Neurocomputing 551 (2023) 126484

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Ice hockey puck tracking through broadcast video

Muyu Li [a,b,*], Henan Hu [b,c,d], Hong Yan [a,b]

[a] Centre for Intelligent Multidimensional Data Analysis Limited, Hong Kong, China
[b] Department of Electrical Engineering, City University of Hong Kong, Hong Kong, China
[c] Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, Jilin, China
[d] University of Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

Article history:
Received 22 February 2022
Accepted 21 June 2023
Available online 24 June 2023
Communicated by Zidong Wang

Keywords:
Ice hockey puck
Detection
Tracking
Correlation filter

ABSTRACT

Ice-hockey puck tracking is a non-trivial task in hockey video analysis as it highlights the puck in the video for broadcasting, tactical play analysis, or referee assisting. However, difficulties, such as high speed, low texture features in images and constantly changing shape, make well-developed object tracker fail to track the puck. This paper introduces a real-time online-learning ice hockey puck detection and tracking system solely depending on video input to tackle this problem. The proposed approach categorizes pucks into free-moving and control-moving states, using a combination of contour fitting, correlation filter, and motion estimation techniques to detect and track them. A thorough analysis is performed focusing on the tracking scenario using broadcast video. To our knowledge, this is the first approach addressing detection and tracking nearly invisible high-speed pucks when shooting actions take place. Experiments with a comparison between a previous work targeting puck tracking show promising results in detection and tracking the ice hockey puck through broadcast video.

1. Introduction

1.1. Motivation

Video analysis systems have spawned much research in the last several decades, and with the emergence of highly developed object detection and tracking algorithms, real-time video analysis system has become possible for many applications. For ball games, the focus point is mainly on the location of the ball, since players on the court, the audience in the stadium, and the cameraman for broadcasting concentrate on it.

Automatic ball detection and tracking system have many commercial applications. From the on-court referees' point of view, this technique is crucial to determine violations or a goal. Broadcasting companies benefit by using such technology to automatically select the gaze region to follow an ongoing play during a live stream. Both tasks are currently performed by humans (referees or cameramen), thus automating the process could significantly avoid manual error and increase accuracy.

Motivated by the above, this paper is committed to developing an approach to automatically track the puck from a single broadcast video. The approach will be able to detect and track the puck regardless of the condition of the puck and retrieve the puck after severe occlusion by players or logos. The proposed approach has two advantages compared to existing techniques: 1) The approach is able to track ice hockey pucks with low texture features, which the existing method, popular single target trackers, fails to do. Prove that it's possible to incorporate well-developed trackers in ice hockey game analysis. 2) The approach firstly addresses a previously avoided problem of tracking nearly invisible pucks after shooting actions in existing methods to our best knowledge.

1.2. Challenges

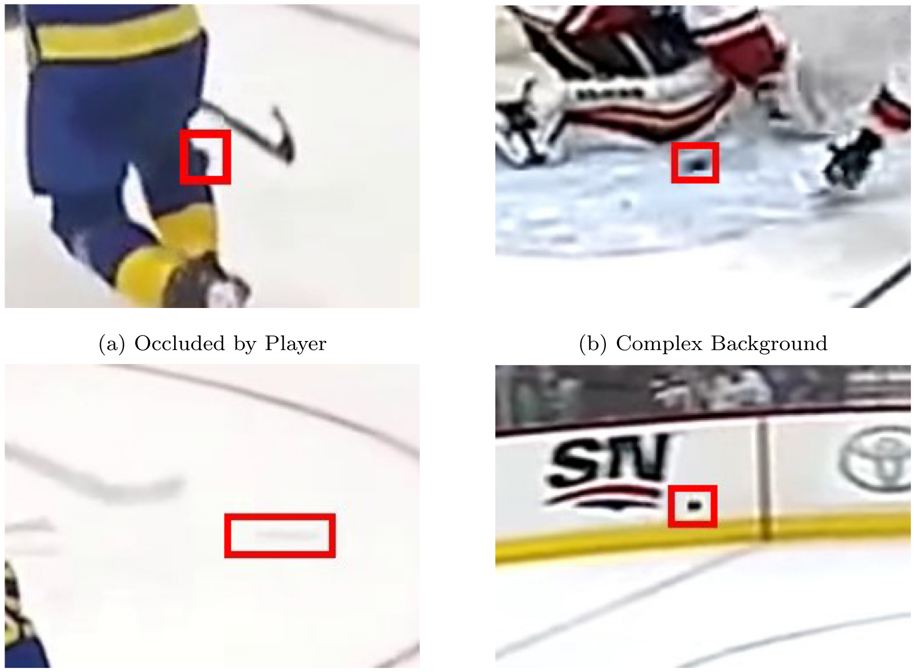
Lots of previous work addressed detecting and tracking balls, yet most of them target soccer balls [1,2], basketballs [3,4], table tennis balls [5], tennis balls [6], etc. Only very few works aim to track the ice hockey puck [7] due to the following difficulties as shown in Fig. 1: (1) Pucks moving at high speed, (2) Lots of similar interference in the rink, (3) Rapidly changing size and shape according to the frame rate of the camera, (4) Low-resolution images, (5) Constantly occluded by players or logos, (6) Illumination changes due to the camera motion.

As for the tracking task itself, the great performance of state-of-the-art trackers depends on rich features extracted by handcraft techniques or even deep convolution neural networks. However,

* Corresponding author at: Centre for Intelligent Multidimensional Data Analysis Limited, Hong Kong, China.

E-mail addresses: muyu@innocimda.com (Muyu Li), huhenan18@mails.ucas.ac.cn (Henan Hu), h.yan@cityu.edu.hk (Hong Yan).

https://doi.org/10.1016/j.neucom.2023.126484
0925-2312/



(a) Occluded by Player (b) Complex Background

(c) High Speed Color and Shape Shifting (d) Small Resolution Compare to Logos

Fig. 1. Hard Cases of Puck Tracking.

when dealing with cases like tracking ice hockey pucks, the extreme lack of appearance features almost determines the failure of state-of-the-art trackers. Thus, how to successfully track ice hockey pucks in broadcast videos still remains a big challenge despite the rapid development of single object tracking.

1.3. Previous Work

Most of the approaches about detecting and tracking the ball can be classified into three aspects: feature-based, model-based and motion-based. Feature-based approaches take advantage of color histogram, geometry, HoG (Histogram of Oriented Gradient) [8], or other handcraft features [9–11]. Model-based approaches use semantic representations or domain knowledge [12–14], while motion-based approaches incorporate the trajectory of balls to discriminate them from other interference or background[1,5,15,16]. All of the approaches mentioned above try to highlight the ball from the rest of the images. Different sports require different sets of techniques or even a combination of several methods.

As far as ice hockey is concerned, the most common complaint by referees on the court and the audience behind TVs is pucks not being followed accurately. In 1997, the first attempt [17] was proposed to highlight pucks for better visualization by designing an electronic puck with infrared sensors placed inside along with an infrared detection system. The approach was not further in use because of complaints by players about the worse feel of an electronic puck than a regular puck, and the complex requirement of equipping the rink.

Different from methods by implementing sensors inside the puck or around the rink, [7] proposed the first image processing method to detect and track ice-hockey pucks. However, the puck, in their literature, is considered in a zoomed-in broadcast video, in which the puck subtended around 200 pixels and 30–50 times smaller in the regular broadcast viewpoint. The proposed method implemented a background subtraction algorithm and a velocity estimation technique to associate detections across frames. However, they only addressed cases when the puck is slow and evident, and discarded circumstances in which the puck is hazy and almost invisible after a shooting action.

1.4. Contributions

This paper introduces a real-time approach solely based on image processing techniques to detect and track ice hockey pucks in broadcast video, incorporating a combination of contour fitting, correlation filter and motion estimation method. The contribution of this paper is twofold. First, this paper presents a solution for tracking low-texture ice hockey pucks using the state-of-the-art online learning correlation filter tracker, combined with weighted constraints by shape similarity and re-identification. Second, this paper presents a template-based online learning re-identification phase for tracking nearly invisible high-speed pucks after shooting actions. To our best knowledge, this is the first approach to address this kind of extreme cases. Thorough experiments on the ice hockey game scenarios and our proposed method show promising results in detecting and tracking ice hockey pucks through broadcast live streams.

2. Related Work

Ball Tracking. Some previous works have addressed the detection and tracking of the ball. However, most of them focus on sports with relatively large ones, e.g. [10,5,1,2,6,3,18]. Reasoning from plays on the court, [19,20,4] improved tracking performance by combining ball detections with player locations and ball possession data. In [21], an approach is proposed to increase the reliability of tracking by introducing physical constraints. Later on, many solutions [21,22] use multiple calibrated cameras to track positions of the ball in 3D space, in order to bypass the problems caused by occlusions from players.

Ice Hockey Puck Tracking. For ice hockey, tracking pucks with high speed (can reach a speed of 160 km/h after being hit by a player) and small size through a broadcast stream is quite challenging. Furthermore, some common difficulties in tracking methods, such as occlusion, illumination variation, and camera motion, may further impede puck tracking. In recent years, deep learning has been introduced into the task of detecting and tracking pucks. The work [7] proposed a method that addressed puck detection and tracking in a zoomed-in broadcast video. The puck occupies as much as 150–250 pixels, which is commonly 20 times in a regular live stream match. The method uses background subtraction to separate the puck and a velocity estimation method to link detections from consecutive frames, but performs well only for short intervals. In work [23], authors propose to recognize tactics of both teams in an ice hockey game addressing puck tracking and use the deep learning-based regression method to locate the puck. They focus on broadcast videos with wide-range capture devices and locating the puck with cascade stages containing players detection, optical flow, and a puck regression. The model is relatively complex and thus runs inference at around 6 fps which is hard to implement in a broadcast manner. Further in the work [24], a PuckNet is proposed to estimate the location of the puck by leveraging hockey event annotations and corresponding broadcast video. A heatmap of the corresponding puck in several consecutive frames is generated using purely Convolutional neural networks architecture. However, the PuckNet adopts CNN to directly capture global semantic, texture, and temporal features, which is not efficient. Also, it can be implemented in an offline manner due to the need for hockey event annotations. Later in [25], a temporal modelling video branch along with a player detection branch is introduced to enhance the attention of the puck based on the locations of players. The network runs at a frame rate of 5 fps and is trained end-to-end with multi-task learning where also play-by-play event annotations are embedded to avoid the influence of occlusion. Another work in [26] introduces modelling spatial and temporal information together with a 3D-CNN architecture. The cost of memory and time is extremely high, with a run time of 2.7 s per frame as reported in their work. Also, the temporal processing stage needs additional four frames ahead which means that real-time implementation is not possible even with the development of computational power. Deep-learning based methods are trained on a specific dataset with a certain pattern of objects and usually fail when facing the problem of motion blur. However, motion blur is the main theme of puck tracking during a game since the speed of the puck is relatively high.

Datasets. Dataset of Ice Hockey Puck tracking is a big issue due to the requirements of a large amount of frame-by-frame puck annotations. Most recent works [24–26,23] generate their own datasets with several sequences of ice hockey videos and evaluate the performance. The only mentioned public available dataset of puck tracking is claimed in the work of [26], but it is still unable to reach since no public link is provided. From our point of view, real-time ice hockey puck tracking is essential for broadcasting, on-court referee assisting, and also on-the-spot coaching. Deep learning methods yield great success in image processing with powerful feature extraction ability but fail to perform well in real-time puck tracking. The reasons are obvious in the following aspects: (1) the puck, the only object that matters in puck tracking, is extremely small in the video frame and thus contains weak context features; (2) motion blur, one of the most influential interference, is beyond solved by deep-learning models since they usually rely on the image features and trained with the inherent pattern of specific datasets; (3) the interference in the field of view usually occupies a large portion of the video and confuse the deep networks; (4) severely lacking datasets with puck annotations leads to inadequate training for all the deep learning models; (5) inference with deep-learning based models are time-consuming and impossible in a real-time fashion. Based on the above discussions, we proposed to tackle the motion blur and the real-time puck tracking problem in a relatively traditional way with a hand-craft localization strategy combined fast tracker and carefully fusion of structure based feature selection. A slightly worse performance of the fast tracker is tolerated in our method because of the remedy ability of the re-identification strategy and thus leads to an efficient performance with the speed-accuracy trade-off.

3. Methodology

3.1. Prerequisite

Ice hockey puck in a broadcast video is constantly shape-shifting. As a prerequisite, we classify the puck's moving state into two categories: controlled moving and free moving state. Controlled moving (CM) state means the puck is being controlled by a player and usually has a similar moving pattern with the player. The trajectory under CM state is orderless and difficult to predict from the previous locations. Another problem is when the puck is controlled by a player, it may suffer from occlusion by hockey sticks every now and then, which shares a similar color, and sometimes shape. As for free moving (FM) state, it is usually formed after a shooting action by a player. If the frame rate is relatively high, the puck is untouched and follows a constant velocity strategy during the FM state. Players or referees on the court are not in the surroundings for most of the cases, thus color segmentation or edge detection is able to separate the puck from the ice rink background, which possesses an obvious contrast of color. However, under FM state, the color of the puck changes paler and the shape lengthens due to the motion blur.

3.2. Puck Tracking

The overview of our method is shown in Fig. 2. The proposed puck detection and tracking approach targets the two states mentioned above and is able to deal with problems like shape-changing and occlusion. The whole process consists of two phases, a plain tracking phase to track the puck at a slow speed without heavy interference and a re-identification phase to target extreme circumstances corresponding to CM and FM state. Considering the color of official pucks is black and the ice surface is mainly white, we adopt CN (color names) feature [27] combined with HoG [8] as our main feature map for the plain tracking phase. With the constant change of the shape, the structure feature is brought into the picture as well.

3.2.1. Plain Tracking

Tracking the puck at a slow speed without heavy environmental interference can be viewed as a simple single object tracking problem. Considering the need for real-time processing on video and the lack of appearance features of pucks, we select the correlation filter based approach as our baseline tracker due to its characteristics of online training and rapid calculation. From the first frame of the video, usually at the start of the game, the puck is located at the center of the image. Based on this assumption, the first image frame is thresholded by an initial gray-level threshold α_{init}, which is the normalized threshold value of 0.55. α_{init} is a statistical value acquired from an analysis of the first frames in a bunch of public available online ice hockey match videos. Using a threshold of α_{init}, the separation between the puck and the rink is robust even with the different environmental settings, such as logos on the ice or different illumination in the rink. Then the correlation filter is initialized using the carefully cropped image patch containing the puck. The plain tracking phase is then followed by a regular correlation filter tracker procedure as ECO tracker [28]. Noticed

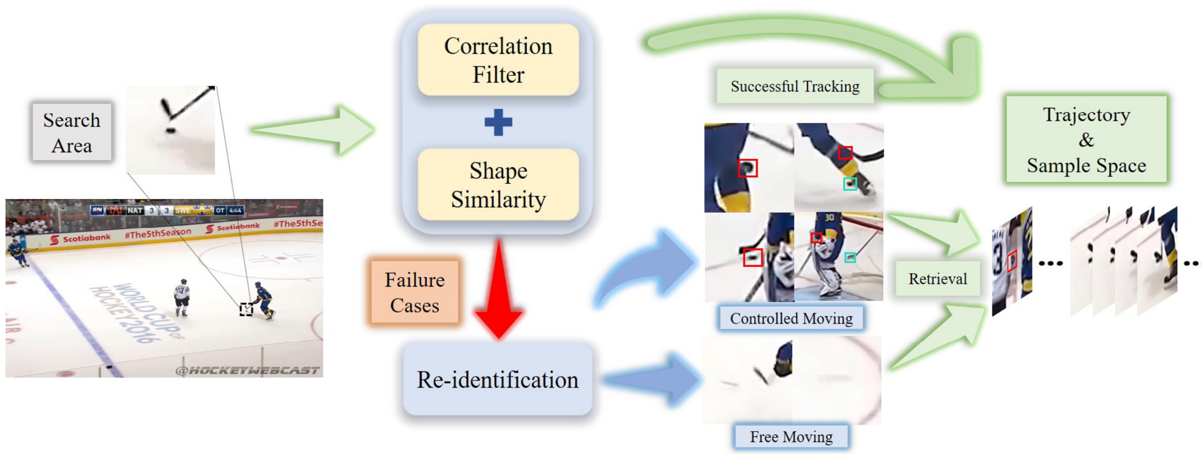


Fig. 2. Illustration of the proposed approach: For each time stamp, a cropped search area suggested from the previous location is fed into the tracker. A Failure Score (FS) combined with scores from the correlation filter and structural similarity is measured to determine whether tracking is failed in the search area. Later a re-identification phase is introduced to retrieve the puck from both controlled moving state and free moving state.

that the tracker is changeable and we will study the influence of the different choices of trackers. The tracker discriminatively learns a convolution filter based on a collection of K training samples $\{x_k\}_1^K \subset \chi$ from previous tracking results, and each feature layer $f_k^m \in \mathbb{R}^{Res_m}$ has an independent resolution Res_m^1. Specifically, in our case, combined HoG (Histogram of Oriented Gradient) [8] and CN (Color Names) feature maps [27] yield little difference in performance compared with deep features extracted by CNN [29], but operating on a much higher speed. This is because the puck in an image frame does not hold rich texture or semantic information, allowing color and edge features of the puck to play the most important role in the feature space.

The learning process is conducted as the following steps, and perform an update of the puck's location. First, the feature map is transferred into the continuous spatial domain $t \in [0, T\}$*unknown*2023 by an interpolation model, with the operator J_m,2162023

$$J_m\{x^m\}(t) = \sum_{res=0}^{Res_m-1} x^m[r] b_m(t - \frac{T}{Res_m} r) \tag{1}$$

Here b_a is an interpolation kernel with a period of $T > 0, R$ denotes the independent resolution of feature layer f_k Then, the entire interpolated feature map $J\{f\}$ is formed by combining all the interpolated feature layer $J_a\{f^a\}$. A factorized convolution operator is introduced to predict the detection scores S_{det} of the puck as:

$$\begin{aligned} S_{det,P,f}\{f\} &= Pf * J\{x\} \\ &= \sum_{n,m} p_{m,n} f^n * J_m\{x^m\} \\ &= f * P^{\mathrm{T}} J\{x\} \end{aligned} \tag{2}$$

The scores show the confidence of the puck's location in each corresponding image region of the feature map $x \in \chi$. Where P is a $M \times N$ matrix which represents the coefficient space. f^n is a smaller set of basis filters $(f^1, \ldots, f^M)$ instead of learning one separate filter for each feature channel m. f is constructed as a linear combination of the filter f^n by a set of learned coefficients $p_{m,n}$. This process can be viewed as a lower dimensional method that leads to a radical reduction of parameters. The filters are learned by minimizing the L2-norm in the Fourier domain to form a more tractable optimization problem as follows,

$$E(f, P) = \|\hat{z}^{\mathrm{T}} P\hat{f} - \hat{y}\|_{l^2}^2 + \sum_{n=1}^{N} \|\hat{w} * \hat{f}_n\|_{l^2}^2 + \lambda \|P\|_F^2 \tag{3}$$

Where $\hat{y}_j$ is the Fourier coefficients of labelled detection scores of samples x_k, which is originally set to a periodically repeated Gaussian function. $\hat{z^m} = X^m \hat{b_m}$ is used to simplify notation as the Fourier coefficients of the interpolated feature map $z = J\{x\}$. The regularization integrates a spatial penalty to mitigate the drawbacks of the periodic assumption, while enabling extended spatial support [30]. The loss is a non-linear least squares problem, thus [28] employ Gauss–Newton [31] and use the Conjugate Gradient method to optimize it and complete the learning process of tracking. Given the above process, the tracker is able to perform an update stage in a search area with 1.5 times the size of the original image patch. With no catastrophic interference such as occlusion or out-of-view problems, the plain tracking phase maintains an acceptable robust performance.

Traditional correlation filter trackers like KCF [32], DCF [33], C-COT [34], or ECO [28] focus on single object tracking where the tracking scenario has an obvious distinction between foreground and background, and are only able to retrieve objects after short-term occlusion or even lost the tracking record. Under this premise, trackers tend to fail when a similar object is nearby or occlusion occurs. This problem is particularly fatal in our task due to the low-texture of the puck in the image. Other objects in the surroundings, such as the end of sticks waved by players, may share a similar color and contour, which is possible to confuse the tracker and cause failure or drift. In order to tackle this inconvenience, a modified update strategy is proposed. Instead of updating on the region with the highest score, another shape similarity score on binary images is introduced to balance the influence of other similar objects. An image patch from the first frame is served as a template to calculate shape similarity between the location predicted by a correlation filter and itself. The shape similarity score is measured by Hu-Moment Invariants [35], which are a set of 7 numbers calculated by central moments that are invariant to image transformations. Central moments using in Hu-Moment invariants are calculated as follows:

$$\eta_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I(x, y) \tag{4}$$

Where centroid $(\bar{x}, \bar{y})$ can be acquired by $\bar{x} = \frac{M_{10}}{M_{00}}, \bar{y} = \frac{M_{01}}{M_{00}}$, and M_{00}, M_{01}, M_{10} are moments calculated by formula $M_{ij} = \sum_x \sum_y x^i y^i I(x, y)$. Then we measure 7 Hu-Moment Invariants using centroid moments. In this paper, we adopted the first 6 moments to compare the shape similarity, since they have been proven to be invariant to translation, scale, rotation, and reflection. By measuring 6 Hu-Moment Invariants, the similarity score S_{ss} is calculated using L1-distance as follows:

$$S_{ss} = \sum_{i=0}^{6} \left| H_i^{template} - H_i^{CF} \right| \tag{5}$$

After getting the shape similarity score between the template and the image patch given by the correlation filter, a failure score (FS) combined with weighted correlation filter and shape similarity is introduced as follows:

$$FS(s_{cf}, s_{ss}) = \frac{1}{1 + e^{(s_{cf} - \delta_{cf})\gamma_{cf}}} \times \frac{1}{1 + e^{-(s_{ss} - \delta_{ss})\gamma_{ss}}} \tag{6}$$

Where FS is a weighted constraint combined with the shape similarity score and the max confident score of the correlation filter result. A transformation of the sigmoid function is used to smooth the threshold of each score and filter the score further away from the threshold. $\delta_{cf} = 0.2, \delta_{ss} = 0.4$ denote the empirical parameter to determine whether the result from the correlation filter or shape similarity measure is valid or not, respectively. To deal with extreme cases when either correlation filter or shape similarity measure returns confident feedback while the other measurement disagrees, a hard gap of $\delta_{cf} < 0.1, \delta_{ss} > 0.7$ are set for both thresholds. γ_{cf}, γ_{ss} are the amplifying factor to weigh the influence of correlation filter and shape similarity respectively. The higher the FS means the tracking process is likely to fail. Any result with a score above 0.8 indicates that the object is either being occluded or hit by a player with a changed shape. For the puck under this circumstance, the correlation filter stops updating or adding patches to the memory, saving the previous samples and handing them over to the re-identification phase for further processing.

3.2.2. Re-identification (Re-ID)

Directly using popular single object trackers like correlation filter tracker may easily fail and lead to problems like drift or loss. This kind of failure tends to pop up regularly even after being weighted by a combined constraint of shape similarity. When the correlation filter can not find a reliable result of the puck in a certain frame, a re-identification phase is proposed to handle the puck retrieving task, under both CM state and FM state. The most decisive factor in determining the state of the puck is speed. The lower the speed means the shape and color are more likely unchanged compared to a regular puck. On the other hand, the puck will become hazy like a shadow, which is hard to locate by using the image processing method conversely. Different methods regarding different states are discussed in the following sections.

For most cases, puck tracking under CM state can be done through the plain tracking phase only, since the moving speed is related to the speed of the controlling player and obviously not fast enough to cause motion blur under frame rate in a real-time video. Most of the failure cases are caused by the occlusion of other players, which usually lasts for several frames resulting in disappearance and reappearance at a location far away from the predicted candidate region indicated by the previous trajectory. If the reappearing location lies outside of the correlation filter's search region (1.5 times larger than the bounding box of the puck), the tracker tends to drift to another similar object and fails to retrieve the puck ever again.

As mentioned above, the puck's moving speed under CM state is rather slow, which means that the shape of the puck presented in the image will not change significantly, but only rotate or change its scale. In view of this assumption, to retrieve the puck after constant occlusion, a re-identification approach, as shown in Fig. 3, composed by an ellipse detection technique and a shape comparison method is proposed to find puck candidates and select the right one in an adaptive enlarged search area controlled by the number of the puck's lost frames. First, following the previous locations of the puck, a connected component analysis is conducted within the search area and finds contours that share a similar shape and scale with our puck. Then an ellipse fitting approach [36], shown in Fig. 3, is adopted to find candidates with the shape of the ellipse. For the next step of the re-id phase, we carefully select the best match between candidates and the previous sample of the puck using shape similarity calculated by Hu moment invariants, which is the same as we perform in the plain tracking phase. Also, to tackle the problem of similar wrong candidates such as jersey numbers or letters on the billboard, as shown in Fig. 4, a simple player or background segmentation is proposed. Given a search area in the gray-level domain, we first binarize the area using a normalized threshold of 0.5, then perform a closing operation to obtain the segmentation. We finally determine if the candidate lies inside the segment by ray casting algorithm [37], and use I_{in} to denote whether the candidate is within the segment or not. The process is shown in Fig. 5 The overall CM Re-ID score S_{CMreid} is calculated as follows:

$$\begin{aligned} &S_{CMreid} = \tfrac{1}{1+e^{(s_{ss}-\delta_{ss})\gamma_{ss}}} \times I_{in} \\ &where \quad I_{in} = \begin{cases} 0 & if\ candidate\ is\ inside \\ 1 & if\ candidate\ is\ outside \end{cases} \end{aligned} \tag{7}$$

If the best match candidate lies outside of any players or billboard and shares a similar shape of the template within a threshold of $\delta_{ss} = 0.4$, the re-identification score is near 1. Then the candidate is attached to the previous tracklet of the puck and fed to the tracker for continuous tracking.

As for FM state, the puck slides on the ice surface freely at a relatively high speed. The geometric shape of the puck is always displayed like a long stick on the image frame as shown in Fig. 6. Under this circumstance, the puck is extremely indistinguishable on gray-level from the ice surface. After conducting a statistical analysis on this particular circumstance, the best threshold to segment the puck from the background is 0.95. However, this unreliable threshold, to some extent, causes constant confusion between the puck and the shadow of the player. To extract the puck with hazy color and gray-level, we first detect the edge in the search area, then find the puck candidates using ellipse detection. Since the candidate is easily influenced by shadows, shape similarity is not able to provide a solid result. We then calculate the SSIM (Structural Similarity Index Measure) score between the candidate and a pre-cropped stick-like sample. Another obvious clue is that the direction of the stick-like candidate shares a similar angle with the moving direction between the last known location of the puck and the candidate. Hence, calculating the angle similarity from both directions may eliminate wrong candidates easily. Considering both the SSIM score and the angle distance, we have the joint score of re-identification for FM state as follows,

$$S_{FMreid} = \frac{1}{1 + e^{-(s_{SSIM} - \delta_{SSIM})\gamma_{SSIM}}} \times \frac{1}{1 + e^{(D_{angle} - \delta_{angle})\gamma_{angle}}} \tag{8}$$

where SSIM score is calculated as follows,

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\alpha_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\alpha_x^2 + \alpha_y^2 + c_2)} \tag{9}$$

Where μ, α is the average and the variance of each image patch respectively. $c_1 = (k_1L)^2, c_2 = (k_2L)^2$ are variables to stabilize the division with weak denominator, and $k_1 = 0.01, k_2 = 0.03$ by default.

And angle distance is obtained by the formula,

$$D_{angle} = \left| \frac{A_{Candidate} - A_{Direction}}{A_{Direction}} \right| \tag{10}$$

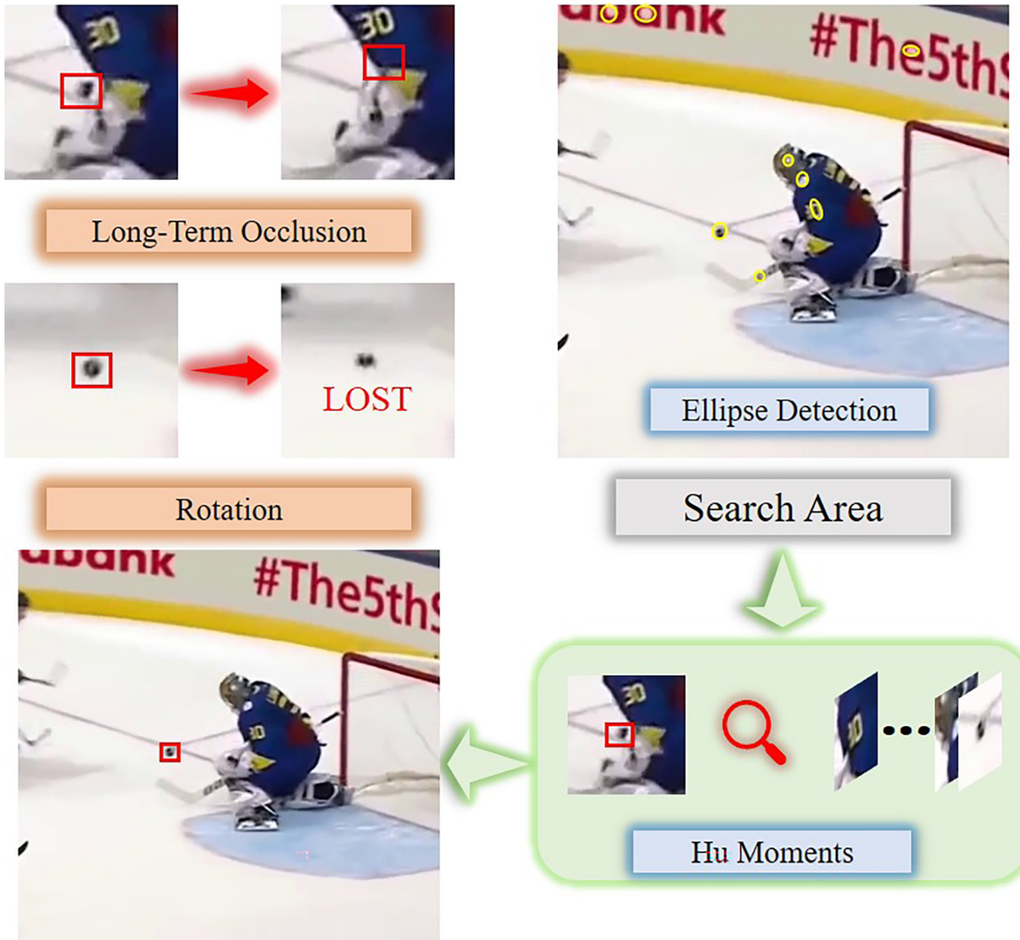


Fig. 3. Re-identification phase dealing with controlled moving state. To tackle the problem of occlusion and some difficult cases of rotation, we first adopt ellipse detection to find objects with the shape similar to the puck. Then calculate Hu Moments between previously confirmed puck and the detected objects, and select the best match as the final retrieval result.

When the candidate itself and the moving pattern share similar angles and the structure is alike with the template, the re-identification score S_{FMreid} will be near 1, then we reclaim the stick-like candidate as the puck and continue tracking.

However, under the extreme circumstance of high speed, logos on the ground could be a serious interference. Considering the moving pattern of the puck is predictable and the trajectory under this moving state can be seen as linear, we incorporate a Kalman estimator [38] to help predict the location of the moving puck. It is proven that if the puck is submerged in the shadow of a player, the puck is likely to be controlled by the very player within a short period of time. Hence, after 5–10 frames, for the puck to be stable and return to its natural geometric shape, we applied the correlation filter to find the puck with the previously learnt feature in the enlarged search area provided by the Kalman estimator. And finally complete the trajectory using the linear interpolation method to finish the tracking procedure under FM state. The whole re-identification phase on FM state is shown in Fig. 6.

4. Experiments and Discussions

To our knowledge, there are no public accessible datasets for pucks detection or tracking. Thus, we select 2 integral videos from ice-hockey game live streams, and carefully crop them into relatively short video clips with puck visible in the image frame. We then manually labelled the puck location for evaluation. Detailed settings about data and evaluation metrics are elaborated in the following sections. Also, the key processes of our proposed method are listed in Algorithm 1.

Algorithm 1 Framework of the proposed method.

```
Output: [F_1, F_2, ··· F_n], 0 <= n <= N   Consecutive frames of
   ice hockey broadcast video contains N frames.
Input: T   Trajectory of the puck.
Initialization
while n <= N do
   s_cf = Tracker(F_n), s_ss = ShapeSimilarity(F_n)
   if FS(s_cf, s_ss) < 0.8 then
      Attached to trajectory T.
   else if FS(s_cf, s_ss) > 0.8 then
      State = (CM, FM)   Separate current state to CM or FM
   by speed.
      if State = CM then
         Calculate S_CMreid
            if S_CMreid > threshold_CM then
               Attached to trajectory T.
            end if
         else if State = FM then
            Calculate S_FMreid = f(s_SSIM, D_angle)
            Attached to trajectory T.
```

(*continued*)

Algorithm 1 Framework of the proposed method.

```
      if S_FMreid > threshold_FM then
        Attached to trajectory T.
      end if
    end if
  end if
end while
```

4.1. Data and Settings

The two selected videos contain 17163 frames, and 12104 of them are images with pucks under either CM or FM state. We then clip the two sequences into 20 small clips and categorize them into 3 groups from easy to hard, based on the level of difficulty. Clips under CM state are categorized on the basis of occlusion rate, while others under FM state are separated in accordance with the moving speed. Combined two different states, the two test sequences are divided into 6 groups of video clips with gradually changing difficulties. The overall number of ground truth pucks, frames without pucks in both CM and FM state are shown in Table 1.

As for the evaluation of the performance, since the bounding box of a puck is rather small and the overlap between the detected and the groundtruth is usually near 100%, thus the widely used single object tracking metrics like *Average Pixel Error (APE)* and *Average Overlap Rate (AOR)* [39] is not fair for puck tracking task. Hence, we adopt the traditional *precision*, *recall* and *F1 score* as our evaluation metrics since the above metrics are capable of reflecting the ability to find and maintain the trajectory of a moving puck. Specifically, *TP* means the correct tracking of the pucks, *FP* contains both mis-detected ones when the correct annotation should be no puck, and inaccurate detections when there actually is a puck in the frame, *FN* means the missing detection of the puck, and *TN* means the correct ones where no puck lies in the field of view.

4.2. Run Time

Our proposed approach currently runs real-time at over 42fps on a 3.60 GHz i7-7700 CPU, which proves the possibility of integrating other fast segmentation or detection methods and remaining real-time.

4.3. Gray-Level Thresholding

Gray-level thresholding plays a big role in both our two phases, and serves as a pre-processing process. To find the best gray-level thresholds for both phases, we calculate the number of detected pucks out of 12104 labelled puck groundtruths in our 2 integral ice-hockey game videos respectively, 9288 out of 12104 pucks are under CM state and 2726 are under FM state. All the frames with the puck under CM and FM state are cropped as our test data. Then the data is separated into two subsets based on the current

(a) Jersey number (b) Letter on the billboard

(c) Ice skates of a player (d) Glove of the goal keeper

Fig. 4. Similar objects in the search area, such as jerseys or letter on the billboard.

Fig. 5. Example of the simple segmentation process to determine if the candidate lies inside a player. Two candidates, one is the jersey number and the other is the ice skate, are marked in red. The closing operation is used to find the segment of the player, later using the ray casing algorithm we can determine if both candidates lie inside the player and save the results for further processing.

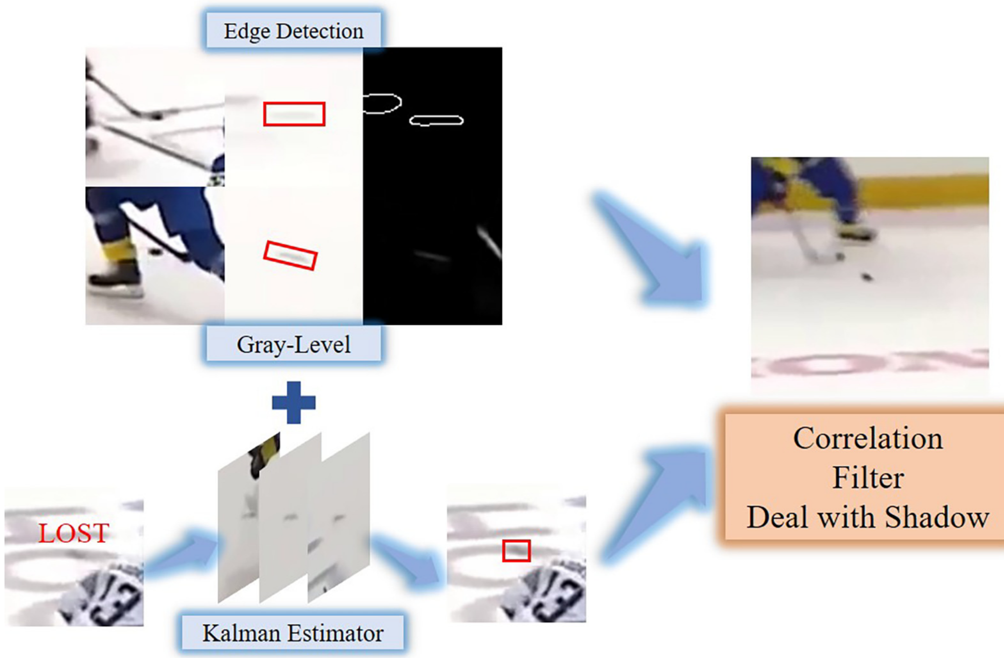


Fig. 6. Re-identification phase dealing with free moving state. To deal with extreme circumstances of fast-moving puck, we use edge detection with a carefully selected gray-level threshold of 0.95 to find the hazy puck. As for logo-like and shadow interference, a Kalman estimator and the existing correlation filter tracker are introduced to find the reasonable trajectory and the puck, respectively.

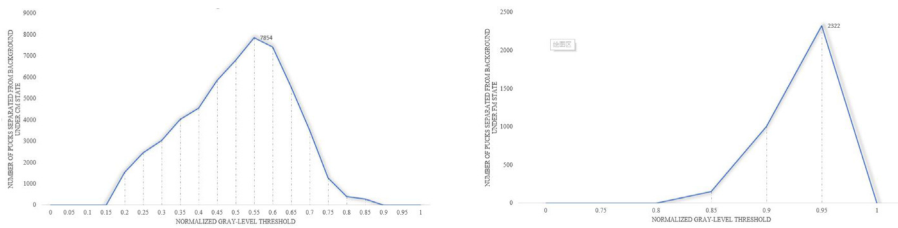
Table 1
Numbers of pucks under different states in the dataset.

State of Data	Frames w/ Pucks	Frames w/o Pucks	Total Frame Number
All	12104	5059	17163
CM	9288	4805	14093
FM	2726	254	2980

state of the puck. As shown in Fig. 7a, the normalized threshold value is varied from 0 to 1, and the number of pucks successfully separated from the background is shown on the vertical axis. From Fig. 7a, a threshold value between 0.5 and 0.6 yields the best result for CM state, the fluctuation in this region indicates the influence of illumination variance. The peak value, 0.55, according to Fig. 7a, which can separate 7854 of 9288 pucks from the background, is selected to find the puck when the game start for tracking initialization in the first frame. As for the FM state, gray-level thresholding determines the possibility of locating the puck at high speed. Due to the high moving speed, the puck is vague and the gray-level is very close to the ice surface. However, according to Fig. 7b, with a threshold of 0.95, 2322 of total 2726 pucks can be separated from the background, and that's a promising preliminary result for the so-called 'invisible' puck. With a threshold of 0.95, incorporate with Kalman estimator and our correlation filter tracker, the detected pucks can serve as semi-supervised labels and provide cues for estimating locations to form a reasonable trajectory. Detailed performance of puck tracking under extreme FM state is demonstrated in the following study.

4.4. Overall Performance

We first compared our method with several previous methods and show the result in Table 2. Then to better exhibit the performance of our two phases, an ablation study of our proposed approach is conducted. We first test the plain tracking phase on our data and analyse the result. Then by adding different parts of re-identification phase, we show the effectiveness of our approach in dealing with hard cases under both CM and FM state. Also a comparison between our proposed approach under CM state and



(a) Successfully separated pucks under CM state. A threshold of 0.55 yields the best result of 7854 out of 9288.

(b) Successfully separated pucks under FM state.A threshold of 0.95 yields the best result of 2322 out of 2726.

Fig. 7. Number of pucks separated from background using different normalized gray-level threshold under both CM and FM state.

Table 2
Comparison with previous works. Best in bold and second best underlined.

Method	GT	TP	FP	FN	TN	Pcs.	Rec.	F1
[24]	9378	6547	2719	2521	2406	70.6%	72.2%	71.4%
[26]	9378	5473	3804	2991	1879	59.0%	64.7%	61.7%
[23]	9378	6583	2437	2399	2764	72.9%	73.3%	73.1%
[7]	9378	4367	4279	3976	1562	50.5%	52.3%	51.4%
[25]	9378	<u>8053</u>	**920**	<u>1223</u>	**3987**	**89.7%**	<u>86.8%</u>	<u>88.2%</u>
Ours(w/o FM)	9288	**8437**	<u>1463</u>	**651**	<u>3542</u>	<u>85.2%</u>	**92.8%**	**88.9%**
Ours(w/ FM)	12104	8599	1618	1515	3741	84.2%	85.0%	84.6%

other methods in puck detection and tracking is performed to show the improvements. Finally, we conduct a statistical experiment on tracking pucks under free moving state, which has never been addressed in any of the previously published work.

4.4.1. Compared with Previous Works

We follow the experiment conducted in [7] to demonstrate our improvements. Table 1 shows the tracking outcome using solely plain tracking phase compared with adding Re-identification phase under either CM state or FM state. Comparison cases are targeting either CM or FM state cases, which contain 9288 and 2726 pucks out of a total number of 12104 groundtruths in 17163 frames, respectively. Plain tracking phase, which only contains the correlation filter tracker, may easily shift to other similar objects like jerseys or sticks due to the extremely low texture of the puck, and cause high *false positives* and low *precision*. Also adding to *false negative* cases, the plain tracking phase possesses no ability to track the puck after a shooting action, as in FM state cases. Compared with the plain tracking phase, most of the failure cases caused by similar objects nearby like jersey numbers or sticks can be addressed by adding our re-identification phase under CM state. And thus significantly lower *FP*, *FN* and increase all the metrics. Re-ID phase can opportunely suspend the shift of the object caused by similar disturbances nearby and retrieve the puck after its re-emerging in the scene. As for FM re-identification, cases are more complicated and combined with multiple cues that may influence the results such as speed, motion of cameras, illumination variants and so on. To our best knowledge, there is no previous investigation on detection or tracking pucks after shooting actions. Thus we conduct the experiment in the same way as before, out of 2726 samples, our approach is able to tackle some of the cases and produces a *F1 score* of 77.6%. With the extreme circumstances of FM state, similar objects that may cause the rise of *FP* are limited. Thus our experiment shows a promising result of *precision*. However, shadows of players, logos on the ground, and the high speed of the camera motion may still lead to failure in finding the high-speed puck. This is also revealed by the relatively low *recall*. A more detailed analysis will be performed in the following sections.

4.4.2. Tracking under Controlled Moving State

A comparison of the detection rate between the proposed approach and previous works is shown in Table 2, all the results of previous works are re-implemented according to their papers and evaluated on our dataset. Since all of the previous works considered pucks after shooting actions are invisible and didn't address cases under FM state, the comparison is only conducted between them and our approach under CM state. The result is shown in Table 2. When using the correlation filter as our base tracker and adding a re-identification phase to tackle the failure cases, detection and tracking performance can be improved with a much larger scale of data. Moreover, by introducing our re-identification phase under FM state, the proposed method is able to tackle previous considered 'invisible' pucks under FM state, leading to an increase of *TP* with a slight decrease of *F1* score. The result indicates that our proposed method can improve the puck tracking under CM state and track the puck in FM state additionally without hurting the performance.

From Table 3, by adding the re-identification phase, *precision* increases significantly due to the reduction of *FP*. This indicates that when the correlation filter tracker drifts because of the influence of similar objects nearby or partially, even fully occlusion, the re-identification phase is able to stop the tracker from absorbing negative samples, retrieving the puck after occlusion, and correct the trajectory to continue the tracking process.

We further classify all the video clips under CM state into 3 difficulty levels, based on how often the puck is occluded. The three difficulties are selected based on the degree of occlusion with below 30%, 50% and above 70%. Then a comparative experiment is carried out to determine the ability of the proposed approach to deal with occlusion problems. Results are shown in Table 4,

For samples of easy or medium levels, the puck usually endures the occlusion of sticks or body parts of players, the difference may be the rate of occlusions during the video clip. When dealing with

Table 3
Overall Performance.

Method	TP	FP	FN	TN	Pcs	Rec	F1
Plain Tracking	3423	3503	8481	1756	49.4%	28.7%	36.3%
PT + CM Reid	8437	1463	651	3542	85.2%	92.8%	88.9%
PT + FM Reid	1762	155	864	199	91.9%	67.1%	77.6%

Table 4
Ability of Re-ID to tackle different rate of occlusion.

Difficulty	GT	TP	FP	FN	TN	Pcs.	Rec.	F1
Easy	5273	4862	733	308	2038	86.9%	94.0%	90.3%
Medium	3449	3228	535	214	1249	85.7%	93.8%	89.6%
Hard	655	347	195	129	255	64.0%	72.9%	68.2%

such kind of problems, our approach is able to retrieve the puck after the short-term occlusion and maintain the tracking process. All the metrics remain at a high level. However, for samples of hard levels, more complex scenes with more players with faster moves serve as a basic environment. Sometimes the possession of the puck may transfer to the other team behind occlusion, which leads to a problem of changing moving direction secretly and moving out of the search area. This kind of problem can easily cause failure and increase *FP* and *FN*, affecting *precision* and *recall*. Through the comparison result, our approach demonstrates the ability to tackle usual occlusions with high-level metrics and shows the potential of dealing with hard conditions with a *recall* of 72.9%.

4.4.3. Tracking under Free Moving State

There are no prior published attempts to track the puck at high speed, which is hazy and unclear. We therefore calculate metrics of samples with different speed, and show puck examples with different speed in Fig. 8. Different speed is classified with respect to the average moving pixels in the video between consecutive1 5 frames. For fast, medium, and slow speed, we select 5 *pixels/frame*, 10 *pixels/frame*, and 20 *pixels/frame* under a frame rate of 25 fps for video sequences. As shown in Table 5, metrics of dealing with different speed indicate that pucks with medium speed are the hardest ones. Our Re-ID phase targeting different scenarios should hold the responsibility of this. Re-ID under CM state is plausible to deal with slow-speed pucks exhibiting rather a ball-like shape and darker color, while techniques used under FM state possess a strong ability to find the puck under hazy color and stick-like shape. Both mentioned above leave an awkward situation of pucks with medium speed, where colors are neither clear nor hazy, and the shape is hard to be categorized. To compensate for this situation, Kalman estimator is introduced to predict the location when neither of the techniques is able to find the puck. Also, Kalman estimator acts as a remedy procedure on the hard case shown in Fig. 8c, where simple gray-level thresholding is incapable of separating the puck from lines on the ice surface. Correlation filter is used to complete the trajectory where the puck sinks in the shadow of players. Combined with detection results from ellipse fitting, Kalman estimator, and the correlation filter, the evaluation result, shown in Table 5, gives an acceptable outcome of detecting and tracking the hard-to-find high-speed puck. Quantitative results prove that our re-

Table 6
Effect of Different Tracker. Best in bold.

Tracker	Pcs.	Rec.	F1
KCF[32]	75.7%	80.7%	78.0%
DCF[33]	78.4%	75.3%	76.8%
C-COT[34]	80.3%	84.2%	82.3%
ECO[28](Our Selection in paper)	**84.2%**	**85.0%**	**84.6%**

(a) Low (b) Medium (c) High

Fig. 8. Examples of pucks with different speed.

Table 5
Ability of Re-ID to find puck with different speed.

Speed($\frac{pixels}{frame}$)	GT	TP	FP	FN	TN	Pcs.	Rec.	F1
5	773	476	53	255	142	90.0%	65.1%	75.6%
10	526	202	58	302	33	77.7%	40.1%	52.9%
20	1427	1084	44	307	24	96.1%	77.9%	86.1%

Table 7
Effect of Different Parameter Settings for all the components.

State	Parameters		Pcs.	Rec.	F1
CM & FM	$\delta_{cf} = 0.2$	$\delta_{ss} = 0.8$	86.3%	78.4%	82.2%
	$\delta_{cf} = 0.4$	$\delta_{ss} = 0.2$	80.4%	86.2%	83.2%
	$\delta_{cf} = 0.2$	$\delta_{ss} = 0.4$	84.2%	85.0%	84.6%
CM	$\delta_{SSIM} = 0.2$		82.0%	80.2%	81.1%
	$\delta_{SSIM} = 0.8$		83.5%	85.0%	84.2%
	$\delta_{SSIM} = 0.4$		85.2%	92.8%	88.9%
FM	$\delta_{SSIM} = 0.8$	$\delta_{angle} = 0.2$	70.4%	45.9%	55.6%
	$\delta_{SSIM} = 0.2$	$\delta_{angle} = 0.8$	75.2%	70.8%	72.9%
	$\delta_{SSIM} = 0.5$	$\delta_{angle} = 0.5$	91.9%	67.1%	77.6%
CM & FM	$1.5\times$		84.2%	85.0%	84.6%
	$2\times$		70.6%	90.5%	79.3%
	$3\times$		56.3%	92.6%	70.0%

identification is able to locate and track the puck under extreme circumstances of high speed.

4.5. Ablation Study

In this section, we evaluate the effectiveness of different parameter settings for all the components of our proposed method. An ablation study of different tracker selections is carried out first in Table 6. Then different parameter settings for our Failure Score, Re-identification and also the search region are detailed evaluated in Table 7.

4.5.1. Effect of Different Tracker Selection.

To evaluate the performance of our proposed method when using different trackers, we select KCF [32], DCF [33], C-COT [34], and ECO [28] as our base tracker. The results are shown in 6, only the *precision*, *recall*, and *F1* score are shown for clear comparison. As shown by the result, the choice of the tracker does not hold a decisive influence on the performance, with different correlation trackers selected, the overall performance only changes relatively small margin. And with the improvement of the tracker, the *F1* score along with the *precision* and *recall* are increased, which shows the potential of our method by embedding with a powerful tracker.

4.5.2. Effect of Different Parameter Settings

We conduct experiments in Table 7 on four groups of parameters in this section, weights selected in the Failure Score, weights selection in extreme cases, settings of Re-ID score, and the size of the search window. Demonstrated in the top of Table 7, when enlarging the weight of the correlation filter and downgrading the importance of shape-similarity, the *FP* increases and *FN* drops and leads to changes of *precision* and *recall*, but eventually hurt the performance of *F1* score and vice versa. Then the middle of Table 7 shows that $\delta_{SSIM} = 0.4$ yields the best results under CM state for all the metrics and the weights for *SSIM* and the angle of movement should be balanced since they share the similar importance for estimating pucks under FM state. Finally, by enlarging the search window size for our method, more areas are introduced to search the puck, and this leads to a decrease of *FN* and an increase of *FP*. However, since much more interference will be selected if the window is large, the overall *F1* score will drop significantly. Hence, we select 1.5 times of current puck area as the search window.

5. Conclusion, Limitations and Future Work

In this paper, we propose a real-time ice hockey puck detection and tracking approach solely depending on image processing, which consists of two phases, plain tracking and re-identification. We use a combination of correlation filter, contour fitting and motion estimation approaches to detect and track pucks from both controlled moving and free moving state. We provide a solution for incorporating state-of-the-art trackers to track the low-texture ice hockey pucks, and present a re-identification phase to track high-speed pucks after shooting actions. This is the first attempt at trying to locate and track pucks with shifted shapes caused by high speed after shooting actions by a player to our best knowledge. Thorough analysis and experiments are conducted to show that our proposed approach can maintain a steady tracking process and is able to deal with constantly existing severe occlusion and reclaim the puck after its reappearance.

However, failure cases show that there are still challenges and difficulties in detecting and tracking pucks. Using techniques such as action recognition or scene understanding to analyze players' movements may provide a reasonable search area to eliminate the influence of players' interactions and camera motion. Also, given the fact that the puck in a broadcast video frame is relatively small and contains monotonous colors, thus there are always similar objects that may yield interference. Moreover, due to the peculiarity of the ice hockey game, players crowding together, sudden change of possession of the puck, even fly motion of the puck after a powered hit by players, robust long-term identifying and tracking the puck solely depending on image processing is still burdensome.

In addition, for better coping with the above problems, a more complex analysis approach with scene understanding is possible for future work. By using the action recognition technique, shooting actions can be analyzed. The technique can also determine the possible direction and location of the puck in the next few frames. By incorporating more cameras from different angles, occlusions can be solved thoroughly. Furthermore, by projecting the view angle of cameras to a bird's-eye view, the trajectory of the puck and all the players can be seen clearly. And introducing trajectory estimation approaches may optimize the moving trend of the puck and even predict future locations.

CRediT authorship contribution statement

Muyu Li: Conceptualization, Methodology, Software, Writing - original draft. **Henan Hu:** Software, Validation, Writing - review & editing. **Hong Yan:** Supervision, Writing - review & editing.

Data availability

The authors do not have permission to share data.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported by Hong Kong Innovation and Technology Commission (InnoHK Project CIMDA) and Hong Kong Research Grants Council (Project CityU 11204821).

References

[1] X. Yu, C. Xu, H.W. Leong, Q. Tian, Q. Tang, K.W. Wan, Trajectory-based ball detection and tracking with applications to semantic analysis of broadcast soccer video, in: Proceedings of the eleventh ACM international conference on Multimedia, 2003, pp. 11–20.
[2] X. Yu, C. Xu, Q. Tian, H.W. Leong, A ball tracking framework for broadcast soccer video, in: 2003 International Conference on multimedia and expo. ICME'03. proceedings (Cat. No. 03TH8698), Vol. 2, IEEE, 2003, pp. II–273.
[3] W.-L. Lu, J.-A. Ting, J.J. Little, K.P. Murphy, Learning to track and identify players from broadcast sports videos, IEEE transactions on pattern analysis and machine intelligence 35 (7) (2013) 1704–1716.
[4] X. Wei, L. Sha, P. Lucey, P. Carr, S. Sridharan, I. Matthews, Predicting ball ownership in basketball from a monocular view using only player trajectories, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2015, pp. 63–70.
[5] W. Chen, Y.-J. Zhang, Tracking ball and players with applications to highlight ranking of broadcasting table tennis video, in: The Proceedings of the Multiconference on Computational Engineering in Systems Applications, Vol. 2, IEEE, 2006, pp. 1896–1903.
[6] F. Yan, W. Christmas, J. Kittler, A maximum a posteriori probability viterbi data association algorithm for ball tracking in sports video, 18th International Conference on Pattern Recognition (ICPR'06), Vol. 1, IEEE, 2006, pp. 279–282.
[7] M. Yakut, N. Kehtarnavaz, Ice-hockey puck detection and tracking for video highlighting, Signal, Image and Video Processing 10 (3) (2016) 527–533.
[8] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), Vol. 1, Ieee, 2005, pp. 886–893.
[9] M.A. Zaveri, S.N. Merchant, U.B. Desai, Small and fast moving object detection and tracking in sports video sequences, 2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No. 04TH8763), Vol. 3, IEEE, 2004, pp. 1539–1542.
[10] K. Choi, Y. Seo, Probabilistic tracking of the soccer ball, in: International Workshop on Statistical Methods in Video Processing, Springer, 2004, pp. 50–60.
[11] P. Viola, M. Jones, et al., Robust real-time object detection, International journal of computer vision 4 (34–47) (2001) 4.
[12] A.M. Sánchez, M.A. Patricio, J. Garcia, J.M. Molina, Video tracking improvement using context-based information, in: 2007 10th International Conference on Information Fusion, IEEE, 2007, pp. 1–7.
[13] J. Czyz, Object detection in video via particle filters, 18th International Conference on Pattern Recognition (ICPR'06), Vol. 1, IEEE, 2006, pp. 820–823.
[14] H. Chang, Z. Feng, X. Wei, Object recognition and tracking with maximum likelihood bidirectional associative memory networks, Neurocomputing 72 (1–3) (2008) 278–292.
[15] J.-N. Chi, C. Qian, P. Zhang, W. Xiao, L. Xie, A novel elm based adaptive kalman filter tracking algorithm, Neurocomputing 128 (2014) 42–49.
[16] J.M. Pak, C.K. Ahn, Y.H. Mo, M.T. Lim, M.K. Song, Maximum likelihood fir filter for visual object tracking, Neurocomputing 216 (2016) 543–553.
[17] R. Cavallaro, The foxtrax hockey puck tracking system, IEEE Computer Graphics and Applications 17 (2) (1997) 6–12.
[18] D. Liang, Y. Liu, Q. Huang, W. Gao, A scheme for ball detection and tracking in broadcast soccer video, in: Pacific-Rim Conference on Multimedia, Springer, 2005, pp. 864–875.
[19] X. Wang, E. Türetken, F. Fleuret, P. Fua, Tracking interacting objects optimally using integer programming, in: European Conference on Computer Vision, Springer, 2014, pp. 17–32.
[20] X. Wang, E. Türetken, F. Fleuret, P. Fua, Tracking interacting objects using intertwined flows, IEEE transactions on pattern analysis and machine intelligence 38 (11) (2015) 2312–2326.
[21] A. Maksai, X. Wang, P. Fua, What players do with the ball: A physically constrained interaction modeling, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 972–981.
[22] J. Ren, J. Orwell, G.A. Jones, M. Xu, Tracking the soccer ball using multiple fixed cameras, Computer Vision and Image Understanding 113 (5) (2009) 633–642.
[23] H. Pidaparthy, J. Elder, Keep your eye on the puck: Automatic hockey videography, in: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2019, pp. 1636–1644.
[24] K. Vats, W. McNally, C. Dulhanty, Z.Q. Lin, D.A. Clausi, J. Zelek, Pucknet: Estimating hockey puck location from broadcast video, arXiv preprint arXiv:1912.05107 (2019).
[25] K. Vats, M. Fani, D.A. Clausi, J. Zelek, Puck localization and multi-task event recognition in broadcast hockey videos, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 4567–4575.
[26] X. Yang, Where is the puck? tiny and fast-moving object detection in videos (2021).
[27] J. Van De Weijer, C. Schmid, J. Verbeek, D. Larlus, Learning color names for real-world applications, IEEE Transactions on Image Processing 18 (7) (2009) 1512–1523.
[28] M. Danelljan, G. Bhat, F. Shahbaz Khan, M. Felsberg, Eco: Efficient convolution operators for tracking, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 6638–6646.
[29] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems 25 (2012) 1097–1105.
[30] M. Danelljan, G. Hager, F. Shahbaz Khan, M. Felsberg, Learning spatially regularized correlation filters for visual tracking, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 4310–4318.
[31] J. Nocedal, S. Wright, Numerical optimization, Springer Science & Business Media, 2006.
[32] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE transactions on pattern analysis and machine intelligence 37 (3) (2014) 583–596.
[33] A. Lukezic, T. Vojir, L.?Cehovin Zajc, J. Matas, M. Kristan, Discriminative correlation filter with channel and spatial reliability, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 6309–6318.
[34] M. Danelljan, A. Robinson, F.S. Khan, M. Felsberg, Beyond correlation filters: Learning continuous convolution operators for visual tracking, in: European conference on computer vision, Springer, 2016, pp. 472–488.
[35] M.-K. Hu, Visual pattern recognition by moment invariants, IRE transactions on information theory 8 (2) (1962) 179–187.
[36] A.W. Fitzgibbon, R.B. Fisher, et al., A buyer's guide to conic fitting, Citeseer, 1996.
[37] S.D. Roth, Ray casting for modeling solids, Computer graphics and image processing 18 (2) (1982) 109–144.
[38] R.E. Kalman, A new approach to linear filtering and prediction problems (1960).
[39] Y. Wu, J. Lim, M.-H. Yang, Online object tracking: A benchmark, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2013, pp. 2411–2418.

Muyu Li received the BSc degree in Electrical Science and Technology from University of Science and Technology of China in 2014 and the Ph.D. Degree from University of Chinese Academy of Sciences, China. Currently, he is working in the Center for Intelligent Multidimensional Data Analysis Limited, Hong Kong. His research interests include visual tracking and deep learning.

Henan Hu received his B.Eng. degree in Mechanical Engineering from Changchun University of Science and Technology, China in 2018. Currently, she is working towards the Ph.D. degree at University of Chinese Academy of Sciences, China. Her research interests include 3D object detection and deep learning.

Hong Yan received his Ph.D. degree from Yale University. He was a Professor of Imaging Science at the University of Sydney and is currently Chair Professor of Computer Engineering at City University of Hong Kong. His research interests include image processing, pattern recognition, and bioinformatics, and he has over 600 journal and conference publications in these areas. Professor Yan is an IEEE Fellow and IAPR Fellow, and he received the 2016 Norbert Wiener Award from the IEEE SMC Society for contributions to image and biomolecular pattern recognition techniques.