A robust sperm cell tracking algorithm using uneven lighting image fixing and improved branch and bound algorithm

Ahmad Alhaj Alabdulla, Abdulsamet Haşıloğlu, Emrah Hicazi Aksu

1 Department of Computer Engineering, Faculty of Engineering, Ataturk University, Erzurum, Turkey
2 Department of Reproduction and Artificial Insemination, Faculty of Veterinary Medicine, Ataturk University, Erzurum, Turkey

Abstract
An accurate and robust sperm cells tracking algorithm that is able to detect and track sperm cells in videos with high accuracy and efficiency is presented. It is fast enough to process approximately 30 frames per second. It can find the correct path and measure motility parameters for each sperm. It can also adapt with different types of images coming from different cameras and bad recording conditions. Specifically, a new way is offered to optimize uneven lighting images to improve sperm cells detection which gives us the ability to get more accurate tracking results. The shape of each detected object is used to specify collided sperms and utilized dynamic gates which become bigger and smaller according to the sperm cell's speed. For assigning tracks to the detected sperm cells positions an improved version of branch and bound algorithm which is faster than the normal one is offered. This sperm cells tracking algorithm outperforms many of the previous algorithms as it has lower error rate in both sperm detection and tracking. It is compared with six other algorithms, and it gives lower tracking error rates. This method will allow doctors and researchers to obtain sperm motility data instantly and accurately.

1 | INTRODUCTION

Computer-aided sperm analysis systems began in the 1980s. Since then, dozens of sperm movement analysis and morphological analysis systems have been developed. Many researches have been done to develop highly efficient algorithms to detect and track sperm cells and calculate their motility parameters. These algorithms and systems help us in the analysis and production of semen needed for animal production. Some of them help us analyse male fertility. Sperm cells are classified according to their speed and the way of movement. Fast progressive, slow progressive, non-progressive and immotile are the main classes for sperm cells in general. Depending on them, we can calculate the percentage of progressive motility for a semen sample.

Lots of researches have been made for tracking sperm cells. In 2005, Nafisi et al. presented a template matching algorithm to track sperm cells in video taken using a camera attached to the eyepiece of a regular microscope [1]. They applied image enhancement steps which enabled the algorithm to work with different image acquisition conditions, and then they used template matching for tracking only one sperm at the same time. The lab technician should specify the start point for tracking. This algorithm works well, but it may fail if several collisions occur in highly concentrated samples.

In 2006, Shi et al. presented a robust single sperm cell tracking algorithm [2]. It creates a region of interest around the selected sperm and uses a modified four-class thresholding method to detect the sperm. Nearest neighbour, speed check and collision detection features are used to track the sperm. The algorithm provides a very good result in tracking the sperm but it needs to be trained by the statistical measurements of normal deviations from closely swimming sperm cells to create costs functions to choose the most likely association before and after a collision occurs.

In 2008, Sørensen et al. used a combination of Kalman filter and particle filter to track multiple human spermatozoa in photo-contrast images [3]. For assignment they used Hungarian algorithm with rectangular cost matrix. The costs were calculated using hidden Markov models. This algorithm has a very good tracking results but sometimes labels swap occurs after collisions between sperm cells. In 2011, Ravanfar et al. focused on solving sperm cells collisions by using particle filter for
tracking and Watershed algorithm for segmentation [4]. They used a common camera to take videos to make it easier for laboratories to use the algorithm. An adaptive temporal median filter was used to extract the background of each frame. In addition to particle filter they used knowledge based constrains to estimate collisions, regardless of the sperm head orientation.

In 2013, Liu et al. presented a multi-sperm tracking algorithm [5]. It tracks sperm tail in addition to sperm head. And they detected that there is a correlation between sperm tail beating amplitude and sperm velocity. In 2014, Nurhadiyatna et al. made a comparison between multiple sperm cells detection and tracking algorithms [6], and they got a result that Gaussian mixture model with hole filling algorithm was the best for detection step. In the same year, Imani et al. used frame differencing to extract the background and proposed an improved version of a non-linear diffusion filtering to smooth the image without removing sperm cells edges [7]. They used Hungarian algorithm to make the association. The accuracy of the algorithm is 97% if the sperm cells count is less than 10, but it does not succeed with highly concentrated samples.

In 2015, Jati et al. used particle swarm optimization to track one sperm cell [8]. They used search window technique to reduce the search space. The search window should be tuned carefully because it depends on data and frame resolution. Wrong size for search window will decrease the performance, so the precise size for search window is necessary to get optimal performance.

Beya et al. created a dictionary using SURF, HOG and LBP features with bag-of-words approach and support vector machine SVM classifier to improve the detection step, and they used mean shift filter for tracking step [9]. The results show that this algorithm has high precision and accuracy with low RMSE. In 2016, Jati et al. utilized Kalman filter to estimate sperm cells positions [10]. Then they used Hungarian assignment method to assign each new detected sperm to the correct trajectory. For sperm cells detection they used noise reduction filter. After that they applied Otsu's method for thresholding. To find $\times \times \times$ position for each sperm they tried to find the maximal intensity pixel. Low frame rate videos were used to test the algorithm. It has 90% accuracy in detection step but sometimes it swaps labels between closely swimming sperm cells.

In 2017, Hesar et al. made a modified version of Gaussian mixture probability hypothesis density (GM-PHD) filter to make it able to work with the non-linear movement of the sperm cells [11]. They used local thresholding with sliding window pixel by pixel to segment sperm cells, and they got good results in tracking collided sperm cells. In 2017, Urbano et al. also presented a robust and accurate method for tracking sperm cells in recorded videos [12]. The method starts with smoothing the image with Gaussian filter, then they used Mexican hat filter to enhance the sperm cells in the image. To binarize the image, they used Otsu's method, and after that they used blob analysis to detect sperm cells positions. For sperm cells assignment they presented a new method utilizing the join probabilistic data association filter. But this assignment method has high computational cost. So, they used a clustering algorithm to divide the assignment problem into smaller ones. Urbano et al.'s algorithm was robust. But they said that they only processed two frames in one second. So, the algorithm was not fast enough to work with live video stream coming from a camera.

In 2018, Arasteh et al. used a hybrid dynamic Bayesian network HDBN to solve the tracking problem [13], and they used Munkres algorithm for assignment. They said that their algorithm was better than all other sperm cells tracking algorithm, but it is slow and needs a few seconds for processing one frame. In the same year, Imbachí et al. developed the first version of open-source software for sperm cells tracking [14]. The algorithm utilized Kalman filter for each track. To handle occlusion, they used a simplified occlusion handling method. In 2018, Masdiyasa et al. also presented a new method for sperm cells tracking [15]. They used frame differencing to extract the background, then they used Otsu’s method for thresholding. After that they made some morphological operations to remove noise. At tracking step, they checked the tracking paths using linear regression.

Wang et al. compared four algorithms Nearest Neighbour (NN), probabilistic data association (PDA), joint probabilistic data association (JPDA) and joint probability data association filter with track-coalescence-avoiding (JPDA*) [16], and they said that (JPDA*) has the best performance, but it has high Computational Complexity. In 2019, Mostajer et al. presented a new sperm cells tracking method [17]. It starts with detecting sperm cells by a hybrid segmentation method that depends on Kittler and modified-Kittler thresholding methods. Then they removed the non-motile sperm cells form the detections to make tracking step easier and simpler. For tracking, they used an enhanced version of adaptive window average speed AWAS algorithm. They said that this algorithm has superior accuracy and higher speed to the other competitive algorithms, but for low density samples Kalman filter works better than AWAS algorithm.

Qi et al. also presented another sperm tracking algorithm utilizing Gaussian mixture model GMM for segmentation and tracking [18], and they got good results for detecting forward progressive. For counting the non-progressive sperm cells, they used threshold segmentation which is not reliable for noisy images. Separating the motile sperm cells processing from non-motile sperm cells makes tracking step easier and simpler, but it will need more calculations to determine the non-motile sperm cells. And it may have errors if the non-motile or died sperm cells are moving slowly in the liquid.

This paper presents a new sperm cells tracking method that starts by fixing the image light. Then it utilizes Otsu’s method to binarize the fixed image. After that, it uses a method similar to [19] to find the contours of the detected objects. Based on the size and shape of each contour, a decision will be made to determine if it is a sperm cell, collided sperm cell or non-sperm object. Every collision is considered as two detections before the assignment (association between the last position of each track and the newly detected sperm cells positions). For assignment, an improved version of branch and bound algorithm (B&B) is proposed. Three results will be obtained from the assignment (assigned tracks, unassigned tracks, and unassigned detections). For the assigned tracks, their Kalman filters
will be updated by the newly detected positions and each position will be saved in the corresponding track. For the unassigned tracks, only the estimated position will be saved in the corresponding track. For the unassigned detections, if this detection is not a collision, a new track will be initialized with a new Kalman filter. When the detection is collision, it means that it is added two times to the detections vector.

If none of the two added detections has been assigned to any track, one new track will be initialized with a new Kalman filter.

2 | MATERIALS AND METHODS

We present a robust and adaptive sperm detection and tracking method, which gives accurate detection and tracking results for tens of motile sperms in the recorded videos or even in live stream.

In addition, it is able to deal with videos taken from multiple cameras, or even videos taken with bad acquisition conditions.

The most difficult problem in tracking sperms is to track closely swimming sperm cells or to track sperms whose trajectories intersect. In the suggested method, uneven lighting images are fixed to get a clear and precious foreground. Collisions can be detected, and every sperm can be assigned to its correct track after the collision happens. An improve version of the (B&B) algorithm is also suggested to assign newly detected sperms positions to the estimated tracks. This assignment algorithm takes in consideration the cost of not assigning a track to any detected sperm. For each track, a Kalman filter is used to estimate the new position of the track. This algorithm is tested by coloured and greyscale videos of various resolution which have been taken by different cameras and different magnifications. Figure 1 shows the algorithm block diagram.

While tracking sperms, trajectories are made for each one, and used to calculate the important sperm motility parameters like curvilinear velocity (VCL), straight line velocity (VSL), average line velocity (VAP), linearity of progressive motility (LIN), curvilinear path wobbling (WOB), average path straightness (STR), the average rate of the curvilinear path crossing the average path (Beat Cross Frequency BCF), amplitude of lateral head displacement (ALH), and mean angular displacement (MAD). These parameters can be used to classify multiple sperms in the recorded videos, and they give us the percentage of progressive motility and total motility. They play important role in determining male fertility, and they can be used in semen production units.

To specify the location of each sperm cell, a processing algorithm was made to enhance the image before binarizing it to get the foreground. The code was written in python with the help of OpenCV library. The recorded videos were in AVI format with multiple dimensions (640 × 480) pixels and (768 × 576) pixels. Frame rate is 25 for some videos and 50 for others. Two coloured videos were taken using phase contrast Olympus BX43 microscope (400X magnification) with Basler pIA640-210ge GigE camera. The coloured videos are clear and have green background. Every sperm has a dark grey ellipse head and plain tail. There is a blur liquid moving around the head of the motile sperm cells. The other videos were taken using dark field microscopes (300X magnification). These videos were greyscale where sperm cells, other obstacles and debris appeared in white colour and the background was dark grey. Every sperm cell had ellipse head with blur tail. The videos were taken for multiple species such as bull, stallion, and ram. Ten videos for bull sperm cells. Five videos for stallion and five for ram.

2.1 | Sperm cells detection

The most important step in sperm cells motility analysis algorithms is the sperm cells detection step. If some sperm cells were missed or if false ones were detected, this will extremely affect tracking and lead to higher error rate in calculating sperm cells motility parameters.

In previous studies, multiple methods are used. Some algorithms depend on filtering the image by low pass filter like Gaussian filter multiple times to reduce noise and then they apply Laplacian of Gaussian filter. Other algorithms enhance the image by applying contrast stretching. After that they apply global thresholding like Otsu’s method, or local adaptive thresholding; and sometimes they use ellipse detection algorithm after thresholding. Some algorithms are only interested in motile sperm cells, so they use methods like Gaussian mixture model (GMM) to detect the sperms.

We present a new method to enhance the image and fix uneven light in sperm cells images. The ideal sperm cells image has two classes in general, the background and the sperm cells. The background has single intensity value and sperm cells are detected. In addition, it is able to deal with videos taken from multiple cameras, or even videos taken with bad acquisition conditions.

The most difficult problem in tracking sperms is to track closely swimming sperm cells or to track sperms whose trajectories intersect. In the suggested method, uneven lighting images are fixed to get a clear and precious foreground. Collisions can be detected, and every sperm can be assigned to its correct track after the collision happens. An improve version of the (B&B) algorithm is also suggested to assign newly detected sperms positions to the estimated tracks. This assignment algorithm takes in consideration the cost of not assigning a track to any detected sperm. For each track, a Kalman filter is used to estimate the new position of the track. This algorithm is tested by coloured and greyscale videos of various resolution which have been taken by different cameras and different magnifications. Figure 1 shows the algorithm block diagram.
FIGURE 2  Light fixing steps (a) original image, (b) light difference, (c) smoothed light difference, (d) fixed image

calculated. Then, the light difference of the image will be extracted and after that this light will be smoothed by average filter. And the last step will be re-adding the fixed light difference to the image.

These steps are illustrated in Equations (1)–(5). And as shown in Figure 2, the original image is in (a). In (b), the light difference. In (c), the light difference after smoothing. The final step in (d) is to re-add the light difference to the original image to get the fixed one.

\[
\text{AvgIm}(i, j) = \frac{1}{(2w + 1)^2} \sum_{k = i-w}^{i+w} \sum_{h = j-w}^{j+w} \text{Im}(k, h)
\]

\[
\text{Average} = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} \text{Im}(i, j)}{M \times N}
\]

\[
\text{Light DIFF}(i, j) = \text{Average} - \text{AvgIm}(i, j)
\]

\[
\text{Fixed LIGHT DIFF}(i, j) = \frac{1}{(2w + 1)^2} \sum_{k = i-w}^{i+w} \sum_{h = j-w}^{j+w} \text{Light DIFF}(k, h)
\]

\[
\text{Fixed Image}(i, j) = \text{Im}(i, j) + \text{Fixed LIGHT DIFF}(i, j),
\]

where, \(\text{Im}\) is the original image, and \(\text{AvgIm}\) is the average image. \(\text{Average}\) is the mean of the original image. \(\text{Light DIFF}\) is the light difference. \(\text{Fixed LIGHT DIFF}\) is the fixed light difference, and \(\text{Fixed Image}\) is the fixed image. \(M\) and \(N\) are the image’s height and width, \(w\) is window size.

The most difficult thing in image segmentation is to choose the right threshold. But with this method this step becomes very easy by fixing the image light before. To see the result of this method, we compare it with applying Otsu threshold directly to the original image. In Figure 3 the difference between the original image and the fixed image histograms is shown. The background becomes a single big pulse and the segmentation process becomes easier. And as shown in Figure 4, the suggested detection method gives us accurate segmentation of sperm cells when we apply Otsu thresholding.

After getting the black and white image, sperm cells’ tails and the small particles will be removed by applying some morphological operations. Erosion will be applied by \((3 \times 3)\) disk shaped structural element twice. Then dilation will be applied by the same structural element. The result is shown in Figure 5a.

2.2 Collision detection

To detect the positions of the sperm cells, most of the previous studies used blob analysis algorithm. They were concerned about the sperm cells positions. In the suggested method, the sperm shape can be accurately detected, so the shape of every sperm can be analysed. So, we apply a method similar to Suzuki [19] algorithm which does a topological structural analysis of
binary images to find the contour of each object in the image. Figure 5b shows the contour of each sperm in the image. The centre and the area of each contour will be calculated. Then the convex hull algorithm [20] for each contour will be applied to find the smallest convex polygon that contains all the points of the contour. After that the steps in Table 1 will be applied to classify each contour.

Average sperm head area $ma = \text{Average of detected contours areas}$. This value can be dynamically calculated because sperm cells have similar head shapes and sizes.

Or it can be fixed to a specific number, because the average sperm head area of a specific species can be precalculated if we fix the magnification in the microscope to a specific valued. So, the user can specify it before starting the algorithm.

- **Minimum Sperm Head Area** $mina = ma/2$
- **Maximum Sperm Head Area** $maxa = ma \times 3$

In this algorithm, for every frame the mean area for all contours will be calculated and used to specify the minimum and maximum area for sperm cells. Secondly, the convex hull area and centre for each contour will be calculated. Then every contour will be checked to decide whether it is a sperm or not. If the area of the current contour is between the minimum and maximum, then this contour represents a sperm cell. After that, the existence of collision or occlusion at this contour will be checked. If the contour area is big enough, it represents two sperm cells not only one. If a big difference between the contour’s area and its convex hull area is detected, it means that the current contour is not convex, and it represents a collision between two or more sperm cells. The constants $a$ and $b$ will be controlled by the user.

**TABLE 1** Collision detection pseudo code

| Step | Description |
|------|-------------|
| 1.   | **For** each contour in the frame: |
| 1.1  | $ca = \text{area of the current contour}$ |
| 1.2  | centre = current contour centre |
| 1.3  | Hull Area $ha = \text{current contour convex hull area}$ |
| 1.4  | **If** $ca > mina$ and $ca < maxa$ |
| 1.4.1| **If** $ca > a \times ma$ or $(ca > ma$ and $ha > b \times ca)$ |
| 1.4.1.1| Add current contour to detections list $DL$ twice |
| 1.4.1.2| Label them as collision |
| 1.4.2| **Else** |
| 1.4.2.1| Add current contour to detections list $DL$ |
| 1.4.2.2| Label current contour as not-collision |
| 1.5  | **Else** |
| 1.5.1| Label contour as rejected |
2.3 Sperm cells tracking

2.3.1 Kalman filter

In order to track sperm cells, a model which can predict the next position of a sperm cell depending on its previous positions is needed. This is very hard and complex to achieve because the abnormal sperm cells in general do not follow a regular physical model. So, Kalman filter is used as an approximated model to predict the new position for each sperm. One Kalman filter will be initialized for each sperm (track). This Kalman filter will predict the position and speed of each sperm and will be corrected by the new measured position of each sperm cell. This is the suggested Kalman filter:

\[
X_{k|k-1} = AX_{k-1|k-1} + \omega_k \tag{6}
\]

\[
P_{k|k-1} = AP_{k-1|k-1}A^T + Q_k \tag{7}
\]

(6) (7) are the predication equations. \(X\) is 4x1 state matrix and \(P\) is 4x4 state covariance matrix. We consider \(\omega_k\)—the predicted state noise—as zero. \(Q\) is the process noise covariance. \(Q\) is updated according to [21]

\[
Q_k = \alpha Q_{k-1} + \beta v_k v_k^T + \gamma Q_0, \tag{8}
\]

where \(v_k\) is the new detected position minus the predicted position of the sperm cell and \(\alpha + \beta + \gamma = 1\).

\[
A = \begin{bmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\]

\(A\) is the transition matrix and \(dt = \frac{1}{\text{frame rate}}\).

\[
Z_k = HX_k + v_k, \tag{9}
\]

\[
H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.
\]

\(X_k\) is 2x1 the measured new position for the sperm cell and \(v_k\) is measurement noise. \(v_k\) here is considered as 0.

\[
K = \frac{P_{k|k-1}H^T}{HP_{k|k-1}H^T + R_k}, \tag{10}
\]

\[
R_k = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix},
\]

\[
X_{k|k} = X_{k|k-1} + K \left(Z_k - HX_{k|k-1}\right), \tag{11}
\]

\[
P_{k|k} = (I - KH) P_{k|k-1}. \tag{12}
\]

\(K\) is the Kalman gain. \(R_k\) is the covariance of the measurement noise. Equations (8)–(12) are the correction equations.

After predicitng the new position of each sperm cell, each track should be assigned to its correct sperm cell. Multiple algorithms have been used to do this like nearest neighbour, where each track is assigned to the nearest sperm cell detection. Nearest neighbour algorithm is fast but not accurate. Global nearest neighbour algorithm was also used where the assignment between the tracks and the sperm cells detection is done by finding the minimum cost of assigning each track to one sperm cell detection taking in consideration that the track may not be assigned to any sperm. It is fast and accurate. Multi-hypothesis tracker uses all the history of measurements for each track to assign the track to its new correct sperm cell detection. Whereas joint probabilistic data association filter uses only the previous state to find the correct assignment. But both of MHT and JPDAF have high computational complexity.
TABLE 2 The suggested improved branch and bound assignment algorithm pseudo code

1. root = node{trackID = -1, detectionID = -1, assigned = AV}
2. root.pathCost = 0
3. root.cost = 0
4. min = root
5. For i = 0:n-1
   5.1. nodeNoAssigning = node{trackID = i, detectionID = -1, assigned = min.assigned}
   5.2. nodeNoAssigning.pathCost = min.assigned + \[\text{GNN}\]
   5.3. nodeNoAssigning.cost = nodeNoAssigning.pathCost + calculateCost(i, min.assigned)
   5.4. bestNode = nodeNoAssigning
   5.5. For j = 0:n-1
      5.5.1. If not min.assigned[j]
         5.5.1.1. temp.assigned = min.assigned
         5.5.1.2. temp.assigned[j] = true
         5.5.1.3. child = node{trackID = i, detectionID = j, assigned = temp.assigned}
         5.5.1.4. child.pathCost = min.pathCost + \[\text{DM}\]
         5.5.1.5. child.cost = child.pathCost + calculateCost(i, child.assigned)
         5.5.1.6. If child.cost < bestNode.cost
            5.5.1.6.1. bestNode = child
      5.6. min = bestNode
   5.7. result[min.trackID] = min.detectionID
1. Function calculateCost(i, assigned)
   1.1. available vector AL = boolean vector, default true
   1.2. b = i + 1
   1.3. While b < m
      1.3.1. md = gate of track b
      1.3.2. mi = -1
      1.3.3. For j = 0:n-1
         1.3.3.1. k = SIM[b][j]
         1.3.3.2. If not assigned[k] and AL[k] and DM[b][j][k] <
            1.3.3.2.1. mi = k
            1.3.3.2.2. md = DM[b][j][k]
         1.3.3.3. break
      1.3.4. If mi not equal -1
         1.3.4.1. AL[mi] = false
         1.3.5. cost += md
      1.3.6. b = b + 1
   2. return cost

2.3.2 Assignment algorithm

In the proposed tracking algorithm an improved version of branch and bound algorithm is suggested as shown in Table 2. It is considered as global nearest neighbor algorithm. Urbano et al. [12] have compared some tracking algorithms with some synthesis images and found that JPDAF and GNN are the best. GNN is faster than JPDAF and MHT and gives as accurate results as them when they are applied to track sperm cells. The suggested assignment algorithm is fast enough to be used to track live stream of frames coming from a camera.

In order to assign tracks to their newly detected sperm cells we need to calculate the distance between the estimated position of the track which comes from the Kalman filter and the detected sperm cells. In some previous studies statistical distance is used to calculate distance. But statistical distance is not accurate and may mislead us when we have noisy image or when we deal with sperm cells which move fast in a random way. It can suddenly turn to another direction like Figure 6. The statistical distance here will appear farther than the real distance. To solve this, we use Euclidean distance because it is accurate and gives real value. As shown in Figure 6b the sperm in the fifth position was lost and became out of gate G4.

The distance from each track predicated position to each newly detected sperm cell will be calculated. Every track has its own gate. The gate is used to reduce calculation and prevent errors in tracking. The gate of each track is a circle around the predicted position of each track. The radius of the gate is calculated again at every new frame by the following equation

\[
G_i = b_1 G_{i-1} + b_2 (\text{maxSpeed}_{i-1} + \text{meanSpeed}_{i-1}) + b_3 G_0, \quad (13)
\]

where \(b_1, b_2 \) and \(b_3 \) are between 0 and 1. \(G_0 \) is the initial gate, so it will be large enough to detect the fast-moving sperms. \(\text{meanSpeed}_{i-1} \) and \(\text{maxSpeed}_{i-1} \) are the mean and maximum speeds of the track \(i \) at frame \(t-1 \). \(G_{i-1} \) is the gate size for track \(i \) at frame \(t \). So, this will allow the gate to shrink and expand according to the sperm cell’s speed.

\[
distanceMatrix(i, j) = \begin{cases} 
\text{distance} \left( X_{k_{i,t-1}}, X_{j,t} \right) \quad & \text{if distance} \left( X_{k_{i,t-1}}, X_{j,t} \right) < G_i \\
G_i \text{ otherwise} 
\end{cases}
\]

where \(i \) is the track’s number. \(j \) is the newly detected sperm’s number.

FIGURE 6 Comparing ellipse and circular gates: (a) Circular gate with Euclidean distance, (b) ellipse gate.
\(X_{i|t-1}^j\) is the predicted position for track \(i\) and it is multiplied by \(H\) to take only the predicted position without the predicted velocity.

\(X^j\) is the measured newly detected sperm cell \(j\) position.

\(G^j\) is the gate for track \(i\).

\(DM\) is the distance matrix (or cost matrix).

\(SIM\) is the indices matrix of sorted distance matrix for example if \(DM\) is \([[[90, 40, 30], [20, 10, 50]]\), \(SIM\) will be \([[2, 1, 0], [1, 0, 2]]\)).

\(m\) is the number of tracks.

\(n\) is the number of detections.

Assigned detections Vector \(AV\) is a Boolean vector with length of \(n\) and the default value of every element is false.

Every node in the search tree of the improved B&B has a trackID, detectionID, pathCost, cost, and the assigned vector.

The pathCost is the cost of the path between the root node and the current node. The node cost is the pathCost plus the minimum cost from the current node to any leaf node.

The assigned vector of the node is a Boolean vector with length of \(n\). The consecutive positions of each track will be saved in a coordinates list. If a track has been labelled as unassigned after that, three vectors will be extracted from the result vector: the assigned tracks vector which contains all the tracks assigned to detections, the unassigned tracks which contains all of the unassigned tracks (which have \(-1\) value in result vector), and the unassigned detected sperm cells vector which contains all of the unassigned detection.

In GNN algorithms like Hungarian algorithm, to allow a track to be not assigned to any detection we need to double the cost matrix size. To explain that suppose that we have the following cost matrix with two detections and three tracks \([[[11, 15, 30, 30, 30], [16, 12, 30, 30, 30], [11, 61, 30, 30, 30], [30, 30, 30, 30, 30]]\) and the gate size for each track is 30. If we use Hungarian algorithm, we need to expand the matrix to allow tracks to be not assigned to any detection like this \([[[11, 15, 30, 30, 30], [16, 12, 30, 30, 30], [11, 61, 30, 30, 30], [30, 30, 30, 30, 30]]\). Whereas the improved B&B can deal with the cost matrix without any expansion.

The suggested tracking algorithm pseudo code

| TABLE 3 | The suggested tracking algorithm pseudo code |
|---------|---------------------------------------------|
| 1.      | For each new frame:                         |
| 1.1.    | Fix light.                                  |
| 1.2.    | Find contours.                              |
| 1.3.    | Add non-collided accepted contours to the detections list of the current frame \(DL\). |
| 1.4.    | Add collided accepted contours to \(DL\) twice |
| 1.5.    | For each track:                             |
| 1.5.1.  | Predict the new position using the Kalman filter. |
| 1.5.2.  | If the prediction is outside the frame's borders, finish and save the track. |
| 1.5.3.  | If the track was unassigned three consecutive times, delete it (noise). |
| 1.6.    | Calculate distance matrix between every track predicted position and every new detection. |
| 1.7.    | Do assignment using the improved B&B Algorithm (the result will be the assigned tracks, unassigned tracks, and unassigned detections) |
| 1.8.    | For each assigned track:                    |
| 1.8.1.  | Save the new position to the coordinates list. |
| 1.8.2.  | Update the Kalman filter with the new measurement (the detection). |
| 1.9.    | For each unassigned track:                 |
| 1.9.1.  | Save the predicated position to the track's coordinates list. |
| 1.10.   | For each unassigned detection:             |
| 1.10.1. | If it is not-collided detection or collided but not assigned to any track, |
| 1.10.2. | Create new track |
| 1.10.3. | Create new Kalman filter for this track initialized by the detection's position. |
| 1.11.   | If no more frames, finish all tracks.       |
| 2.      | Calculate motility parameters for every finished track. |
Figure 7 shows the results for tracking sperm cells in video sample 1. The number above each track represents the ID of the track.

2.4 Parameters calculation

While tracking sperm cells, coordinates list (trajectories) for each track will be saved and used to calculate the important sperm motility parameters like curvilinear velocity (VCL), straight line velocity (VSL), average line velocity (VAP), linearity of progressive motility (LIN), curvilinear path wobbling (WOB), average path straightness (STR), the average rate of the curvilinear path crosses the average path (Beat Cross Frequency BCF), amplitude of lateral head displacement (ALH), and mean angular displacement (MAD). These parameters can be used to classify multiple sperms in the recorded videos. According to this classification we can calculate the percentage of progressive motility and total motility which play important role in determining fertility.

Figure 9 shows the difference between VCL, VSL and VAP. Figure 8 shows VCL, VSL and VAP on a sperm cell. Where VCL is in black colour, VSL in red and VAP in green colour.

In every track the consecutive detected positions of the track will be kept in the coordinates list \((x_0, y_0), (x_1, y_1), (x_2, y_2), \ldots (x_n, y_n)\)

\[
VCL = \frac{\text{frame rate}}{n} \sum_{b=1}^{n} \sqrt{(y_b - y_{b-1})^2 + (x_b - x_{b-1})^2} \quad (16)
\]

\[
VSL = \frac{\text{frame rate}}{n} \sqrt{(y_0 - y_1)^2 + (x_0 - x_1)^2} \quad (17)
\]

\[
VAP = \frac{\text{frame rate}}{n} \sum_{b=1}^{n} \sqrt{(y_{mb} - y_{mb-1})^2 + (x_{mb} - x_{mb-1})^2} \quad (18)
\]

\[
x_{mb} = \frac{1}{5} \sum_{i = b-2}^{b+2} x_i \quad (19)
\]

\[
y_{mb} = \frac{1}{5} \sum_{i = b-2}^{b+2} y_i \quad (20)
\]

\[
LIN = \frac{VSL}{VCL} \quad (21)
\]

\[
STR = \frac{VAP}{VCL} \quad (22)
\]

\[
WOB = \frac{VAP}{VCL} \quad (23)
\]

3 RESULTS

3.1 Test samples

Twenty video samples were used for testing the algorithm collected from various places. They were collected by various
cameras and microscopes. Ten videos were for bull, five videos for ram and five videos for stallion. They are in AVI format with multiple dimensions \((640 \times 480)\) pixels and \((768 \times 576)\) pixels. Frame rate was 25 for some videos and 50 for others. Some videos were greyscale where sperm cells and other obstacles appeared in white colour and the background was dark grey. Every sperm cell had an ellipse head with blurred tail. The coloured videos had a green background. Every sperm had a dark grey ellipse head. The code was written in python utilizing OpenCV library. The algorithm was tested on windows 10 operating system with Intel Core i5 2.60 GH and 8 GB ram. We also used the OR library test data for assignment problem to compare the branch and bound with the improved branch and bound http://people.brunel.ac.uk/~mastjjb/jeb/orlib/assigninfo.html

3.2 Performance metrics

To test the algorithm the test parameters defined in [22] were used. Four parameters of them FP, FN, FIT, and FIO were used. FP: false positive represents the situation when the detection step of the algorithm considers some non-sperm cell objects as sperm cells. Debris and other obstacles are considered as false positive when they are detected by the detection step. FN: false negative represents the situation when the detection step of the suggested algorithm loses the sperm cell. In this situation the track will not be assigned to any sperm cell and the estimated new position of the track will be considered as the current position of the lost sperm. FIT: falsely identified tracker represents the situation when the track loses the sperm cell and after that, this track is finished and a new track is created again for the same sperm cell. In this situation one sperm cell will have multiple tracks. FIO: falsely identified object represents the situation when the track is assigned to the wrong detection. Maybe two tracks swap their sperm cells. So, we can say that FP and FN are used to measure errors in the detection step of the algorithm. FIT and FIO are used to measure errors in the tracking step.

All the samples have been tested by the suggested algorithm to find sperm cells count and motility parameters for every sperm. In each sample performance metrics and the average time needed for processing one frame are calculated. We also compare the suggested algorithm with six other algorithms. The source code for them is written using Matlab. Five of them use the same segmentation method. This segmentation method is taken from Urbano et al.’s paper [12]. The algorithms are the Exact Nearest Neighbour extension to the JPDAF (ENN-JPDAF), the joint probabilistic data association filter (JPDAF), the probabilistic data association filter (PDAF), global nearest neighbour (GNN) and NN nearest neighbour algorithms. The sixth algorithm is the adaptive window average speed algorithm (AWAS) [17].

3.3 Results tables

\(w = 10\) for light fixing step. \(3 \times 3\) disk shaped structural element was used for doing the morphological operations, \((a = 1.5)\) and \((b = 1.25)\) for the collision detection step. For gating, \(G_0 = 30\) as the initial gate and \(b_1 = 0.55, b_2 = 0.25\) and \(b_3 = 0.2\) were used.
The results for the algorithm are in Table 4. As shown in the result table, eight of the 20 sample videos have no tracking errors (FIT = 0 and FIO = 0). False positive error sometime occurs because of sperm-like objects. They will not affect tracking results because they will be considered as dead sperm cells. False negative error happens when a sperm cell gets lost. This happens when the sperm cell collides with large non-sperm cells or when lots of sperm cells collide. Large objects are excluded in detection steps, and Kalman filter will be used to specify the location of the lost sperm.

As shown in Table 5 the improved B&B is many times faster than normal B&B. It takes only 24 milliseconds to solve 100 × 100 problem set. In general sperm cells will not exceed 100 per frame. So, the improved B&B can work fast enough to make the suggested tracking algorithm process more than 30 frames per second.

The suggested algorithm was compared with six other algorithms ENN-JPDAF, JPDAF, PDAF, GNN, NN, and AWAS [17]. As shown in the results in Tables 6 and 7, the suggested algorithm outperforms the other algorithms. The test is made on sample 1 and 2 videos. The suggested algorithm has zero tracking errors and just 4 false negative in sample 1. So, it was the best. ENN-JPDAF result is also very good with only 7 tracking errors. From results we can say that the suggested algorithm is the best in this test. Algorithms are ordered according to tracking error rates in descending order.

NN, GNN, AWAS, and the suggested algorithm are the fastest. PDAF, JPDAF and ENN-JPDAF are slower because they have high computational complexity. So, we can say that the suggested algorithm is robust, fast, and more accurate than many of the state-of-the-art algorithms. So, it can be used to work with recorded videos or live stream of images come from cameras.

4 | DISCUSSION

The proposed algorithm outperforms most of the previous algorithms because we solve most of the weak points of the previous algorithms. In the detection step the light of the image is fixed and depending on the area and the shape of the object a decision will be made to check whether it is a sperm cell or not. Kalman filter is not enough to estimate the new position of sperm cells specially when the sperm is moving randomly. To
solve this, a collision detection step is added. This step is very important because without it, we will have lots of lost sperm cells. JPDAF ellipse shaped gate which uses statistical distance is not suitable for non-progressive sperm cells because they move randomly, and they change their directions rapidly. The time between two video frames is not always equal. Sometimes the same frame is repeated twice when we use high frame rate to capture the video. So, the algorithms will be affected by video capturing conditions. The circular gate used by the suggested algorithm is more robust when we deal with a lot of fast randomly moving sperm cells and when we deal with bad video capturing conditions. The suggested algorithm has lower complexity in detection and tracking than PDAF and JPDAF, so it is working faster. As shown in Table 6 the suggested algorithm has zero tracking error and has just 4 false negative. Whereas JPDAF has 7 errors in tracking, 6 false negative, and 2 false positive.

5 | CONCLUSION

A new fast tracking algorithm is created. It is able to track tens of sperm cells in the recorded video or even from live camera stream. It is also able to deal with bad recording conditions and bad lighting. It fixes the uneven lighting images and detects sperm cells even if they collide depending on the shape and area of the sperm cell. It accurately tracks sperm cells even if they are moving randomly. It can find the correct track for each sperm cell even if they swim near each other or if their paths intersect with lower error rate than the other algorithms. It can calculate all the important motility parameters for every sperm. It has outperformed many of the previous tracking algorithms.

In future we will work on improving a model to estimate the new position of the sperm cell better than Kalman filter. We will do this by studying the various sperm motility models to improve the performance of the tracking algorithm.

ACKNOWLEDGEMENTS

The authors are grateful to the Faculty of Engineering at Ataturk University, and the laboratories of Computer Engineering Department, who helped them in recording and collecting the video samples.

ORCID

Ahmad Alhaj Alabdulla https://orcid.org/0000-0002-9686-8845

REFERENCES

1. Nafisi, V.R., Moradi, M.H., Nasr-Esfahani, M.H.: A template matching algorithm for sperm tracking and classification. Physiol. Meas. 26(5), 639 (2005).
2. Shi, L.Z., et al.: Computer-based tracking of single sperm. J. Biomed. Opt. 11(5), 054009 (2006).
3. Sorensen, L., et al.: Multi-object tracking of human spermatozoa. In: Medical Imaging 2008: Image Processing. International Society for Optics and Photonics, Bellingham (2008)
4. Ravanfar, M.R., Moradi, M.H.: Low contrast sperm detection and tracking by watershed algorithm and particle filter. In: 2011 18th Iranian Conference of Biomedical Engineering (ICBME) Tehran (2011)
5. Liu, J., et al.: Quantitative analysis of locomotive behavior of human sperm head and tail. IEEE Trans. Biomed. Eng. 60(2), 390–396 (2013).
6. Nurhadiyatna, A., et al.: Comparison and implementation of motion detection methods for sperm detection and tracking. In: 2014 International Symposium on Micro-NanoMechatronics and Human Science (MHS) (2014)
7. Imani, Y., et al.: A new method for multiple sperm cells tracking. Journal of medical signals and sensors 4(1), 35 (2014)
8. Jati, G., et al.: Human sperm tracking using particle swarm optimization combined with smoothing stochastic sampling on low frame rate video. In: 2015 International Symposium on Micro-NanoMechatronics and Human Science (MHS) (2015)
9. Beya, O., et al.: Automatic detection and tracking of animal sperm cells in microscopy images. In: 2015 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS) (2015)
10. Jan, G., et al.: Multi-sperm tracking using Hungarian Kalman filter on low frame rate video. In: 2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS) (2016)
11. Hesar, H.D., et al.: Multiple sperm tracking in microscopic videos using modified Gmm-Phd filter. Machine Vision and Applications 29(3), 433–451 (2017).
12. Urbano, L.F., et al.: Automatic tracking and motility analysis of human sperm in time-lapse images. IEEE Trans Med Imaging 36(3), 792–801 (2017).
13. Arasteh, A., Vosoughi Vali, B., Salman Yazdi, R.: Multi-target tracking of human spermatozoa in phase-contrast microscopy image sequences using a hybrid dynamic bayesian network. Sci. Rep. 8(1), 5068 (2018).
14. Imabuchi, F.B., et al.: Objective evaluation of Ram and Buek sperm motility by using a novel sperm tracker software. Reproduction 156(1), 11–21 (2018).
15. Masdiyasa, I.G.S., Purnama, I.K.E., Purnomo, M.H.: A new method to improve movement tracking of human sperms. IJENG International Journal of Computer Science 45(4), (2018)
16. Wang, C., et al.: Tracking of human sperm in time-lapse images. In: 2018 IEEE 3rd International Conference on Signal and Image Processing (ICSIP) (2018)
17. Mostajer Kheirkhah, F., Sadegh Mohammadi, H.R., Shahverdi, A.: Efficient and robust segmentation and tracking of sperm cells in microscopic image sequences. IET Comput. Vision 13(5), 489–499 (2019)
18. Qi, S., et al.: A sperm cell tracking recognition and classification method. In: 2019 International Conference on Systems, Signals and Image Processing (IWSSIP) (2019)
19. Suzuki, S.: Topological structural analysis of digitized binary images by border following. Computer Vision, Graphics, and Image Processing 30(1), 32–46 (1985).
20. Sklansky, J.: Finding the convex hull of a simple polygon. Pattern Recognit. Lett. 1(2), 79–83 (1982)
21. Genovesio, A., Belhassine, Z., Olivo-Marin, J.-C.: Adaptive gating in Gaussian Bayesian multi-target tracking. In: 2004 International Conference on Image Processing. 2004. ICIP’04. IEEE (2004).
22. Smith, K., et al.: Evaluating multi-object tracking. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)-Workshops. IEEE (2005)

How to cite this article: Alabdulla, A.A., Ha¸sılo˘glu, A., Aksu, E.H.: A robust sperm cell tracking algorithm using uneven lighting image fixing and improved branch and bound algorithm. IET Image Process. 15, 2068–2079 (2021). https://doi.org/10.1049/ipr2.12178