CNN online update

Preparing training samples is a laborious and time-consuming work. To reduce the number of manually prepared training samples while ensuring tracking performance, a good strategy of CNN update is essential. If the tracked objects in the previous frames are all picked up as training samples, the number of samples will increase quickly which is impractical for efficient and effective CNN update. In the proposed tracking method, a sample pool is maintained and a sample selection strategy is proposed to add samples into sample pool during tracking.

After all the extracted fish head image patches in one frame are processed by CNN, each image patch I_s^i with low CNN output softmax function value that satisfies $\varepsilon_\xi < p(r^i) < \varepsilon_O$ is verified by HoG based SVM, as introduced in Section 2.2. If it is a real fish head, I_s^i and its corresponding identity label r^i will be used as a sample for CNN update. To prevent CNN overfitting to those less convincing samples, the image patches that obtain high softmax function value which satisfies $p(r^i) \geqslant \varepsilon_O$ and their identity labels r^i are also put into the sample pool. These convincing samples occupy a large part of the images, to reduce the number of samples for CNN update, these convincing samples are selected with a lower probability:

$$p = \frac{1}{n_{fish}} \tag{14}$$

the selection probability is independent on time to prevent CNN overfitting to these recent samples [39].

The size of the sample pool is set as N_{sp}. CNN update will be performed when the sample pool is full during tracking process.

2.3.6 *Iterative offline tracking*

To further improve the performance of the tracking method and reduce the training samples needed before tracking, an iterative tracking strategy is employed. When the entire video has been processed, the parameters of CNN are updated during tracking process. The whole video can be processed the second, third or even more times. Experiment results show that in this way parameters of CNN can be further optimized and the tracking performance can be improved, some detected image patches that cannot be correctly identified in the previous iterations may be correctly identified in the next iterations. To achieve better tracking performance, the more iterations the better. However, the computation overhead and tracking performance need to be balanced. A detailed evaluation of tracking performance under different times of iterations is discussed in Section 3.3.

2.4 Trajectory relinking

In the proposed method, when the CNN identification result of a detection is judged as incorrect or when the target is not detected mainly caused by occlusion, an SVM classifier is employed to verify if the image patch corresponding to the predicted motion state is a real fish head at up-right position. If it is, the target's state is updated as the predicted value, otherwise, the state will not be updated, which will lead to trajectory fragmentation. Though this kind of situation does not happen very often, to further improve tracking performance and obtain complete trajectory for each target, we add a postprocessing step to relink the fragmented trajectories.

As the proposed method is an identity preserved method, for each target before fragmentation, we can easily find the corresponding trajectory of the target after fragmentation. Considering that the frame rate is relatively high ($\geqslant$100fps), the displacement of fish in several consecutive frames is small. If the time interval of fragmentation is less than ε_t (we set to 5 frames), then we assume the fish is moving with constant velocity during the fragmentation interval, thus the position of the fish head at each moment during fragmentation can be determined. For the fragmentation with time interval longer than ε_t frames, the motion during the fragmentation interval is considered to be unpredictable and the two fragmented trajectories will not be relinked.

3 Results and discussion

3.1 Experiment setup

To evaluate the performance of the proposed method, 5 videos of zebrafish (*Danio rerio*) school at different density (10, 20 49, 11 and 27 fish respectively) are captured with one high speed monochrome camera (IO Industries Canada, Flare 4M 180-CL, 2048w×2040h pixels at 100fps). The size of the transparent acrylic water tank is 20cm×20cm, and the depth of water is about 8cm. The light source and the camera are both located above the water tank. We use this kind of top light illumination because this is similar to the condition of fish's natural habitat.

The proposed tracking method is implemented with MATLAB^{TM}, and the implementation of CNN is based on MatConvNet [36]. The computer hardware includes an octa-core Intel i7-2600, 3.40GHz CPU, 16GB RAM.

3.2 Performance metrics

We use 10 widely used performance evaluation metrics as shown in Table 1 to measure the detection and tracking performance [22].

To evaluate the performance of the detection stage of the proposed method, we randomly selected 300 frames from each of the 5 data sets DS1∼DS5 and manually annotated the head point (approximately at the middle of the two eyes) of each recognizable object in each frame. We used 'Miss ratio' and 'Error ratio' to evaluate the performance of the detection stage. Definition of the 2 metrics are shown in Table 1. Correct detection of a fish is defined as: the distance between the manually annotated fish head and one detected fish head point

Table 1 Description of the performance metrics

Metric	Description
Identification Rate (IR)	Average identification rate by CNN.
Miss ratio	Percentage of fish that are undetected in all frames.
Error ratio	Percentage of wrongly detected fish in all frames.
Precision (P)	Sum of correctly tracked objects in all frames / total groundtruth objects in all frames, the larger the better.
Recall (R)	Sum of correctly tracked objects in all frames / total tracked objects in all frames, the larger the better.
F1-score (F1)	Harmonic mean of precision and recall, the larger the better.
Mostly Tracked Trajectories (MT)	Percentage of GT which are correctly tracked more than 80 % in length, the larger the better.
Mostly Lost Trajectories (ML)	Percentage of GT which are correctly tracked less than 20 % in length, the smaller the better.
Fragments (Frag)	Average total number of times a groundtruth trajectory is interrupted in the output tracking results, the smaller the better.
ID Switches (IDS)	Average total number of times that a resulting trajectory switches its matched groundtruth identity with another trajectory, the smaller the better.

Ground Truth (GT) means the number of ground truth trajectories, which is the same as the number of fish in our case. Occlusion probability (OP) measures the probability that one fish is in an occlusion event at one moment is less than a threshold ε_d, which we set to 10 pixel. For a groundtruth head point, if there is no detected head point within threshold ε_d, then the object is considered as missed. If for a detected head point there is no groundtruth head point within the same threshold ε_d, then the detection is considered as a wrong one.

The tracking performance of the proposed method was compared with other two state-of-the-art fish tracking systems, namely idTracker [28] and Qian et al.'s [30]. idTracker is an open source multiple object tracking system based on intensity distribution of the object's body, which can keep the objects' identities during tracking period. Qian et al.'s method is based on a scale-space DoH fish head detector and Kalman filter. The groundtruth trajectories (the head position of each fish individual in each frame) are obtained in a semi-automatic way, i.e., we first perform fish tracking using Qian et al.'s method, then manually check the tracking result and correct the tracking errors. The proposed approach and Qian et al.'s method track the center of fish head based on the detection result of scale-space Determinant of Hessian (DoH) blob detector. And correct tracking of an object in one frame must satisfy that the distance between the tracking result (center of fish head) and groundtruth is within a given threshold ε_d. Different from the proposed approach and Qian et al.'s method, idTracker tracks the center of each detected fish blob in each frame, as a result, the center point falls on fish body within a larger range. Hence, we relax the criteria of correct tracking for idTracker. Those tracking results (fish positions) that fall on the right fish blobs in the image are judged as correct. For fair comparison of the methods that keep or do not keep the individuals' identities, the judgement of correct tracking which is used to calculate precision, recall and F1-score does not take the correctness of identity into consideration. The performance of identity preserving is measured by ID Switches (IDS).

3.3 Experiment results and discussions

We captured 5 video clips (denoted as DS1∼DS5 respectively) to evaluate the performance of the proposed method. Each video clip consists of 2000 frames. The size of fish school in each video clip is 10, 20, 49, 11 and 27 respectively. In addition, the illumination in DS4 and DS5 keeps changing and there are small stones scattered on the bottom of the water tank. Details of the 5 data sets are described in Table 2. All of the 5 video clips are now available online at http://www.cv.fudan.edu.cn/fishtracking.htm.

The resultant trajectories of the 5 data sets are shown in Fig. 7a–e respectively. The supplementary videos show the tracking results of 300 frames of each video clip DS1∼DS5.

Evaluation of the proposed method was conducted from 3 aspects, i.e., performance of detection, performance of CNN identification and performance of tracking. The evaluation results are presented in Section 3.3.1–3.3.3 respectively.

3.3.1 Performance of detection

As shown in Table 3, Occlusion Probability (OP) directly affects the performance of detection stage, but both 'Miss ratio' and 'Error ratio' are less than OP, which indicates that when the fish body is occluded but the head isn't, it can still be detected at most times. Moreover, the changing illumination and stones in the water tank also have influence on detection performance, but most of these detection failures can be eliminated in tracking stage. If an object is failed to be detected (detection miss) probably due to occlusion, the state of it will be updated as the predicted value. Detection false positives will be further identified by

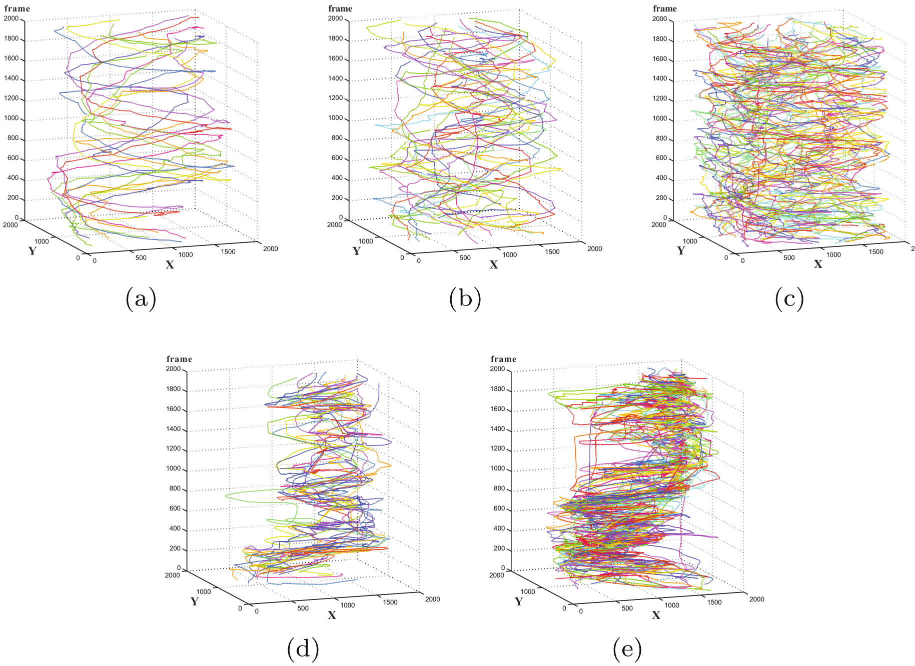
Table 2 Description of the 5 data sets

Data Set	Description	OP (%)
DS1	10 zebrafish with uniform illumination	11.4
DS2	20 zebrafish with uniform illumination	12.4
DS3	49 zebrafish with uniform illumination	24.3
DS4	11 zebrafish with changing illumination, stones on the bottom	12.9
DS5	27 zebrafish with changing illumination, stones on the bottom	21.5

CNN, in which the false positives are either with low CNN classification scores or far away from the state predictions of existing trackers, thus the wrong detection will be discarded.

3.3.2 Performance of CNN identification

We applied the metric Identification Rate (IR) to evaluate the performance of CNN identification. It can be concluded from the results presented in Table 3 that IR is hardly affected by the changing illumination and unclean background. But IR is largely affected by OP, because when OP is high, there exist few occlusion cases which result in detection errors such as serious head orientation deviation. The proposed sample rectification and weighting



(a) (b) (c)

(d) (e)

Fig. 7 Resultant trajectories on DS1∼DS5. Z-axis indicates the frame number, X-axis and Y-axis are coordinates of the image plane, different colors represent different individuals. **a** Results on DS1 (10 fish); **b** Results on DS2 (20 fish); **c** Results on DS3 (49 fish); **d** Results on DS4 (11 fish); **e** Results on DS5 (27 fish)

Table 3 Detection performance and CNN identification rates of the 5 data sets

Data Set	Miss ratio (%)	Error ratio (%)	IR (%)
DS1	0.35	0.05	85.3
DS2	0.40	0.25	84.5
DS3	0.65	0.53	74.4
DS4	0.55	0.45	84.4
DS5	1.19	1.04	82.6

strategy may fail to handle with them. However, most of such failures can be resolved in tracking stage using motion continuity cue.

3.3.3 Performance of tracking

The detailed evaluation results for the tracking stage on the 5 data sets are presented in Table 4. The results show that, the proposed method outperforms others on most of the metrics. As identity preserved tracking systems, idTracker and the proposed method achieved high MT and low IDS value on DS1 and DS2. But idTracker failed to run DS3 mainly caused by the large group size and its performance dropped a lot if the illumination is not controlled and the background is not clean (DS4 and DS5). Qian et al.'s method performed well and achieved high F1 on data set DS1 and is more robust than idTracker when illumination undergoes changes. However, when the group size and occlusion frequency increases,

Table 4 Performance comparison of the proposed tracking method and other two state-of-the-art fish tracking systems

Data Set	Method	P	R	F1	MT (%)	ML (%)	Frag	IDS
DS1	Ours	0.986	0.994	0.990	100.0	0.0	1.9	0.5
	idTracker	0.969	0.983	0.976	100.0	0.0	2.9	0.4
	Qian et al.	0.967	0.973	0.970	40.0	0.0	6.5	0.7
DS2	Ours	0.968	0.992	0.980	100.0	0.0	1.6	0.6
	idTracker	0.831	0.952	0.887	95.0	5.0	6.3	0.7
	Qian et al.	0.949	0.964	0.956	15.0	25.0	5.4	1.3
DS3	Ours	0.933	0.970	0.951	100.0	0.0	4.1	1.4
	idTracker	—	—	—	—	—	—	—
	Qian et al.	0.856	0.951	0.901	4.1	46.9	12.2	1.8
DS4	Ours	0.971	0.981	0.976	100.0	0.0	2.3	0.7
	idTracker	0.884	0.890	0.887	63.6	36.4	3.8	19.1
	Qian et al.	0.957	0.960	0.958	18.2	36.4	5.9	2.9
DS5	Ours	0.917	0.968	0.942	100.0	0.0	4.3	1.6
	idTracker	0.556	0.965	0.706	0.0	77.8	24.8	18.3
	Qian et al.	0.878	0.921	0.899	3.7	70.4	17.6	6.4

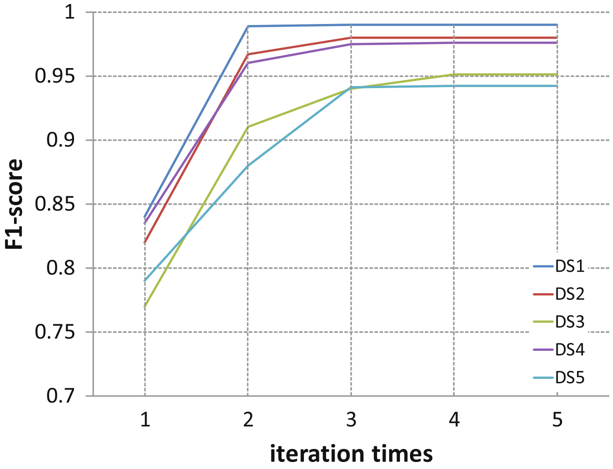


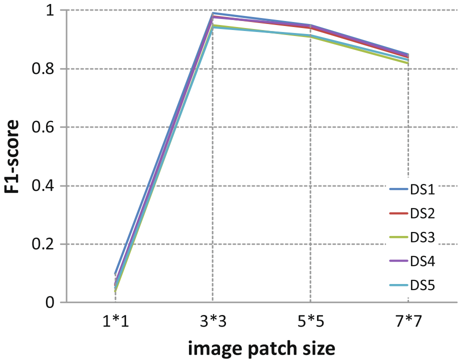
Fig. 8 Relationship between F1-score and iteration times

its corresponding F1 will drop a lot. What's worse, their method does not keep individuals' identities, resulting in low MT, high ML and IDS, which makes their method not capable of reliable long term tracking. Moreover, switch of individuals' identities is not preferred in researches that require reliable tracking.

The proposed tracking method employs an iterative tracking strategy, tracking performance can be improved by adding iteration rounds. The tracking performance (measured by F1-score) on the 5 data sets with different iteration counts are shown in Fig. 8. According to the results, we can conclude that the performance of the tracking method improves remarkably after the 2nd iteration, and becomes stable after the 3rd iteration. In conclusion, 3 iterations may be the most recommendable for the proposed method considering the balance of system efficiency and performance.

For the CNN architecture in the proposed method, the output sample image size of the last sub-sampling layer (S3 layer) is 3×3, here we discuss why we choose such size. We have tried to convolute and down-sample the image patch to other sizes before full connection layer, such as 1×1, 5×5, 7×7, *etc*. The comparison of tracking performance (measured by F1-score) corresponding to each size is shown in Fig. 9. The proposed method performs

Fig. 9 Relationship between F1-score and final image patch size after S3 layer



best with final image patch size equals to 3×3, and worst with size 1×1, which is different to common CNN based image recognition or classification applications [21, 23, 37].

4 Conclusions

We have proposed in this paper a tracking method capable of tracking individuals in a large fish school and keeping their correct identities. The data association step is accomplished by combining prediction of motion state and CNN identification results. In case of occluded fish head, changing illumination and noisy background, a novel weighted normalized fish head image is employed as input to CNN to identify each individual. The parameters of CNN are updated adaptively during the tracking process to adapt to slight illumination and fish appearance changes and reduces the amount of samples needed in CNN training stage before tracking. An iterative tracking strategy further improves performance of the proposed method.

Experiment results show that this method is capable of reliably tracking individuals in a zebrafish school exhibiting frequent occlusions. And the identity of each fish individual can be kept even when the trajectory is fragmented. With this advantage, the complete trajectory of each individual can be obtained, thus more thorough investigation into individual's collective behavior in a large group can be conducted. To other fish species, our method is also applicable resulted from two characteristics of fish head: 1) The head part of fish keeps partially elliptical which can be easily detected via the proposed scale-space DoH method; 2) The robust color and shape of head region during swimming ensure CNN is capable of identifying each individual across frames. For other kinds of animals, if the individuals have one part of their bodies which are robust during movement process, and slight difference of the part exists among different individuals, then the CNN identification based tracking method is capable of tracking them as well. For other kinds of animals, if the individuals have one part of their bodies which are robust during movement process, and slight differences of the part exist among different individuals, then the CNN identification based tracking method is capable of tracking them as well.

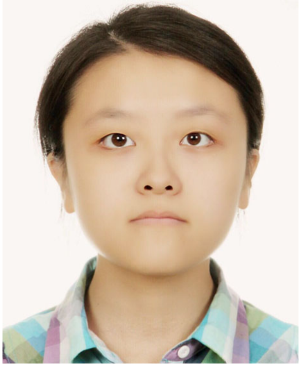
Acknowledgments The authors would like to thank Ye Liu for the valuable discussions and insightful comments.

References

1. Andriluka M, Roth S, Schiele B (2008) People-tracking-by-detection and people-detection-by-tracking. In: IEEE Conference on computer vision and pattern recognition, 2008. CVPR 2008. IEEE, pp 1–8
2. Arulampalam MS, Maskell S, Gordon N, Clapp T (2002) A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. IEEE Trans Signal Process 50(2):174–188
3. Bercla J, Fleuret F, Fua P (2006) Robust people tracking with global trajectory optimization. In: 2006 IEEE Computer society conference on computer vision and pattern recognition, vol 1. IEEE, pp 744–750
4. Bruyndoncx L, Knaepkens G, Meeus W, Bervoets L, Eens M (2002) The evaluation of passive integrated transponder (pit) tags and visible implant elastomer (vie) marks as new marking techniques for the bullhead. J Fish Biol 60(1):260–262
5. Butail S, Paley DA (2012) Three-dimensional reconstruction of the fast-start swimming kinematics of densely schooling fish. J R Soc Interf 9(66):77–88
6. Chen Y, Yang X, Zhong B, Pan S, Chen D, Zhang H (2015) Cnntracker: online discriminative object tracking via deep convolutional neural network. Appl Soft Comput

7. Ciresan D, Giusti A, Gambardella LM, Schmidhuber J (2012) Deep neural networks segment neuronal membranes in electron microscopy images. In: Advances in neural information processing systems, pp 2843–2851
8. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: IEEE Computer society conference on computer vision and pattern recognition, 2005. CVPR 2005, vol 1. IEEE, pp 886–893
9. Delcourt J, Becco C, Ylieff M, Caps H, Vandewalle N, Poncin P (2006) Comparing the ethovision 2.3 system and a new computerized multitracking prototype system to measure the swimming behavior in fry fish. Behav Res Methods 38(4):704–710. doi:10.3758/BF03193904
10. Delcourt J, Ylieff M, Bolliet V, Poncin P, Bardonnet A (2011) Video tracking in the extreme: a new possibility for tracking nocturnal underwater transparent animals with fluorescent elastomer tags. Behav Res Methods 43(2):590–600
11. Delcourt J, Denoël M, Ylieff M, Poncin P (2013) Video multitracking of fish behaviour: a synthesis and future perspectives. Fish Fish 14(2):186–204
12. Fan J, Xu W, Wu Y, Gong Y (2010) Human tracking using convolutional neural networks. IEEE Trans Neural Netw 21(10):1610–1623
13. Fontaine EI (2008) Automated visual tracking for behavioral analysis of biological model organisms. Ph.D. thesis. California Institute of Technology
14. Fontaine E, Lentink D, Kranenbarg S, Müller UK, van Leeuwen JL, Barr AH, Burdick JW (2008) Automated visual tracking for studying the ontogeny of zebrafish swimming. J Exp Biol 211(8):1305–1316
15. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on computer vision and pattern recognition (CVPR). IEEE, pp 580–587
16. Guo Y, Chen Y, Tang F, Li A, Luo W, Liu M (2014) Object tracking using learned feature manifolds. Comput Vis Image Understand 118:128–139
17. Hinton G, Deng L, Yu D, Dahl GE, Mohamed Ar, Jaitly N, Senior A, Vanhoucke V, Nguyen P, Sainath TN et al (2012) Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. Signal Process Mag IEEE 29(6):82–97
18. Jarrett K, Kavukcuoglu K, Ranzato M, LeCun Y (2009) What is the best multi-stage architecture for object recognition? In: 2009 IEEE 12th International conference on computer vision. IEEE, pp 2146–2153
19. Kalal Z, Mikolajczyk K, Matas J (2012) Tracking-learning-detection. IEEE Trans Pattern Anal Mach Intell 34(7):1409–1422
20. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
21. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
22. Li Y, Huang C, Nevatia R (2009) Learning to associate: hybridboosted multi-target tracker for crowded scene. In: IEEE Conference on computer vision and pattern recognition, 2009. CVPR 2009. IEEE, pp 2953–2960
23. Li H, Li Y, Porikli F (2015) Robust online visual tracking with a single convolutional neural network. In: Computer vision–ACCV 2014. Springer, pp 194–209
24. Liu J, Hu H (2010) Biological inspiration: from carangiform fish to multi-joint robotic fish. J Bionic Eng 7(1):35–48. doi:10.1016/S1672-6529(09)60184-0
25. Miller N, Gerlai R (2007) Quantification of shoaling behaviour in zebrafish (danio rerio). Behav Brain Res 184(2):157–166
26. Miller N, Gerlai R (2012) Automated tracking of zebrafish shoals and the analysis of shoaling behavior. In: Zebrafish protocols for neurobehavioral research. Springer, pp 217–230
27. Noldus LP, Spink AJ, Tegelenbosch RA (2001) Ethovision: a versatile video tracking system for automation of behavioral experiments. Behav Res Methods 33(3):398–414
28. Pérez-Escudero A, Vicente-Page J, Hinz R, Arganda S, de Polavieja G (2014) idTracker: tracking individuals in a group by automatic identification of unmarked animals. Nat. Methods 11(7):743–751. doi:10.1038/NMETH.2994
29. Pirsiavash H, Ramanan D, Fowlkes CC (2011) Globally-optimal greedy algorithms for tracking a variable number of objects. In: 2011 IEEE Conference on computer vision and pattern recognition (CVPR). IEEE, pp 1201–1208
30. Qian Z, Cheng X, Chen Y (2014) Automatically detect and track multiple fish swimming in shallow water with frequent occlusion. PLoS ONE 9(9):e106,506. doi:10.1371/journal.pone.0106506
31. Reid DB (1979) An algorithm for tracking multiple targets. IEEE Trans Autom Control 24(6):843–854

32. Rosemberg D, Braga M, Rico E, Loss C, Córdova S, Mussulini B et al (2012) Behavioral effects of taurine pretreatment in zebrafish acutely exposed to ethanol. Neuropharmacology 63(4):613–623
33. Rosenthal SB, Twomey CR, Hartnett AT, Wu HS, Couzin ID (2015) Revealing the hidden networks of interaction in mobile animal groups allows prediction of complex behavioral contagion. Proc Nat Acad Sci 112(15):4690–4695
34. Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, LeCun Y (2013) Overfeat: integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229
35. Szegedy C, Toshev A, Erhan D (2013) Deep neural networks for object detection. In: Advances in neural information processing systems, pp 2553–2561
36. Vedaldi A, Lenc K (2014) Matconvnet-convolutional neural networks for matlab. arXiv preprint arXiv:1412.4564
37. Wang T, Wu DJ, Coates A, Ng AY (2012) End-to-end text recognition with convolutional neural networks. In: 2012 21st International conference on pattern recognition (ICPR). IEEE, pp 3304–3308
38. Yu Q, Medioni G, Cohen I (2007) Multiple target tracking using spatio-temporal markov chain monte carlo data association. In: IEEE Conference on computer vision and pattern recognition, 2007. CVPR'07. IEEE, pp 1–8
39. Zhou X, Xie L, Zhang P (2015) Online object tracking based on cnn with metropolis-hasting re-sampling. In: Proceedings of the ACM international conference on multimedia. ACM, pp 1–4



Shuo Hong Wang received her BEng degree from East China University of Science and Technology in 2012. She is now pursuing her PhD degree in School of Computer Science, Fudan University. Her research interests include multiple object tracking, biological and medical image analysis.

Jing Wen Zhao received her BEng degree from East China University of Science and Technology in 2013. She is now pursuing her PhD degree in School of Computer Science, Fudan University. Her current research interests include biological image processing, medical image analysis and pattern recognition.

Yan Qiu Chen received his PhD degree from Southampton University, United Kingdom in 1995, and his MEng and BEng degrees from Tongji University, Shanghai, China in 1988 and 1985 respectively. Dr. Chen is currently a full professor and director of Computer Vision Lab with School of Computer Science of Fudan University, Shanghai, China,. He had been Chairman of Department of Communication Science and Engineering from 2004 through 2007, and Associate Chairman of Department of Computer Science and Engineering from 2002 through 2004. Dr. Chen was an assistant professor with School of Electrical and Electronic Engineering of Nanyang Technological University, Singapore from 1996 through 2001, and was a postdoctoral research fellow with Glamorgan University, UK in 1995.