



Assignment of bachelor's thesis

Title:	Artificial Intelligence based Detection and Tracking of Sperm Cells
Student:	Jakub Hořenín
Supervisor:	Ing. Miroslav Čepek, Ph.D.
Study program:	Informatics
Branch / specialization:	Knowledge Engineering
Department:	Department of Applied Mathematics
Validity:	until the end of summer semester 2024/2025

Instructions

The University of Twente (UT) and the University of Waterloo (UW) are working on research into sperm bundles. This thesis aims to help the UT and UW teams understand the video recordings of sperm cell movements and speed up the analysis. The important parameters of the cell movements are the actual path, speed, interaction between individual cells, and whether the cells are moving individually or in a bundle. Currently, used Computer-Assisted Sperm Analysis (CASA) systems do not distinguish between single sperm cells and sperm cell bundles, leaving this work up to the researcher. This thesis aims to build a machine learning-based solution to identify individual sperm cells and bundles, identify the paths, and estimate the velocity. The solution will take a step-by-step approach, with each step using available data. The video recordings will be provided by the UT and UW teams.

Individual steps:

1. Conduct a literature review and review the current state of the art.
2. Use a semantic segmentation algorithm to identify objects of interest in the frame.
3. Create a classifier to automatically identify whether the object of interest is either an individual sperm cell, a group of nearby cells, a sperm cell bundle, or noise in the frame.
4. Identify the path of individual sperm cells or groups with the help of Kalman filters [1]. The possible alternative approaches are described in [2][3].
5. Calculate the speed of the cell or groups. Compare your approach to [4].



6. Document the accuracy of individual steps and the whole solution.
7. Create a web application to implement and demonstrate the proposed approach.

Literature:

- [1] Q. Li, R. Li, K. Ji and W. Dai, "Kalman Filter and Its Application," 2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS), Tianjin, China, 2015, pp. 74-77, doi: 10.1109/ICINIS.2015.35. <https://doi.org/10.1109/ICINIS.2015.35>
- [2] X. Rong Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," in IEEE Transactions on Aerospace and Electronic Systems, vol. 39, no. 4, pp. 1333-1364, Oct. 2003, doi: 10.1109/TAES.2003.1261132. <https://doi.org/10.1109/TAES.2003.1261132>
- [3] Choi, Jw., Alkhoury, L., Urbano, L.F. et al. An assessment tool for computer-assisted semen analysis (CASA) algorithms. Sci Rep 12, 16830 (2022). <https://doi.org/10.1038/s41598-022-20943-9>
- [4] OKUMUŞ, F., KOCAMAZ, F., & ÖZGÜR, M. E. (2021). Using polynomial modeling for calculation of quality parameters in computer assisted sperm analysis. Computer Science, 6(3), 152-165. <https://doi.org/10.53070/bbd.999296>

Bachelor's thesis

**ARTIFICIAL
INTELLIGENCE BASED
DETECTION AND
TRACKING OF SPERM
CELLS**

Jakub Hořenín

Faculty of Information Technology
Department of Applied Mathematic
Supervisor: Ing. Miroslav Čepěk, Ph.D.
May 15, 2024

Czech Technical University in Prague

Faculty of Information Technology

© 2024 Jakub Hořenín. All rights reserved.

This thesis is school work as defined by Copyright Act of the Czech Republic. It has been submitted at Czech Technical University in Prague, Faculty of Information Technology. The thesis is protected by the Copyright Act and its usage without author's permission is prohibited (with exceptions defined by the Copyright Act).

Citation of this thesis: Hořenín Jakub. *Artificial Intelligence based Detection and Tracking of Sperm Cells*. Bachelor's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2024.

Contents

Acknowledgments	vi
Declaration	vii
Abstract	viii
Seznam zkratek	x
1 Introduction	1
2 Theory	2
2.1 Spermatozoa Introduction	2
2.2 Research Context	3
2.2.1 Research of Sperm Cell Bundles	3
2.2.2 Sperm Cells in Nanorobotics	3
2.3 Computer Assisted Sperm Analysis Systems	4
2.4 Sperm Cell Velocity Calculation	5
2.5 Bundle Formation	5
2.6 Technologies	6
2.6.1 You Only Look Once (YOLO)	6
2.6.2 Real-ESRGAN	6
2.6.3 Kalman Filter	7
2.6.4 Residual Neural Network (ResNet)	9
2.6.5 Thresholding	9
2.6.6 Template Matching	10
2.6.7 Background Subtraction (Median Filter)	10
2.6.8 Dynamic Time Warping (DTW)	11
2.6.9 Optimal Subpattern Assignment Metric (OSPA)	12
2.6.10 Morphology	12
3 Implementation	14
3.1 Implementation Using YOLO for Detection	14
3.1.1 YOLO Based Approach	14
3.1.2 Video Restoration	15
3.1.3 Sperm Cell Detection	15
3.1.4 Sperm Cell Path Reconstruction	15
3.2 Approach Using Background Subtraction for Detection	18
3.2.1 Algorithm	18
3.2.2 Sperm Cell Detection (Image Segmentation)	19
3.2.3 Path Reconstruction	21
3.2.4 Bounding Box Classification (ResNet)	23
3.2.5 Velocity Calculation	25
3.2.6 Final Analysis	26

4	Testing/Evaluation	28
4.1	Testing of Approach Utilizing YOLO	28
4.1.1	Duplicity Detection by YOLO	28
4.1.2	Missed Detections	28
4.2	Tests of Approach Using Background Subtraction	29
4.2.1	Path Estimation Analysis and Tests	29
4.2.2	Sperm Cell Detection Accuracy	31
4.2.3	Classification Accuracy	31
4.2.4	Velocity Accuracy	31
5	User Interface	34
6	Discussion	37
6.1	Final Approach Selection	37
6.2	Program Use	38
6.3	Future Development	38
7	Conclusion	39
A	Additional Video Analysis Results	41
	Contents of the attached media	50

List of Figures

2.1	Spermatozoa examples	2
2.2	Sperm cell movement	3
2.3	MagnetoSperm	4
2.4	Sperm Velocity Parameters	6
2.5	Differences between analytical display, linearization in EKF, and UT in UKF [36]	9
2.6	Dynamic time warping illustration [46]	11
2.7	OSPA closest to reality problem [48]	12
2.8	Example of an image after erosion and dilation	13
3.1	Real-ESRGAN video enhancement	15
3.2	Sperm cell position determination	17
3.3	Video background calculated using Median Filter	19
3.4	Step-by-step process of sperm cell detection	20
3.5	Sperm bundle reflecting light into the microscope lens	21
3.6	Paths of all detected sperm cells	22
3.7	Comparison of paths reconstructed using different R_t	23
3.8	Examples of each classified class	24
3.9	Bounding box extension demonstration	25
4.1	Tests of path reconstruction accuracy	30
4.2	Comparing path reconstruction using different target tracking approaches [54]	32
4.3	Test of path reconstruction accuracy	33
4.4	Compression of calculated velocities	33
5.1	Web application layout	34
5.2	Web application results display	35
5.3	Web application appearance while processing videos	36
A.1	Video where bundle formation has occurred and has been flagged	41
A.2	High clutter and high sperm cell concentration in sperm cell video analysis	42
A.3	Full User interface with high density video and CSV output	43
A.4	Collision path distortion illustration	44

List of Tables

3.1	Object classification confusion matrix of the first trained model	24
3.2	Object classification confusion matrix of the final trained model	25

4.1	Duplicates counted	28
4.2	number of miss detections out of 400	29
4.3	DTW Methods for Different Paths	31
4.4	Calculated differences	33

List of code listings

Primarily, I would like to thank my supervisor, Ing. Miroslav Čepek, Ph.D., for his help while working on this thesis. His assistance in understanding certain topics and suggestions for improvements were invaluable. Additionally, I would like to thank Mgr. Alexander Kovalenko, Ph.D., who helped during the introduction to this topic and later assisted me with writing the paper upon which this thesis is based. His collaboration with both teams at the University of Waterloo and the University of Twente made this work possible. I express gratitude to both of those teams for providing the data that I used for training and testing my program. Special thanks go to Assistant Professor Dr. Veronika Magdanz, who lent me her expertise to help me understand the complexities of sperm cell analysis and assisted in verifying my results.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular the fact that the Czech Technical University in Prague has the right to conclude a licence agreement on the utilization of this thesis as a school work pursuant of Section 60 (1) of the Act.

This work was supported by the Student Summer Research Program 2023 of FIT CTU in Prague.

In Prague on May 15, 2024

Abstract

This thesis introduces a novel machine learning-based methodology for automated detection and tracking of sperm cells within microscopic video recordings, aiming to elucidate the dynamics and motion patterns of both individual sperm cells and sperm cell bundles. At first, the method identifies sperm cells across successive frames within a video sequence, facilitating the reconstruction of each cell's trajectory over time. Subsequently, I introduce a classification algorithm that distinguishes between single sperm cells, clusters of adjacent cells, sperm cell bundles, and clutter, addressing a gap in existing methodologies. Finally, I employ three conventional metrics for velocity assessment: Straight Line Velocity (VSL), Average Path Velocity (VAP), and Curvilinear velocity (VCL), to quantify the movement speed of both individual sperm cells and bundles. The approach represents a significant advancement in the automated analysis of sperm motility and aggregation phenomena, providing a robust tool for researchers to study sperm behavior with enhanced accuracy and efficiency. A web-based user interface has been created, and the latest version of the program utilizing this methodology is publicly available at https://apps.datalab.fit.cvut.cz/sperm_tracking/ with source code publicly available at gitlab: https://gitlab.fit.cvut.cz/horenjak/sperm_cell_tracking_app/.

Keywords Sperm cell tracking, Sperm Motility, Bundle Detection, Video analysis, ResNet, Kalman filter

Abstrakt

Tato práce představuje novou metodiku založenou na strojovém učení pro automatickou detekci a sledování spermíí v rámci mikroskopických videozáznamů, jejímž cílem je objasnit dynamiku a vzorce pohybu jednotlivých spermíí a shluků spermíí. Metoda nejprve identifikuje jednotlivé spermie v po sobě jdoucích snímcích videosekvence, což usnadňuje rekonstrukci trajektorie každé buňky v čase. Následně nasazji klasifikační algoritmus, který rozlišuje mezi jednotlivými spermíími, shluky sousedících buněk, shluků spermíí a znečištěním, čímž řeším mezeru ve stávajících metodikách. Nakonec používám tři konvenční metriky pro hodnocení rychlosti: Straight Line Velocity (VSL), Average Path Velocity (VAP), and Curvilinear velocity (VCL), aby bylo možné kvantifikovat rychlost pohybu jednotlivých spermíí a shluků spermíí. Tento přístup představuje významný pokrok v automatické analýze pohyblivosti a agregačních jevů spermíí a poskytuje výzkumníkům robustní nástroj pro studium chování spermíí s vyšší přesností a efektivitou. Bylo vytvořeno webové uživatelské rozhraní a nejnovější verze programu využívajícího tuto metodiku je veřejně dostupná na adrese https://apps.datalab.fit.cvut.cz/sperm_tracking/, přičemž zdrojový kód je k dispozici na gitlabu: https://gitlab.fit.cvut.cz/horenjak/sperm_cell_tracking_app/.

Klíčová slova Sledování spermií, Pohyblivost spermií, Detekce shluků, Analýza videa, ResNet, Kalmanův filtr

Seznam zkratek

AI	Artificial Intelligence
ALH	Lateral Head Displacement
CASA	Computer Assisted Sperm Analysis
CSV	Comma-Separated Values
DTW	Dynamic Time Warping
EKF	Extended Kalman Filter
ESRGAN	Enhanced Super-Resolution Generative Adversarial Networks
FPS	Frames Per Second
GNN	Global Nearest Neighbor
HR	High-Resolution
JPDA	Joint Probabilistic Data Association
LIN	Linearity
ML	Machine Learning
MF	Median Filter
NN	Nearest-Neighbor
OMAT	Optimal Mass Transfer
OSPA	Optimal Subpattern Assignment
PDA	Probabilistic Data Association
ResNet	Residual Neural Network
SR	Super-Resolution
SRGAN	Super-Resolution Generative Adversarial Networks
STR	Straightness
TM_CCOEFF	Template Matching Correlation Coefficient
TM_CCOEFF_NORMED	Template Matching Correlation Coefficient Normed
UI	User Interface
UKF	Unscented Kalman Filter
UT	University of Twente
UW	University of Waterloo
VAP	Average Path Velocity
VCL	Curvilinear Velocity
VSL	Straight-Line Velocity
VyLet	Research Summer at FIT
YOLO	You Only Look Once

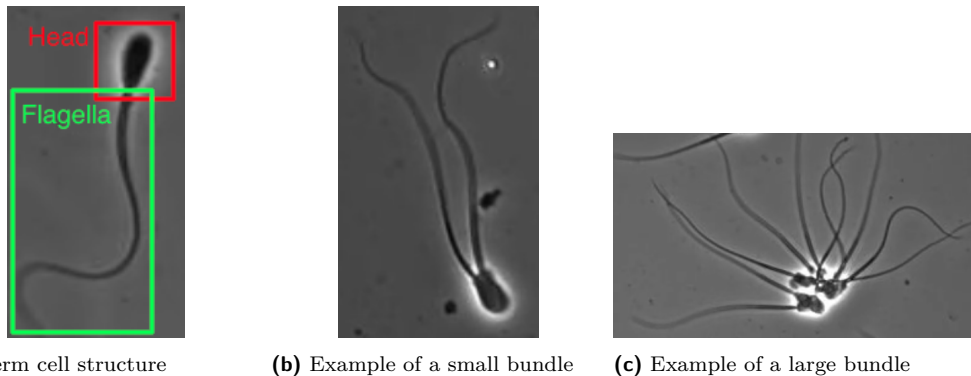
Chapter 2

Theory

= In the initial chapter of this thesis, I will lay the groundwork for the reader to understand the background of sperm cell research, for which this tool is being developed. I will present the current state-of-the-art tools used for sperm cell analysis and introduce the technologies used in the development.

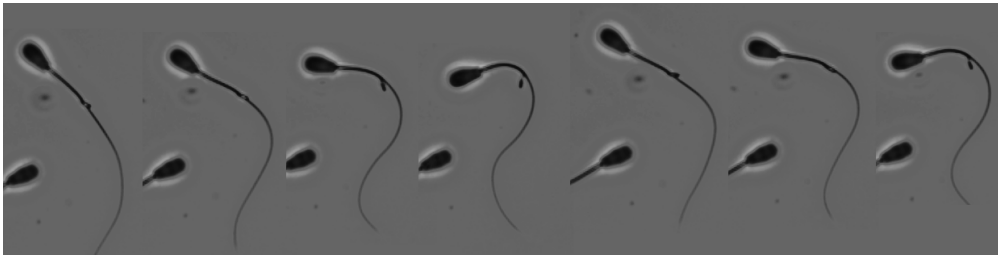
2.1 Spermatozoa Introduction

My program is developed primarily to analyze bull spermatozoa samples. An example of a single bull sperm cell can be seen in Figure 2.1a. Bull sperm cells tend to create bundles to improve their motility capabilities. These bundles can vary in size; an example of a small bundle consisting of two sperm cells can be seen in Figure 2.1b. In Figure 2.1c, you can see an example of a larger bundle consisting of 10 sperm cells. One requirement for my program was for it to be able to distinguish between a single sperm cell and a sperm bundle.



■ **Figure 2.1** In this figure, you can see the structure of a single sperm cell, which consists of a head and flagella. Additionally, examples of both small and large sperm cell bundles are presented.

Worth mentioning is also the unique way sperm cells move, in a wave-like pattern where the sperm cell head moves from side to side. This movement is illustrated in Figure 2.2. This specific type of movement pattern is the reason for a different way of calculating sperm cell velocity, as discussed in Section 2.4, compared to the methods used with other common organisms like humans, dogs, or moving objects such as cars and bikes.



■ **Figure 2.2** Frame sequence illustrating sperm cell movement.

2.2 Research Context

In this section, I will introduce two avenues of sperm cell research that the Canadian team at the University of Waterloo (UW), under the leadership of Mrs. Veronika Magdanz, and the Dutch team at the University of Twente (UT), under the leadership of Mr. Islam Khalil, are working on, for which this program is being developed to aid with the research.

2.2.1 Research of Sperm Cell Bundles

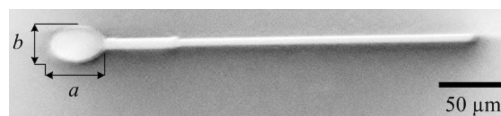
One avenue of spermatozoa research involves observing the process of sperm cells assembling into bundles. These bundles help sperm cells improve their swimming efficacy compared to individual cells [6]. This part of spermatozoa research is still underexplored, and modern machine learning (ML) tools could significantly aid in this field of research. The types of bundles that sperm cells form vary from organism to organism. In this thesis, I will be discussing a tool that helps with the analysis of sperm cell bundles of bovine spermatozoa. It is noted that other organisms' spermatozoa can also create bundles but with different types of formations [14, 9, 7]. When it comes to sperm cell bundles, research is focused on how they form [6], their motility capabilities [5], and how the surrounding environment can affect sperm cell bundle formation [7]. The first application of the developed tool in this thesis should be to help analyze videos of sperm cells in an enhanced environment that supports bundle formation and controlled samples, and to assess whether more bundles have formed in the enhanced environment and what the motility capabilities of single sperm cells and sperm cell bundles are in the enhanced sample compared to those in the control samples.

2.2.2 Sperm Cells in Nanorobotics

The Canadian team and Dutch team with whom I have been working, and who have been helping me develop my program, require this specific type of tool for sperm cell and sperm cell bundle analysis to aid their research, which aims to utilize sperm cells in nanorobotics. It has been reported that sperm cells were used as blueprints [11] or templates [12] for microrobots that can be used for targeted therapy. The reason why spermatozoa are being considered as a model for creating microrobots or even as components of microrobots is *due to their strong flagellar driving source and ability to take up drugs*[15]. One way that researchers are trying to utilize the knowledge of sperm cell movement and morphology is by creating MagnetoSperm, as developed by researchers at the University of Twente, as shown in Figure 2.3. Subsequent research has been done to compare the movement of motile and magnetically actuated sperm cells to better understand the movement differences. Magnetically actuated sperm cells are dead bull sperm cells that have been coated with nanoparticle aggregates [15, 16]. This helps reveal insights into flagellar waveforms that will aid in the design of efficient microswimmers.

Research has also been conducted to find viable options for the navigation of these microswimmers. One viable technique is using deep reinforcement learning [17]. This study shows that this technique could be used to train swimmers that are able to navigate through complex environments and perform difficult maneuvers.

In summary, understanding the motion of living cells can serve as inspiration for microrobot design [13]. Also, understanding sperm cell motion and bundle formation refers to the collective behavior of these microorganisms. By observing and analyzing the collective behaviors of cells, it is possible to glean significant statistical data that can inform and refine our understanding of the biosphere at the microscopic level, as the concept of swarm intelligence emerges from collective behavior. Unraveling these intricacies could shed light on the decentralized, self-organized systems that can lead to the emergence of complex behavior from simple individual entities.



■ **Figure 2.3** *MagnetoSperm* is a structure resembling a sperm-cell with a magnetic head and a flexible tail of $42\ \mu\text{m}$ and $280\ \mu\text{m}$ in length. *MagnetoSperm* can be steered and propelled using weak oscillating magnetic fields [11].

2.3 Computer Assisted Sperm Analysis Systems

CASA (Computer Assisted Sperm Analysis) image analysis systems allow for the accurate, repetitive, automatic, and efficient assessment of sperm motility, concentration, morphology, and other characteristics. These systems have revolutionized the study and clinical assessment of spermatozoa by utilizing advanced imaging and computational algorithms to automate the process of tracking and analyzing sperm movement. CASA technologies provide objective, repeatable, and high-throughput analysis, overcoming the limitations of manual microscopy methods. Prominent systems include the Hamilton-Thorne analyzer [18], Sperm Class Analyzer (SCA) [19], and Microptic's SCA system [20], among others. These systems vary in their imaging techniques, analysis algorithms, and the range of motility parameters they measure.

The analysis of sperm motility is pivotal in assessing male fertility and understanding reproductive behaviors [21, 22]. It aids in diagnosing male infertility, guiding clinical decisions in assisted reproductive technologies [23, 3], and evaluating the effects of environmental and lifestyle factors on sperm health. The motility of sperm cells is indicative of their viability and fertilizing potential, making it a fundamental parameter in both clinical and research settings.

CASA systems employ a combination of video microscopy and digital image processing to capture and analyze sperm movement. The core steps involve:

- Capturing sequences of video frames by a high-resolution camera, documenting sperm movement over time.
- Identifying sperm cells within the video frames.
- Tracking sperm cells across frames, employing methods such as particle tracking or optical flow to calculate their trajectories.
- Analyzing trajectories to compute various motility parameters, including but not limited to average path velocity (VAP), straight-line velocity (VSL), and curvilinear velocity (VCL) (see Section 2.4).

CASA systems have provided invaluable tools for the detailed analysis of sperm motility, contributing to advances in reproductive medicine and biological research. Despite their benefits,

ongoing development is necessary to address existing limitations, enhance detection algorithms, and ensure consistency across different platforms. Therefore, machine learning has vast potential to bring additional value to CASA systems by improving analysis accuracy and developing models that more closely mimic the natural environment of sperm migration.

2.4 Sperm Cell Velocity Calculation

One way to assess sperm quality is to look at sperm cell velocity. It is standard for modern CASA systems to calculate straight-line velocity (VSL) 2.2, curvilinear velocity (VCL) 2.1, and average path velocity (VAP) 2.3 (see Figure 2.4). With these, we can calculate linearity (LIN) 2.5, which tells us the linearity of the curvilinear path, straightness (STR) 2.6, which tells us the linearity of the average path, and magnitude of fluctuation (WOB) 2.7. All of these help assess how the sperm cell is behaving.

$$VCL = \frac{\sum_{i=1}^M (\sqrt{(x_{j+1} - x_j)^2 + (y_{j+1} - y_j)^2})}{(M - 1)\Delta t} \quad (2.1)$$

$$VSL = \frac{(\sqrt{(x_M - x_1)^2 + (y_M - y_1)^2})}{(M - 1)\Delta t} \quad (2.2)$$

$$VAP = \frac{\sum_{i=1}^{M-1} (\sqrt{(x_{j+1} - x_j)^2 + (y_{j+1} - y_j)^2})}{(M - 1)\Delta t} \quad (2.3)$$

$$x_k = \frac{1}{2 * n + 1} \sum_{i=k-n}^{k+n} x_i \quad (2.4)$$

$$LIN = \frac{VSL}{VCL} \quad (2.5)$$

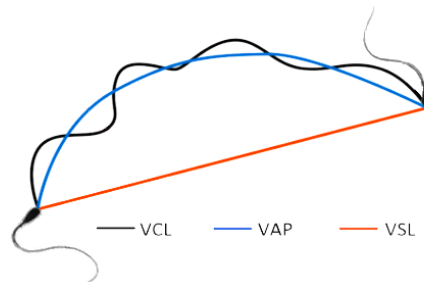
$$STR = \frac{VSL}{VAP} \quad (2.6)$$

$$WOB = \frac{VAP}{VCL} \quad (2.7)$$

[24, 25, 26]

2.5 Bundle Formation

There is a lot to be analyzed when it comes to spermatozoa. Modern CASA systems utilize modern machine learning (ML) tools to significantly help with analysis by offering a wide range of ways to analyze sperm cell samples for different use cases [18, 19, 1, 2, 3]2.3. However, there are still under-explored fields when it comes to sperm cell analysis, where ML can bring unquestionable benefits. One such field is the automated exploration and detection of sperm bundles [4, 5, 6]. Sperm bundling occurs due to the combined forces of hydrodynamics and cell adhesion. Sperm bundle formation can be observed in various animals ranging from bovines [7]



■ **Figure 2.4** Sperm Velocity Parameters: straight-line velocity (VSL), curvilinear velocity (VCL), and average path velocity (VAP)

to ants [8], echidnas, and others [9, 7, 5], and is often associated with higher sperm motility [7, 6, 10] and higher reproduction rates [5]. The type of formation that sperm cells will create depends on the type of sperm. Bull sperm, which our research focuses on, creates bundles. In contrast, wood mouse sperm tends to create trains instead of bundles [14].

Thanks to the superior capabilities of sperm cell bundles, detecting them and quantifying their formation gives a direct insight into the quality of the explored material, which is extremely important in various fields such as assisted reproductive technologies [5], including artificial insemination and *in-vitro* fertilization.

2.6 Technologies

In this section, I will briefly introduce the main technologies that I have utilized in developing my system for sperm cell analysis.

2.6.1 You Only Look Once (YOLO)

You Only Look Once (YOLO) is an algorithm used for object detection. It is a small-sized model that performs fast calculations. The name comes from the fact that the image for which the detection is being performed needs to be passed only once to the model. That is also where the speed of this model comes from. *A single convolutional network simultaneously predicts multiple bounding boxes and class probabilities for those boxes* [27]. YOLO has some significant advantages over other models that stem from its architecture. Its high speed of prediction is notable. Unlike other models, YOLO looks at the whole image when performing object detection, which can provide helpful context when predicting classes. YOLO learns generalized features, making it more flexible, meaning it outperforms other models when predicting over different datasets than the one it was trained on. This flexibility also enables the model to predict well when it receives unexpected input, even from the same dataset. However, YOLO does not perform as well when detecting objects close to each other or objects that are small.[28, 27]

2.6.2 Real-ESRGAN

Super-Resolution Generative Adversarial Networks (SRGANs) are used to obtain high-resolution (HR) images from their low-resolution (LR) counterparts. This is accomplished using a neural network that estimates the HR image given the LR image, a task known as Super-Resolution (SR). SRGAN is built on ResNet, allowing it not to rely on minimizing mean square error, which

has been proven inadequate when it comes to SR results. This is because it is unable to capture perceptually relevant differences. Additionally, SRGAN is optimized using perceptual loss, a method of quantifying the difference between two images based on their perceptual similarity as perceived by humans. A discriminator is added to encourage solutions perceptually close to HR reference images. The discriminator's task is to classify examples as either real (from the domain) or fake (generated). The model is trained until the discriminator is fooled about half the time, indicating that the generator model is generating plausible examples.

Enhanced Super-Resolution Generative Adversarial Networks (ESRGANs) change the structure by adding and removing some layers. This makes the training of the model easier and increases capacity. The discriminator is improved to compare images not based on whether they are "real or fake" but on whether "one is more realistic than the other," and by changing the perceptual loss. All these changes result in the final HR image being more visually pleasing.

Real-ESRGAN, unlike previous models, attempts to work with LR images that have undergone unknown and complex degradations. These are degradations that cannot be modeled by a simple combination of blur, downsampling, noise, and JPEG compression. To achieve this, a new degradation process is created to better simulate real-world degradation. Changes to the discriminator are also made for it to be able to deal with more complex training outputs. [29, 30, 31]

2.6.3 Kalman Filter

The Kalman filter is a set of mathematical equations that provides an efficient computational means to estimate the state of a process, in a way that minimizes the mean of the squared error[32] taken in to account the measurements observed over time.

It attempts to estimate the next state x , which is operated by the equation:

$$x_t = A_t x_{t-1} + B_t u_t + w_t, \quad w_t \sim (0, Q_t) \quad (2.8)$$

with measurement y that is:

$$y_t = H_t x_t + \varepsilon_t, \quad \varepsilon_t \sim (0, R_t) \quad (2.9)$$

where w_t is the process noise and ε_t measurement noise and they are independent of each other. u_t is as optional control variable. A_t, B_t, H_t are matrices of corresponding dimensions.

Kalman filter algorithm has consist of three phases. Initialization here the initial state x_0 and P_0^+ the initial posteriori estimate error are set.

$$\text{Priori estimate error : } P_t^- = \mathbb{E} [(x - \hat{x}_t^-)(x - \hat{x}_t^-)^\top]. \quad (2.10)$$

$$\text{Posteriori estimate error : } P_t^+ = \mathbb{E} [(x - \hat{x}_t^+)(x - \hat{x}_t^+)^\top]. \quad (2.11)$$

\hat{x}_t^- is priori state estimate at step t given knowledge of the process prior to step t and \hat{x}_t^+ is the posteriori state estimate at step t given measurement y_t .

The next phase is estimation. Estimation is performed at each step t :

$$\hat{x}_t^- = A_t \hat{x}_{t-1}^+ + B_t u_t, \quad (2.12)$$

$$P_t^- = A_t P_{t-1}^+ A_t^\top + Q_t. \quad (2.13)$$

If it was in the same step t measurement obtained, the correction phase is performed. First, the gain matrix is computed. Then, the estimate is updated based on the obtained measurement y_t

and then the posterior error covariance is calculated.

$$K_t = P_t^- H_t^\top (H_t P_t^- H_t^\top + R_t)^{-1}, \quad (2.14)$$

$$\hat{x}_t^+ = \hat{x}_t^- + K_t (y_t - H_t \hat{x}_t^-), \quad (2.15)$$

$$P_t^+ = (I - K_t H_t) P_t^- (I - K_t H_t)^\top + K_t R_t K_t^\top. \quad (2.16)$$

Matrix K_t is the gain that minimizes the a posteriori error covariance. From that, the posteriori state estimate \hat{x}_t^+ is computed as a combination of the priori estimate \hat{x}_t^- and the weighted (K_t) difference between an actual measurement y_t and a measurement prediction $H_t \hat{x}_t^-$ [32].

In practice, the measurement noise covariance matrix R_t should be calculated by taking a sample of measurement data and calculating their variance. With the process noise matrix Q_t it is not so simple as it is not usually possible to observe the process that is being estimated.

The standard Kalman filter is able to only work with processes that are linear. In the real world, there are almost none. To deal with processes that are non-linear, the Extended Kalman filter and the Unscented Kalman filter have been developed. The state model changes to the non-linear form:

$$x_t = f_t(x_{t-1}, u_t) + w_t, \quad (2.17)$$

$$y_t = h_t(x_t) + \varepsilon_t \quad (2.18)$$

where f_t and h_t are non-linear functions. It needs to be noted that neither the Extended Kalman filter nor the Unscented Kalman filter are optimal in the sense of mean square error minimization. [33, 32, 34]

2.6.3.1 Extended Kalman Filter (EKF)

EKF deals with a non-linear model by performing linearization using partial derivatives. The EKF algorithm then has two extra phases compared to the original Kalman filter: prediction linearization and measurement linearization.

$$F_t = f'_t(\hat{x}_{t-1}^+) \quad (2.19)$$

and correction linearization

$$H_t = h'_t(\hat{x}_t^-). \quad (2.20)$$

The rest of the equations change accordingly, with \hat{x}_t^- being the priori state estimate and \hat{x}_t^+ being the posteriori state estimate, obtained directly from the functions of the state model and not using the partial derivation F_t, H_t .

$$\hat{x}_t^- = f_t(\hat{x}_{t-1}^+, u_t), \quad (2.21)$$

$$P_t^- = F_t P_{t-1}^+ F_t^\top + Q_t, \quad (2.22)$$

$$K_t = P_t^- H_t^\top (H_t P_t^- H_t^\top + R_t)^{-1}, \quad (2.23)$$

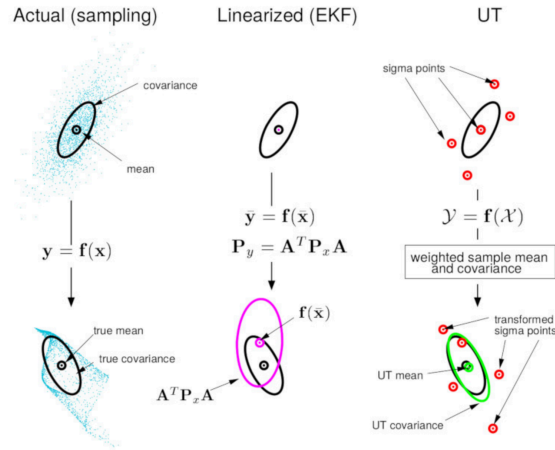
$$\hat{x}_t^+ = \hat{x}_t^- + K_t [y_t - h_t(\hat{x}_t^-)], \quad (2.24)$$

$$P_t^+ = (I - K_t H_t) P_t^- (I - K_t H_t)^\top + K_t R_t K_t^\top. \quad (2.25)$$

[32, 34, 33]

2.6.3.2 Unscented Kalman Filter (UKF)

Unscented Kalman Filter (UKF) is another way to modify the Kalman Filter to work on non-linear models. Unlike EKF, this time, to get from a non-linear representation to a linear one,



■ **Figure 2.5** Differences between analytical display, linearization in EKF, and UT in UKF [36]

unscented transformation (UT) is used. What this transformation does is that it approximates the distribution of state variables with points that ideally maintain the first two moments. The illustration of this process and its comparison to EKF can be seen in Figure 2.5. It has been proven that UKF is giving similar or better results for non-linear problems compared to EKF [35]. [35, 36]

2.6.4 Residual Neural Network (ResNet)

Residual neural networks are popular for image classification [37, 38]. Residual neural network (ResNet) is a modified standard neural network extended by residual mapping. This modification allows the network to take full advantage of the increased depth of a network, boosting its accuracy. Unlike a standard neural network that faces the degradation problem, causing it to suffer from higher training error when the depth of the neural network is increased. Residual mapping means that given input x , a standard neural network is trying to learn mapping $\mathcal{H}(x)$. In ResNet, layers attempt to fit a different mapping $\mathcal{F}(x) = \mathcal{H}(x) - x$, representing the difference between the desired mapping and its input. The original mapping is then $\mathcal{H}(x) = \mathcal{F}(x) + x$. The advantage comes from it being easier to optimize the residual mapping than the original one. [39, 40]

2.6.5 Thresholding

Thresholding is one of the core techniques in digital image processing. A threshold is set, and the grayscale image is transformed into a binary image. This is mainly done to extract objects from their background. One of the techniques for selecting the ideal threshold to separate foreground from background is Otsu's method. This method calculates the grayscale image histogram, and ideally, there will be two high peaks, one representing the foreground and the other the background of an image. In that scenario, the ideal threshold is the deepest point between these two peaks. However, in the real world, locating this valley is not so simple. Most images do not have two distinct peaks. If the peaks are uneven in height or the valley between them is flat or filled with noise, finding the ideal threshold is not a simple task. The optimal threshold k is one that maximizes the between-class variance σ_B^2 .

$$\sigma_B^2 = \omega_0 \omega_1 (\mu_1 - \mu_0)^2 \quad (2.26)$$

All pixels of an image are represented in L gray levels $[1, \dots, L]$. The number of pixels at each level is represented by n_i , and the total number of pixels is $N = n_1 + n_2 + \dots + n_L$. The gray-level histogram is then normalized and in the form of a probability distribution:

$$p_i = n_i/N, \quad p_i \geq 0, \quad \sum_{i=1}^L p_i = 1 \quad (2.27)$$

Pixels are then dichotomized in to two classes C_0 and C_1 by threshold k .

$$\begin{aligned} C_0 &= [1, \dots, k], \\ C_1 &= [k + 1, \dots, L] \end{aligned}$$

then:

$$\omega_0 = \sum_{i=1}^k p_i, \quad (2.28)$$

$$\omega_1 = \sum_{i=k+1}^L p_i, \quad (2.29)$$

$$\mu_0 = \sum_{i=1}^k ip_i/\omega_0, \quad (2.30)$$

$$\mu_1 = \sum_{i=k+1}^L ip_i/\omega_1. \quad (2.31)$$

[41]

2.6.6 Template Matching

The goal of template matching is to take a given template and find it in a larger image. This task is not as easy as all instances of the template in this larger image do not have to be exact. Its different instances can be corrupted by additional noise, they can be captured from a different angle, illumination can be different, etc. Template matching can be done by using ordinal correlation measures. The template image is systematically compared to sections of the larger image using a selected comparison method. This approach is unable to deal with the template being rotated in the large image. In this work, I tried Pearson correlation coefficient (TM_CCOEFF) and also its normed version (TM_CCOEFF_NORMED) that should perform better when the template and its instances in the large image are of different scales. [42, 43]

2.6.7 Background Subtraction (Median Filter)

The process of identifying moving objects within a video is a common problem in video analysis. One of the techniques vastly used for this is background subtraction. This technique compares each frame of a video to a model of the background; deviating pixels are considered the foreground (moving objects). There are several hurdles that background subtraction must deal with, like varying illumination of the video throughout the video sequence, changing shadows that moving objects are casting that should not be part of the foreground. Also, good background subtraction algorithms should also be able to deal with varying speeds of moving objects and them even stopping for periods of time. There are many background subtraction algorithms as discussed in [44]. Most of these algorithms follow four steps:

- Preprocessing: This part of the algorithm takes raw videos and prepares them for background subtraction.
- Background Modeling: Now, separate video frames are used to model the background.
- Foreground Detection: In this step, using the background model, foreground pixels are identified.
- Data Validation: In this step, received foreground is processed and false flags are removed.

One of the most simple but still robust and widely used techniques for background subtraction is Median Filter (MF). It is a non-recursive technique where each pixel of a background model is defined by the median value of all the frames in the buffer. One drawback of this technique is the requirement of maintaining the buffer. [45, 44]

2.6.8 Dynamic Time Warping (DTW)

Dynamic time warping (DTW) is a technique that finds the optimal alignment between two time-dependent series $X = (x_1, \dots, x_N)$ and $Y = (y_1, \dots, y_M)$ of length $N, M \in \mathbb{N}$ Figure 2.6, regardless of how they vary in time and speed. Let F be a feature space. Then $x_n, y_m \in F$ for $n \in [1 : N]$ and $m \in [1 : M]$. Then local cost measure is defined as:

$$c : F \times F \longrightarrow \mathbb{R}_0^+.$$

Here, $c(x, y)$ is small if x and y are similar and high otherwise. By evaluating all elements of sequences X and Y , a cost matrix $C \in \mathbb{R}^N \times M$ is obtained, defined by $C(n, m) = c(x_n, y_m)$. The goal then is to find the alignment of X and Y that gives the overall minimal cost. To achieve the minimal cost, a warping path is used. Every possible mapping of sequence X to Y can be represented as a warping path, and the goal is to find the cheapest one.

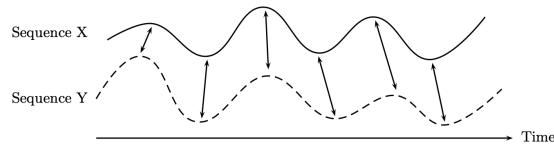
An (N, M) -warping path is a sequence $p = (p_1, \dots, p_L)$ with $p_l = (n_l, m_l) \in [1 : N] \times [1 : M]$ for $l \in [1 : L]$ satisfying three conditions:

1. Boundary condition: $p_1 = (1, 1)$ and $p_L = (N, M)$.
2. Monotonicity condition: $n_1 \leq n_2 \leq \dots \leq n_L$ and $m_1 \leq m_2 \leq \dots \leq m_L$.
3. Step size condition: $p_l + 1p_l \in (1, 0), (0, 1), (1, 1)$ for $l \in [1 : L]$.

. DTW is then defined as:

$$DTW(X, Y) = c_{p^*}(X, Y) = \min_{p \text{ is an } (N, M) \text{ - warping path,}} c_p(X, Y) = \sum_{l=0}^L Lc(x_{n_l}, y_{m_l})$$

where $c_p(X, Y)$ is the total cost of the warping path p between X and Y . [46, 47]

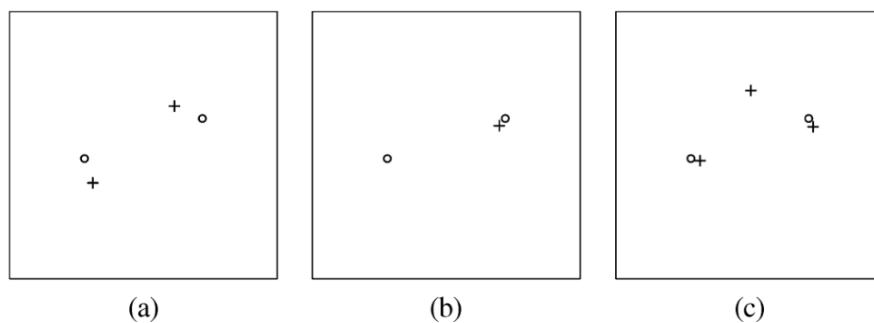


■ Figure 2.6 Dynamic time warping illustration [46]

2.6.9 Optimal Subpattern Assignment Metric (OSPA)

The Optimal Subpattern Assignment (OSPA) metric is used for evaluating the accuracy of multi-object tracking systems. Unlike when tracking a single object, determining the error between a reference quantity and its estimated value is not straightforward when tracking multiple objects. This is because the number of tracked objects can be unknown. Even when it is known, there are scenarios where it is difficult to decide which scenario should be considered more accurate. For example, which is more important: less accurate estimates or more accurate estimates with some missed detections. In Figure 2.7, you can see three scenarios that commonly occur, making it difficult to decide which one is closest to reality.

OSPA addresses these problems by measuring the distance between two finite sets of paths. The number of paths in each set can also change over time as objects appear or disappear. OSPA builds upon its predecessors, the Optimal Mass Transfer (OMAT) metric and the Hausdorff metric, aiming to solve some of the problems they encountered. Some of these problems include the inconsistency of OMAT in certain scenarios, and the geometry-dependent behavior of OMAT and the Hausdorff metric. Additionally, these metrics are undefined if the cardinality is zero, meaning that the distance $d(X, Y)$ is undefined if X or Y is zero. By addressing these issues, OSPA has become a standard metric for evaluating the results of multiple target tracking. [48, 49, 50, 51]



■ **Figure 2.7** Dots represent real targets and + represents estimates. This figure presents three scenarios that are not perfectly estimating the targets' positions. Each of them is wrong in a different way and represents scenarios that regularly happen when tracking targets, which OSPA needs to overcome and decide which one of these three scenarios is the most accurate and which is the least.

2.6.10 Morphology

Mathematical morphological operations are used for simplifying images while maintaining their core shape characteristics. The most common ones are image dilation and erosion. In Figure 2.8, you can see the effect on an image after performing erosion and subsequent dilation.

2.6.10.1 Dilation

One way to look at image dilation is by representing the image to be dilated as a set A and set B as the structuring element. Then dilation is the operation:

$$A \oplus B = \{c \in E^N | c = a + b \text{ for some } a \in A \text{ and } b \in B\} \quad (2.32)$$

Example:

$$\begin{aligned} A &= \{(0, 1), (1, 1), (2, 3), (3, 3)\} \\ B &= \{(1, 0)\} \\ A \oplus B &= \{(0, 1), (1, 1), (2, 1), (2, 3), (3, 3), (4, 3)\} \end{aligned}$$

2.6.10.2 Erosion

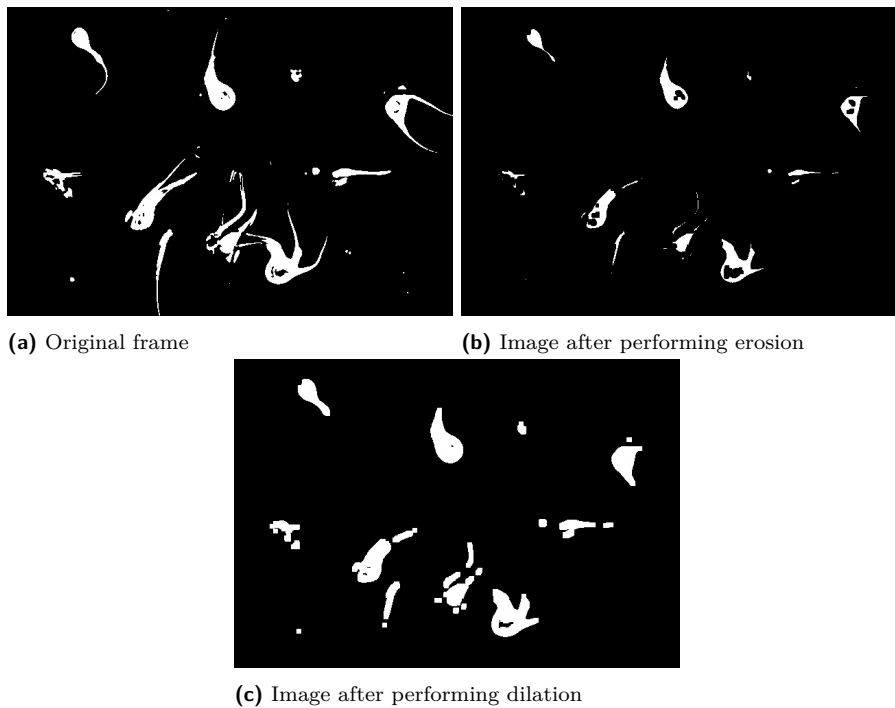
Erosion is morphologically dual to dilation. Again, let's have two sets A representing the image to be eroded and set B as the structuring element. The erosion of A by B is the set of elements x for which $x + b$ is in A :

$$A \ominus B = \{x \in E^N \mid x + b \in A \text{ for every } b \in B\} \quad (2.33)$$

Example:

$$\begin{aligned} A &= \{(0, 1), (1, 1), (2, 3), (3, 3)\} \\ B &= \{(1, 0)\} \\ A \ominus B &= \{(0, 1), (1, 3), (2, 3)\} \end{aligned}$$

[52]



■ **Figure 2.8** Example of an image after erosion and dilation

Implementation

In this chapter, I will discuss two approaches that differ significantly in how individual sperm cells are detected. The first approach utilizes the You Only Look Once (YOLO) algorithm (see Section 2.6.1), while the second approach employs background subtraction (see Section 2.6.7). I will then explain the selection process and rationale for choosing one approach over the other in Section 6.1.

3.1 Implementation Using YOLO for Detection

The first approach for sperm cell video analysis revolved around classification performed by YOLO. YOLO was chosen as it is a standard tool for object detection in modern tools. A secondary reason was that my colleague, Daniil Pastukhov, had already trained a model that he was using in his work, in which he was analyzing sperm cell flagella [53]. This neural network was used in the initial step of detecting all sperm cells in a video. The next step was to match sperm cells across all frames, achieved by calculating the bounding box overlap in subsequent frames. After that, each bounding box was analyzed to isolate the sperm cell within and identify its head. The head's position was then saved to render the path taken by the sperm cell, determining its direction of movement.

However, after several experiments, I abandoned this approach due to the results falling short of the expected precision. In many cases, the model was unable to detect sperm cells at all. Additionally, when detected, sperm cells were often identified multiple times, leading to challenges in bounding box matching. Also, this trained model was unable to distinguish between single sperm cells and sperm cell bundles, so to meet all the requirements, it would have to be retrained or a second classification would have to be implemented, slowing down the process of analysis significantly. Finally, the isolation of the head proved to be more challenging than initially anticipated.

With a deeper understanding of the problem, an alternative approach emerged. The only element that was ultimately retained from the initial approach was the bounding box overlap matching.

3.1.1 YOLO Based Approach

In this section, I describe the methodology of my first approach step by step. It must be noted that this approach was never completed, and to satisfy all the criteria of this thesis, more steps would have to be added. The implemented steps are as follows:



■ **Figure 3.1** Before and after using Real-ESRGAN to enhance video quality.

1. Video restoration: In this step, all videos underwent restoration using Real-ESRGAN to enhance their resolution and refine their quality. This was done to aid the neural network in the classification process (see Section 3.1.2).
2. Sperm cell detection: In this step, YOLO is utilized to detect sperm cells in each frame of a video (see Section 3.1.3).
3. Sperm path reconstruction: In this step, sperm cells detected by YOLO are matched across frames using bounding box overlap, with the support of retrieved sperm cell directional and location information (see Section 3.1.4).

These functionalities have never been implemented:

1. Sperm cell classification to distinguish between single sperm cells and sperm cell bundles.
2. Velocity calculation for each detected sperm cell.

3.1.2 Video Restoration

Prior to their submission to the YOLO model videos underwent restoration using Real-ESRGAN to enhance their resolution and refine their quality (see Figure 3.1). This process was undertaken to assist the neural network in effectively detecting sperm cells. All videos underwent mass restoration through a script before being subjected to analysis.

Initially, video enhancement was also applied in the second approach; however, it was eventually discontinued. The new neural network proved to be less sensitive to variations in video quality, and the extended time required for the restoration of each video was ultimately deemed not worth it, as enhancing one video can take up to several minutes.

3.1.3 Sperm Cell Detection

The whole video is given to the trained YOLO model that processes this video and returns an array containing all frames of the video, and for each frame, an array of all the detected objects with their corresponding bounding box and class. YOLO was trained for only one class, *sperm cell*, under this class were classified single sperm cells but also sperm cell bundles or multiple sperm cells crossing each other's paths.

3.1.4 Sperm Cell Path Reconstruction

To analyze sperm cell movement, it first needs to be tracked across all frames. To match a detected sperm cell to a sperm cell in a previous frame and determine if it is the same, an

array of already processed sperm cells in previous frames is maintained. The size of this buffer is variable and by default set to 20 frames. A larger frame memory gives better results but increases processing time. A frame buffer smaller than 10 is insufficient; YOLO is not that precise, and sperm cells can go undetected for several frames, meaning that if the buffer is too small, the information about detected and matched sperm cells can be lost. To see more about YOLO detection accuracy see Section 4.1.2. The program then needs to deal with sperm cells appearing out of nowhere in the middle of the frame. Sperm cells are initially matched based on bounding box overlap. The necessary amount of overlap needed to match two sperm cells reduces the further back in frame memory the program goes. This is the first phase of matching; the second is done later when the direction of a sperm cell within the bounding box is determined. In the second phase, the program tries to match sperm cells that could not be matched during the first phase. The requirement for bounding box overlap is minimized, yet the direction of the sperm cells must align, and the location of each sperm cell needs to be consistent with the trajectory it has previously followed. The position of a sperm cell in the current frame is forecasted using its velocity and the direction last recorded. These forecasts are preserved even if a sperm cell does not find a match, and should they eventually align with a detected sperm cell in future frames, these predictions are utilized in the final image to bridge any gaps that have emerged. These gaps may result from sperm cell collisions or instances where sperm cells go undetected by YOLO. When everything functions as intended, each sperm cell that appears in the video is identified, and all its detections by YOLO are matched.

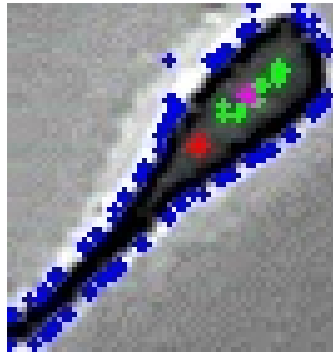
3.1.4.1 Determining Sperm Movement Direction

To determine sperm cell movement direction, each detected bounding box is isolated. Using convolution filters, for sharpening the image and thresholding, the contour of a sperm cell is extracted. The thresholding process is executed in two stages: initially for dark objects and subsequently for bright objects. Following this, the results from both stages are combined into a single image. This approach is specifically adopted to address the *Halo problem* discussed in detail in Section 3.2.2.1, characterized by the reflection of light from sperm cells into the microscope's lens, causing the sperm cells to appear illuminated rather than dark. Once sperm cells are isolated from the background, the contours are detected. Using the contour, the center of mass of each sperm cell is calculated. Then the sperm cell head needs to be detected.

The first approach attempted was template matching, where a general sperm head was created by merging several sperm cells into one. However, this approach did not work due to the varying sizes of sperm cells across all videos and their different orientations. Feature matching was also tested but failed due to the extreme simplicity of a sperm head and the overall lack of details that could be used as distinct features for matching. The approach used in the end was repetitive erosion of the image until only the sperm head was left, as it has the greatest mass.

After that, only the head contour is extracted, and the center of mass of a sperm head is calculated. Using the sperm cell center of mass and sperm cell head center of mass vector of a sperm cell's direction is calculated. The result of contour extraction and center of mass calculation can be seen in Figure 3.2. This process is not the most accurate and it was happening that out of 10 frames in around 2 of them the angle was calculated incorrectly or not at all for that reason angle correction has been implemented.

Angle correcting deals with incorrect angles in a similar way to how I am dealing in this approach with missing detection by YOLO by adding buffer. After angles for all sperm cells are detected, angle correction is performed. This means that the currently measured direction of movement is compared to the direction that was measured for this sperm cell in previous frames. If it is not within reasonable variance, it is not considered a correct measurement, which affects the path calculation of this sperm cell, as the sperm cell position will not be used to update the path. If it is within limits, the angle is saved so it can be used for angle correction in following frames, and the calculated center of the mass of the sperm cell head is used to update the sperm



■ **Figure 3.2** In this image you can see single sperm cell. Blue outline signals sperm cell contour. The red dot is the center of a mass of this contour. In green you can see the sperm cell head outline and the pink dot is its center of a mass

cell path. Same as bounding box mating for angle correction we can also calibrate to how many angles in a past the current angle is compared to.

3.1.4.2 Determining the Sperm Path

In the initial approach, path reconstruction is straightforwardly executed by recording the positions of the sperm cell heads detected during the determination of the sperm cell's movement direction. When the head is not detected, its position is estimated based on the sperm cell's previously observed trajectory, and a prediction for its next position is made. This prediction involves examining the sperm cell's past direction of movement and calculating the average distance between all recorded positions. The anticipated position is then calculated using those from the last recorded spot. These predictions are made for every frame and apply to all detected sperm cells, even when the head position is identifiable. Alongside the direction, these predictions are utilized to confirm the accuracy of each measurement. Similarly to how angles are employed to match sperm cells when bounding box overlap alone is insufficient, these predictions are used for matching sperm cells in subsequent frames. This strategy as previously mentioned also aids in bridging detection gaps, thereby preventing potential loss of tracking for the sperm cell. Currently, the paths generated by this method are coarse and lack full connectivity. They were intended to be refined and made more cohesive using more advanced techniques, such as the Kalman smoother. However, these enhancements were never implemented, as the development of this approach was discontinued before reaching that phase.

3.2 Approach Using Background Subtraction for Detection

This approach utilizes background isolation techniques (specifically, Median filtering as described in [45, 44]2.6.7) to detect individual sperm cells. Then, each individual sperm cell is tracked using the Kalman filter [32]2.6.3 and classified using a residual neural network [39]2.6.4. This approach aims to overcome some flaws of the previous approach. YOLO, which proved to be insufficient, has been removed, and the individual sperm cells are detected by calculating the video background for each video. This background is subtracted from each frame, leaving only moving objects in the frame, including all the live sperm cells and usually some small bubbles or other small objects that are floating in the sample. Most of them can be filtered out by size. How this approach deals with the larger ones is discussed later in Section 3.2.4. Once all sperm cells in a frame are detected, the next phase is to try to match them with some sperm cell from the previous frame. This is done very similarly to the first approach by measuring bounding box overlap, and if that is not enough, then the sperm is matched based on the predicted path. The big difference is that in this approach, the predictions are done more systematically using the Kalman filter. The Kalman filter is able to track each sperm cell very accurately and even fill in gaps when a sperm cell was not detected for some reason, as discussed more in Section 3.2.3. In this approach, unlike in the previous one, all detected objects are classified by a residual neural network into four classes: *single sperm cells*, *a group of cells*, *sperm cell bundle*, and *other*. This is done for the purpose of clutter removal and also to distinguish between single sperm cells and sperm cell bundles, which is the main unique feature over traditional CASA systems. The final step is the velocity calculation for each detected object that was tracked long enough. Three velocities are calculated: the Straight-Line Velocity (VSL), the Average Path Velocity (VAP), and Curvilinear velocity (VCL) [24]2.4.

3.2.1 Algorithm

The methodology employed in this study is meticulously structured into several sequential steps, each designed to address distinct aspects of sperm cell analysis through video recordings. This systematic approach enables comprehensive analysis from detection to classification and quantitative assessment. Below, we outline the key components of our methodology:

1. **Sperm Cell Detection:** Initial analysis focuses on isolating live sperm cells from the background. This process involves employing contour detection techniques to accurately identify and delineate individual sperm cells, subsequently surrounding each identified cell with a bounding box for further analysis (see Section 3.2.2).
2. **Path Reconstruction:** The trajectory of each sperm cell throughout the video sequence is reconstructed by examining the overlap of bounding boxes across consecutive frames. In instances where direct measurements are unavailable or measurements are noisy, the Kalman filter is applied to estimate the sperm cell's position, ensuring continuous tracking unencumbered by inaccurate measurements (see Section 3.2.3).
3. **Bounding Box Classification:** Upon successful detection and tracking, each bounding box is subjected to classification using a residual neural network. This classification categorizes the contents of each box into one of four distinct groups: a single sperm cell, a bundle of cells, a group of nearby cells, or other entities (see Section 3.2.4).
4. **Velocity Calculation:** With the paths of individual sperm cells established, we proceed to calculate key motility parameters such as the Straight-Line Velocity (VSL), the Average Path Velocity (VAP), and Curvilinear Velocity (VCL). These calculations are performed using the

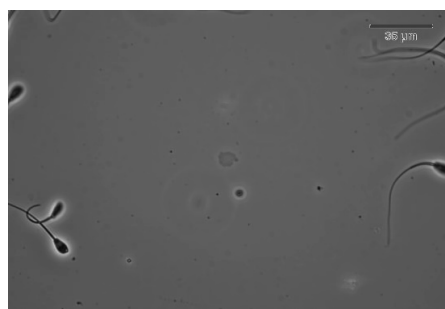
reconstructed paths and are essential for assessing the straightness and linearity of the cells' path (see Section 3.2.5).

5. Final Analysis: The culmination of this research involves a detailed evaluation of the collected data, applying a set of predefined rules to assess and interpret the behavior and characteristics of the sperm cells. This final step synthesizes the findings from the entire analysis, providing insights into sperm cell dynamics (see Section 3.2.6).

3.2.2 Sperm Cell Detection (Image Segmentation)

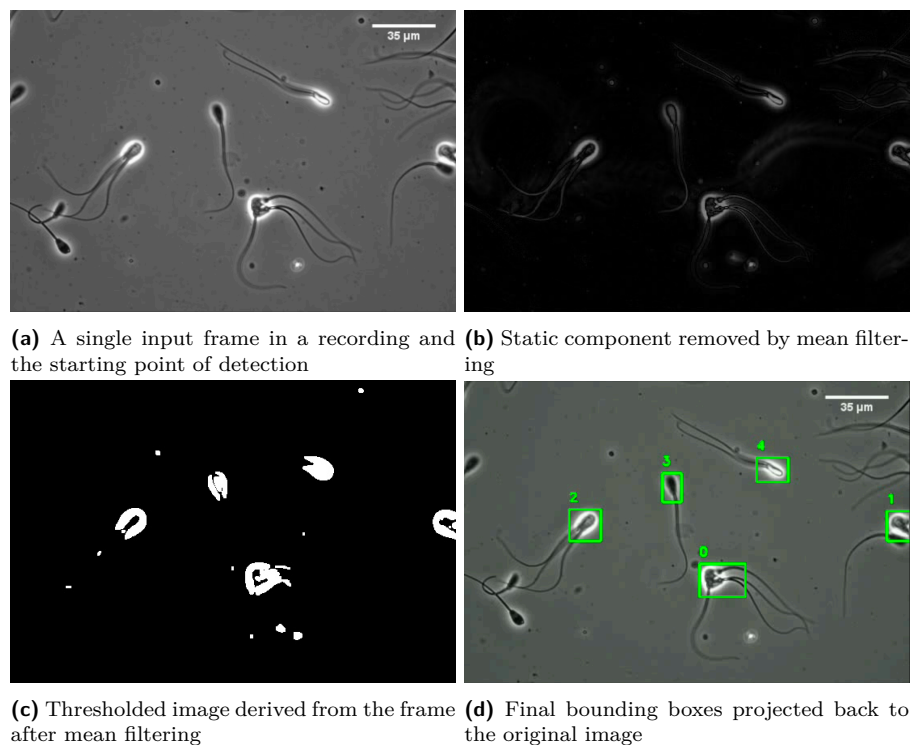
The goal of this step is to detect all live sperm cells in separate frames as precisely as possible. To achieve this, I first remove the static background. I used the background subtraction technique, specifically Median filtering 2.6.7. An example of the retrieved background image can be seen in Figure 3.3. Once obtained, I subtract this background from each frame, resulting in an image with only moving objects, where most of them are sperm cells. This approach is sufficient, as the videos that this program is intended for are short, and the environment is simple and stable. This way, static objects, mostly dead sperm cells and noise, are removed, while moving objects, like live sperm cells and some objects floating in the liquid, are preserved and highlighted.

The whole process step by step is depicted in Figure 3.4. In 3.4a, you can see the original frame. In Figure 3.4b, you can see the same frame after the background from Figure 3.3 has been subtracted from it. The subtracted image is then converted into a black and white image by using a dynamically derived threshold. The threshold is determined using Otsu's method 2.6.5. We perform a few steps of erosion and dilation to eliminate noise as described in [52]2.6.10. The result can be seen in Figure 3.4c. After that, I use the black and white image to derive the bounding boxes of individual sperm cells. Bounding boxes are drawn around each distinct group of white pixels. The size of the bounding box has to satisfy minimum size requirements; this is done to filter out small background noise that can be seen in Figure 3.4c. In parallel with the bounding box detection, the same black and white image (Figure 3.4c) is used to identify the center of mass of each detected object, first of the whole sperm cell and then of just the sperm cell head. The reasoning why both of them are needed is discussed in detail in Section 3.2.5.1 and has to do with the way the sperm cell velocities are calculated. I use the center of mass information to correct the Kalman filter movement prediction, see Section 3.2.3. Finally, the bounding boxes are transferred back into the original image for visual inspection. The result is shown in Figure 3.4d.



■ **Figure 3.3** This figure shows an example of a background calculated using the Median Filter technique. There are clearly still some sperm cells left, but those are dead sperm cells that do not move. This application was developed primarily to focus only on live sperm cells, so this fact is not a problem in this case.

Although the algorithm is straightforward and computationally simple, it has its shortcomings. The main problem that it causes is that the bounding boxes can merge and expand if two sperm cells are too close to each other or colliding. The same happens when a sperm cell collides



■ **Figure 3.4** This figure demonstrates the step-by-step process of sperm cell detection.

or gets close to non-stationary noise or dead sperm cell which was not completely removed. This is a problem because I use the bounding box overlap to match sperm cells across frames, and when these expansions happen, I am unable to match sperm cells. I delve more into this problem and how I am solving it in the next Section 3.2.3. Sometimes it also happens that the sperm is not prominent enough, and the algorithm misses it. The accuracy of this approach is described in more detail in Section 4.2.2.

3.2.2.1 Halo Problem

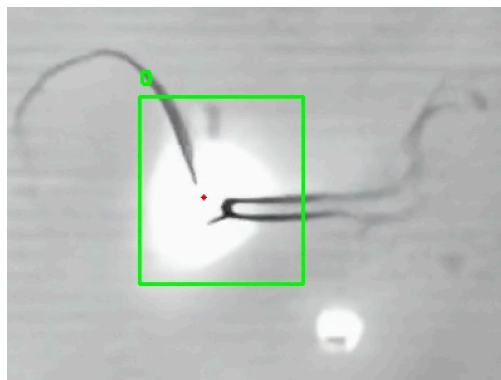
One issue that poses a problem in all steps of this program is what I've termed the Halo Problem. In all videos, sperm cells sometimes reflect light back into the microscope lens, creating a bright halo around the sperm cell head. An extreme case of this can be seen in Figure 3.5.

In the sperm cell detection step, this bright "halo" shifts the threshold for determining which pixel is white and which is black more towards bright colors, causing sperm cells that do not have a halo to sometimes go undetected.

In the path reconstruction step, these "halos" present an even bigger problem. They alter the shape of the sperm cell, introducing more inaccuracies to the measurements of the center of mass and subsequently to the correction of the Kalman Filter. It's important to note that this halo can change from frame to frame as the relative position between the microscope and sperm cell changes, introducing non-independent and non-identical noise, which can completely shift the reconstructed path in one direction; see Figure 4.1a.

Removing these "halos" systematically is challenging. In the final version of the program, I opted to replace ultra-bright pixels with generic gray ones, lowering the threshold for thresholding. This adjustment allowed for a more accurate determination of the center of mass of the sperm cell head for estimating the path to calculate the VCL.

In the Classification step, these "halos" also pose a problem, particularly because in training



■ **Figure 3.5** A sperm bundle reflecting light into the microscope lens.

data that I have received they tend to occur more around sperm cell bundles. The residual neural networks (ResNet) encountered difficulty when a single sperm cell had a large "halo," often classifying it as a sperm cell bundle rather than a single sperm cell. This issue was partially addressed by adjusting the bounding boxes, as discussed in depth in Section 3.2.4.

3.2.3 Path Reconstruction

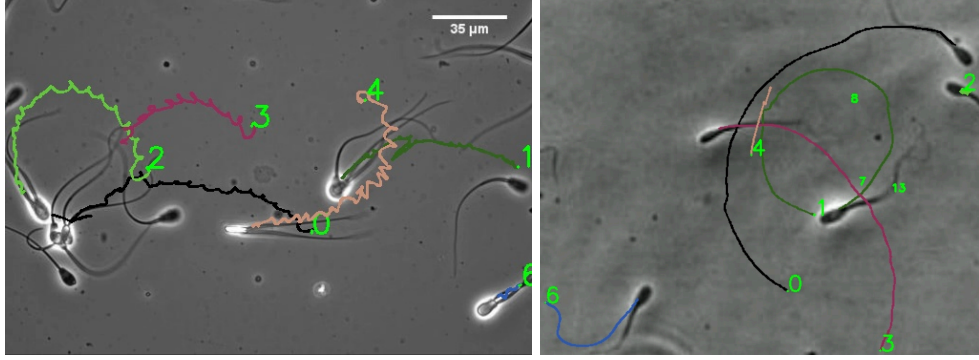
After detecting sperm cells in individual frames, the next step is to reconstruct the path of each sperm cell throughout the entire recording. In this approach, I process the recording sequentially. To achieve high accuracy, I employ a two-step approach. The first step relies on spatial and temporal proximity. I identify two sperm cells in subsequent frames as the same cell if the overlap of bounding boxes is higher than 80%. However, this approach heavily relies on correct bounding box detection, which is not always reliable due to reasons mentioned in Section 3.2.2. This can lead to less than 80% overlap and failure in connecting the path. To address the shortcomings of the simple approach and improve the accuracy of path reconstruction, I employ the Kalman filter [34, 32]2.6.3 to estimate the future position of a sperm cell. The general motivation is that if a sperm cell is lost for a few frames, I may connect it later when its bounding box reappears.

In the first frame of the video, all detected sperm cells are new, and I set up a new Kalman filter for each of them. In subsequent frames, I attempt to match newly detected sperm cells to those I have been tracking based on the bounding box overlap. If successful, I correct the Kalman filter based on the position of the newly detected sperm cell. After matching sperm cells based on bounding box overlap, I predict the next positions of all tracked sperm cells.

The second step of path reconstruction is triggered only if there are still some newly detected sperm cells left unmatched. In this case, I attempt to match them based on the estimated next position of tracked sperm cells that also have not been matched yet. If a newly detected sperm cell is not matched even in the second step, it is considered as a new sperm cell and is set up with a new Kalman filter to track its position.

By adding the second step, I was able to match sperm cells even when they were not detected due to collisions for several frames at a time. At the end of each video, the reconstructed paths are visualized, as shown in Figure 3.6.

As described in Section 2.6.3, the Kalman filter uses a linear model to describe the development of tracked objects. In this work, the linear model used for the Kalman filter is the Constant Velocity Model, a simple equation of motion in 2D space. The next position is calculated based on the current position and object velocity, with acceleration attributed to noise. The equations used are:



■ **Figure 3.6** Two examples of reconstructed paths of sperm cells. Each path has its own colour and sperm cell id that is used for reference in other documents.

For axis x :

$$\begin{aligned} x_{1,t} &= x_{1,t-1} + v_{x_1,t} \Delta t + w_{x_1,t} \\ v_{x_1,t} &= v_{x_1,t-1} + w_{v_{x_1,t}} \end{aligned} \quad (3.1)$$

For axis y :

$$\begin{aligned} x_{2,t} &= x_{2,t-1} + v_{x_2,t} \Delta t + w_{x_2,t} \\ v_{x_2,t} &= v_{x_2,t-1} + w_{v_{x_2,t}} \end{aligned} \quad (3.2)$$

Variables:

$$x_t = \begin{bmatrix} x_{1,t} \\ x_{2,t} \\ v_{x_1,t} \\ v_{x_2,t} \end{bmatrix}$$

From this, I build a state model that the Kalman filter can work with, consisting of two equations: the state equation and the measurement equation:

$$\begin{aligned} x_t &= A_t x_{t-1} + w_t, \quad w_t \sim (0, Q_t), \\ y_t &= H_t x_t + \varepsilon_t, \quad \varepsilon_t \sim (0, R_t). \end{aligned} \quad (3.3)$$

$$\begin{aligned} A &= \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\ H &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \end{aligned}$$

The real process noise covariance Q_t has not been calculated, and the standard default matrix has been used:

$$Q_t = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix},$$

Similarly, the real measurement noise covariance matrix R_t has also not been calculated. For the purpose of velocity calculation (see Section 3.2.5.1), I implemented my own Kalman Filter so I

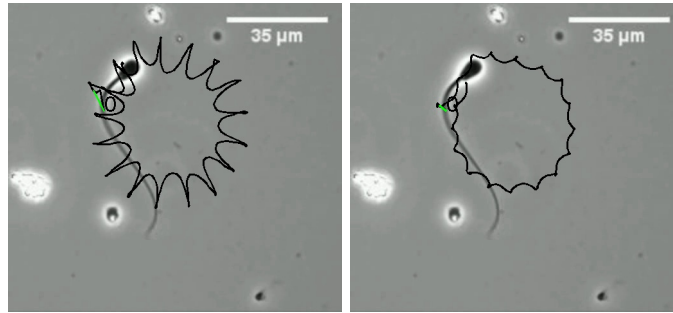
could have two R_t matrices: one that gives more trust to measurements and one that gives less, to predict two paths for each tracked object.

$$R_t = \begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix}, \quad (3.4)$$

$$VCL_R_t = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad (3.5)$$

The side-by-side comparison of the paths when different R_t values are used can be seen in Figure 3.7. The path estimated using the R_t matrix 3.4 gives less trust to measurements and is further smoothed out to obtain more accurate VAP and VSL. This Kalman Filter is corrected using the sperm cell position obtained by calculating the whole sperm cell center of mass.

On the other hand, the path estimated for calculating VCL using R_t 3.5 uses just the sperm cell head center of mass for correction. This seemingly small change makes quite a difference, as this position is less static and jumps a lot as the sperm cell moves.



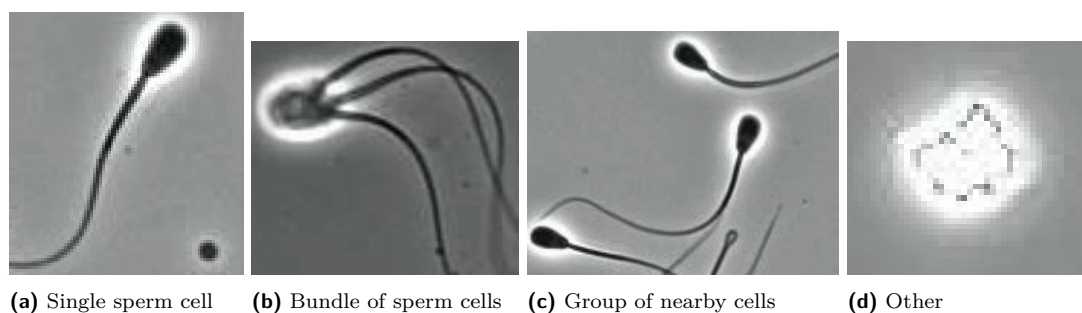
(a) This path was reconstructed using 3.5 and sperm cell head center of mass and was used to calculate the sperm cell VCL. **(b)** This path was reconstructed using 3.4 and sperm cell center of mass and was used to calculate sperm cells VAP and VSL.

■ **Figure 3.7** This figure illustrates how different R_t values, which give different weight to measurements, can affect the final path.

Using these default matrices can have some negative effects on estimation precision, but after performing path reconstruction accuracy tests (more about those in Section 4.2.1) and taking into account the Halo problem (Section 3.2.2.1), I concluded that the change they could bring is negligible.

3.2.4 Bounding Box Classification (ResNet)

I introduced the classification step primarily to distinguish between single sperm cells and sperm cell bundles, setting this system apart from other CASA systems currently in use. Classification also enables the filtering out of clutter in a video that is otherwise difficult to remove. Additionally, this step of classification offers the opportunity to detect patterns that could indicate bundle formation. For instance, if an object is initially detected as a single sperm cell, then as a group of nearby sperm cells, and finally as a bundle, there is a possibility that bundle formation might have occurred. To filter out leftover small objects that are not sperm cells but were not filtered out yet and to improve confidence in the moments of bundle formation, the content of each bounding box is classified into one of four classes: *sperm cell*, *more sperm cells*, *bundle*, or *other*. Examples of each class can be seen in Figure 3.8. "More sperm cells" refers to any bounding box that contains more than one sperm cell or bundle, with these sperm cells or bundles



■ **Figure 3.8** Examples of each classified class

together not forming a bundle. "Other" is classified as anything that is not a sperm cell, more sperm cells, or a bundle.

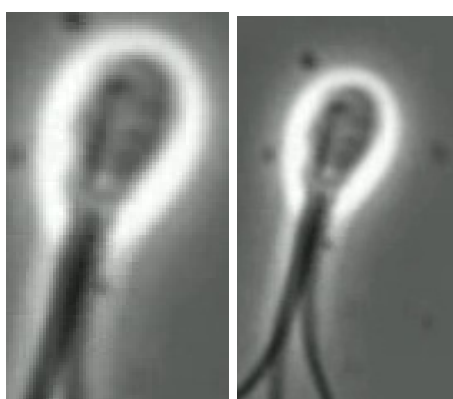
For bounding box classification, I opted to utilize the residual neural network (ResNet) architecture. ResNet is particularly well-suited for categorizing images and is highly efficient in doing so, making it the ideal tool for my problem. I chose to employ the smallest ResNet18 architecture pretrained on the ImageNet dataset. While I also experimented with the slightly larger ResNet50 architecture, the results did not significantly surpass those of ResNet18 to justify the higher computational demands of the larger model. The initial version of the model was trained using **10,568** images. To construct this dataset, I employed an iterative process of training the model, using it to classify new videos, and correcting any misclassifications. The confusion matrix of this model can be seen in Table 3.1.

■ **Table 3.1** Object classification confusion matrix of the first trained model

		Predicted labels			
		Single sperm cell	Other	Sperm cell bundle	Group of sperm cells
True Labels	Single sperm cell	2023	119	108	41
	Other	21	248	9	2
	Sperm cell bundle	73	16	562	41
	Group of sperm cells	35	11	20	264

ResNet requires all images to have the same size. I selected a size of 224 pixels. To address overfitting, I randomly cropped all images, undersampled the majority class (*sperm cell*), and oversampled the minority classes (*bundle*, *group of sperm cells*, and *other*). The oversampling involved selecting the minority class more frequently. For a single sperm cell, I captured its image every 20th frame, while for the minority classes, I captured images every 11th frame. In the initial version of the trained model, I encountered a challenge where the neural network was trained on images that exactly corresponded to the bounding box as detected. Consequently, I often lost part of the flagella during bounding box detection, as described in Section 3.2.2, due to dilation and erosion procedures used to remove noise. This removal of flagella was problematic because the flagella are crucial for distinguishing between a single sperm cell and a sperm cell bundle. Removing a significant portion of the flagella led to more bundles being classified as single sperm cells. This issue was later addressed by extending the bounding box by 30% to include the entire sperm cell when it is presented to the neural network for classification. The difference is illustrated in an example in Figure 3.9.

In the final model, the issue with small bounding boxes has been rectified, and additional training images have been incorporated to enhance performance on newly recorded videos provided by the University of Waterloo. The confusion matrix of this model is presented in Table 3.2. The F1 scores for each category are as follows:



(a) Original bounding box (b) Image passed to ResNet after extending it by 30%

■ **Figure 3.9** This figure demonstrates how the 30% extension can influence bounding box classification. The original bounding box was being classified as a single sperm cell, as the flagella separation was not clearly visible. However, after extending the bounding box by 30%, it becomes easier to detect multiple flagella, allowing for correct classification of this bounding box as a bundle.

- Single sperm cell: 0.940
- Other: 0.876
- Sperm cell bundle: 0.828
- Group of sperm cells: 0.729

The 30% bounding box extension contributed to the improvement of the F1 score for single sperm cells from 0.910 to 0.940, while the F1 score for sperm cell bundles increased from 0.808 to 0.828.

■ **Table 3.2** Object classification confusion matrix of the final trained model

		Predicted labels			
		Single sperm cell	Other	Sperm cell bundle	Group of sperm cells
True Labels	Single sperm cell	1842	48	32	23
	Other	26	421	13	7
	Sperm cell bundle	43	20	565	80
	Group of sperm cells	61	5	46	299

3.2.5 Velocity Calculation

As previously mentioned, a significant advantage of my solution is that, unlike other CASA systems, my system can distinguish between single sperm cells and sperm cell bundles. This, combined with accurate velocity calculation, makes this system a valuable tool for comparing the motility capabilities of single sperm cells and bundles. The system can calculate the straight-line velocity (VSL), average path velocity (VAP), and curvilinear velocity (VCL). For visualization, see Figure 2.4. Initially, I believed that calculating the VCL would be impossible. However, during development, I received new videos recorded under different environmental settings, which

improved the video quality and allowed for more accurate detection of sperm cell paths, thus enabling the calculation of VCL. Nonetheless, even with the new videos, accurately calculating VCL remains challenging, and its accuracy is still the lowest among all velocities. The difference in reconstructed paths between old and new videos can be observed in Figure 3.6. Alongside the velocities, our system can now also calculate the linearity (LIN) and straightness (STR) of the sperm cell's path [25, 24].

My program calculates VSL of sperm after it has been tracked for 2 seconds. During this time period, a sperm cell should roughly travel a distance equivalent to 2 times its length. This method of measuring VSL is standard, as described by the field expert, Mrs. Veronica Magdanz.

VAP is calculated from the average path, as explained in Section 2.4. In my program, I set the parameter n to 30, meaning that to obtain the average point, I consider 30 positions in the past and 30 positions in the future. I did not employ advanced techniques to enhance the reconstructed path, as discussed in the paper [25]. Although these techniques were tested on a few paths, they did not yield significant improvements. This aligns with the paper's findings, as the changes are negligible when working with videos with a high Frames Per Second (FPS) count. The paper's highest FPS count was 90, and the improvements were minimal, whereas this program operates with videos that have 100 FPS.

3.2.5.1 Curvilinear Velocity Correction

Calculating VCL proved to be more complicated than originally expected. At one point, tests showed that the VCL was constantly lower than when calculated manually. These tests are discussed in Section 4.2.4. To enhance the VCL, adjustments to sperm cell position detection had to be made. In the first version, measurements of position were taken at the point where the head meets the flagella. This meant that the detected path was less curvy compared to one where the position of the sperm cell was estimated using measurements of the position of the center of the sperm cell head. Additionally, the Kalman filter model had to be slightly modified to place more trust in the measurements so that the path is less smooth and follows the measurements more closely. However, this proved to be complicated, as giving more trust to the measurements caused the path to curve up and the VCL became more accurate, but at the same time, the accuracy of VAP and VSL dropped. For this reason, in the end, I had to implement my own Kalman filter that has a separate R matrix: one for predicting the path used to calculate the VCL, with this R giving more trust to the measurements, and a second R matrix for predicting the path for VAP and VSL. More information about the separate path estimation is provided in Section 3.2.3. A visual comparison of the path used for calculating VCL and the one used for VAP and VSL can be seen in Figure 3.7.

3.2.6 Final Analysis

After all moving objects have been detected and their paths reconstructed, and each bounding box has been classified, the final analysis is performed. First, an overall sperm cell classification is done. Due to the imbalanced nature of the training dataset, most of the bounding boxes are classified as sperm cells (the majority class). Rules have been set up to balance this and achieve the correct final classification. For example, if a sperm cell is detected in 100 frames, it will be correctly classified as a sperm cell in 95 out of 100 frames. Conversely, if a bundle is detected in 100 frames, it will be classified as a bundle only in 60 frames, and in the rest, it will be classified as a sperm cell.

This implies that if a sperm cell is classified as a bundle in more than 60% of the frames, the program can confidently identify it as a bundle, as it is highly unlikely for a single sperm cell to be misclassified as a bundle 60% of the time. The same principle applies to the other two minority classes, *other* and *group of nearby sperm cells*. After the overall classification, objects classified as *other* are removed. If an object is lost in the middle of a video and it is not

due to it leaving the frame, its last-known position and the frame number when it disappeared are saved. Subsequently, classification patterns are examined, searching for instances where the class prediction transitions from a group of nearby cells to a bundle, indicating potential bundle formation. If this pattern is detected, the program looks for additional evidence of bundle formation, such as sperm cells disappearing in close proximity simultaneously. If such evidence is found, potential bundle formation is included in the final report, along with the relevant frame number and coordinates, facilitating verification by the user.

Testing/Evaluation

In this chapter, I present the tests and performances of both of my attempted approaches. I evaluate detection accuracy, path reconstruction precision, classification accuracy, and velocity calculation accuracy.

4.1 Testing of Approach Utilizing YOLO

Small-scale tests were conducted to showcase the results achieved by the first approach in its final stage before it was abandoned. This was done to better illustrate its limitations and to compare these outcomes with the new approach that learned from previous mistakes and attempted to address these shortcomings.

4.1.1 Duplicity Detection by YOLO

Using YOLO to detect sperm cells presented a challenge: occasionally, a sperm cell would be detected multiple times within a single frame, leading to issues with bounding box matching. In this test, I evaluated the average occurrence of duplicates in a video and my effectiveness in identifying and eliminating these duplications. I selected three random videos and recorded the number of duplicates both when duplicate mitigation was deactivated and when it was employed (see Table 4.1). On average, I observed 250.3 duplicates in videos without duplicate mitigation and 177.7 when it was activated.

4.1.2 Missed Detections

In this section, I evaluate the accuracy of sperm cell detection, focusing on missed detections. The analysis was conducted on six videos, three containing bundles and three containing single sperm cells. I counted the frames in which a sperm cell was completely missed by the YOLO detector. These sperm cells were present throughout the entire videos, and ideally should have

■ **Table 4.1** Duplicates counted

	default	with removal
Video 1	369	303
Video 2	230	118
Video 3	152	112

■ **Table 4.2** number of miss detections out of 400

	1	2	3
bundle	265	129	36
sperm cell	94	49	98

been detected in all 400 frames. The results are summarized in Table 4.2. This test also highlighted an imbalance in the training dataset, with single sperm cells being the predominant class. It became evident that YOLO struggled more with detecting sperm bundles than single sperm cells. On average, a bundle was missed in 143 out of 400 frames, resulting in a detection accuracy of 64.17% for sperm cell bundles. Single sperm cells were missed on average in 80 out of 400 frames, with a detection accuracy of 79.92%. Overall, this yields an average detection accuracy of 72% for either a sperm cell or a sperm cell bundle by YOLO.

4.2 Tests of Approach Using Background Subtraction

The detection accuracy has been tested over a random sample of videos. Path reconstruction has been evaluated using four selected path scenarios, while neural network classification accuracy has been assessed using a random test sample of 59 videos. Velocities were tested using nine videos from the newer batch, where it was possible to calculate VCL.

4.2.1 Path Estimation Analysis and Tests

The current implementation of the application uses a standard Kalman filter with a linear model, as described in Section 3.2.3. To assess the accuracy of Kalman filter estimation for sperm paths, they have been manually labeled. These four paths represent the four most common scenarios that the estimation has to deal with.

The first scenario was an ideal path where there are measurements for all frames (Figure 4.1a).

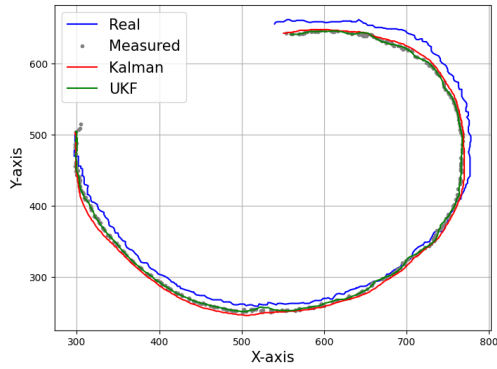
The second path was that of a sperm cell where, for a couple of frames, another sperm cell gets close to the tracked sperm cell to the point that they are in the same bounding box, distorting the measurements (Figure 4.1b). Additionally, in this second path, measurements are lost for a few frames as the sperm cell collides with a motionless sperm cell.

The third path is that of a sperm cell that, in frame **282**, forms a bundle with another sperm cell; therefore, from that frame on, there are no measurements for it (Figure 4.1c). In an ideal scenario, the estimations should be as close as possible to the newly formed bundle's position to give higher confidence that this sperm cell is really a part of the newly formed bundle.

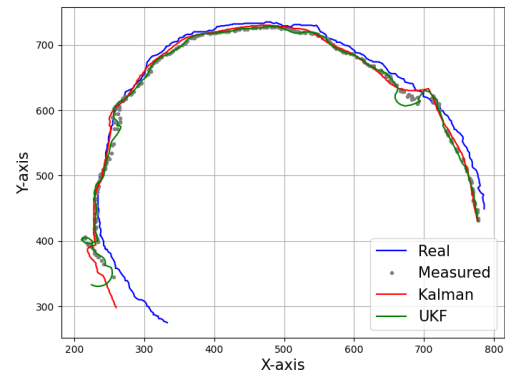
The last path represents the scenario where two motile sperm cell paths collide, and one of the paths is maintained due to the measurements, while the other one is lost (Figure 4.1d). It is expected from the Kalman filter to maintain the sperm cell's trajectory even when measurements are lost so that the sperm cell can be connected to its original path once the collision ends and its measurements resume.

While conducting these tests, I also confirmed, at that point suspected, distortion due to the "halo problem" 3.2.2.1. Additionally, these tests showed that the measurements are unable to capture the curvature of the sperm path if it is not predominant enough, mostly visible in path 4 in Figure 4.1d. This means that the program cannot calculate the curvilinear velocity (VCL). These tests were conducted on the old videos, and on the new ones, I was able to capture the curvature.

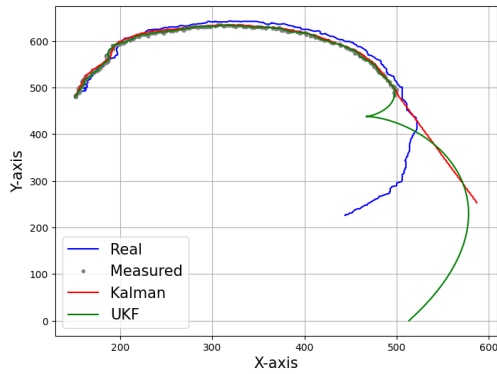
Another observation from these tests is that the halo is determined based on the sperm cell position relative to the microscope lens, best visible in path 1 in Figure 4.1a. At the beginning,



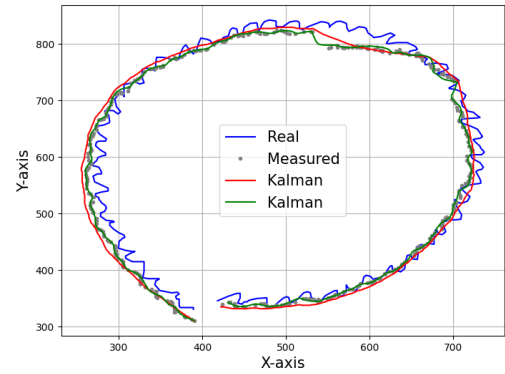
(a) Path 1: Ideal path, measurements are never lost.



(b) Path 2: Distorted measurements by two sperm cell coming in to close proximity.



(c) Path 3: Bundle form in frame 282 and from that point measurements are lost.



(d) Path 4: Two sperm cell collided and one of them loses measurements and the send has measurements distorted.

■ **Figure 4.1** Tests of path reconstruction accuracy

the "halo" distorts the measurements to the right from reality, and as the sperm cells loop, the "halo" moves and distorts the measurements to the left from reality.

The last tests conducted on path estimation were done to try and compare the standard Kalman filter with a linear model to the Unscented Kalman filter (UKF) with a basic nonlinear model. From the tests, it is visible that on the ideal path 1 4.1a with none of the measurements missing, they both perform almost identically when compared using Dynamic Time Warping (DTW) 4.3. But for the other paths where measurements are lost, UKF performs worse when implemented with a basic nonlinear model that does not describe the movement well. However, surprisingly, it performs better on path 4 4.1d as it tends to follow the measurements more compared to the standard Kalman filter, being able to capture the path curvature better.

4.2.1.1 Path Reconstruction Compared to CASA Systems

The second test that I conducted to evaluate the performance of my path reconstruction involved comparing my results to those obtained in a previous study [54]. For this purpose, I had to switch from DTW to the OSPA metric, as it was the metric used in the aforementioned study, which employed algorithms capable of tracking multiple targets simultaneously. In that study, four algorithms were tested for tracking sperm cells: Nearest-neighbor (NN) [55], Global nearest neighbor (GNN) [56], Probabilistic data association (PDA) [57], and Joint probabilistic data

■ **Table 4.3** DTW Methods for Different Paths

	Kalman Filter	UKF
Path1	5378.0	5092.94
Path2	6242.0	7999.46
Path3	9704.0	12942.15
Path4	6126.0	5539.49

association (JPDA) [57]. The comparison of these four algorithms alongside my test results can be seen in Figure 4.2.

4.2.2 Sperm Cell Detection Accuracy

First, I present the sperm cell detection accuracy. For detection accuracy, a random subset of 13 videos was selected, and all moving sperm cells were annotated for each frame. The expected number of detections for each sperm cell was then compared to the number of times this sperm cell was detected using our approach described in Section 3.2.2. The detection accuracy was 95.99%, and all the missed detections were due to the noise filtration by size rule.

4.2.3 Classification Accuracy

Thanks to the imbalance of the training dataset for ResNet, as discussed in Section 3.2.4, I decided not only to look at the individual number of bounding boxes classified properly as a metric to assess classification accuracy but also to examine how objects are classified overall after all rules are evaluated, as exemplified in Section 3.2.6.

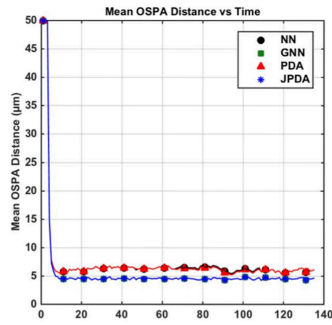
Over a random sample of test videos, the overall classification accuracy is 90.7%, which is a slight improvement compared to the accuracy of the final trained ResNet, which is 88.5%. I would like to mention that when the rules were originally implemented, the ResNet accuracy was only 45.3%. Only with the bounding box extension and new videos was I able to increase the ResNet accuracy to 88.5%, where these rules might not be as necessary anymore, but I decided to keep them in the final version of my program.

4.2.4 Velocity Accuracy

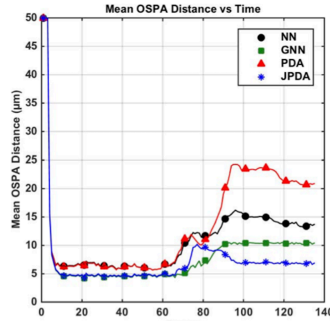
The velocity accuracy has been tested over 7 videos from the latest batch where it was possible to calculate the VCL. My program's estimations have been compared using Percentage differences to those calculated manually by Samantha Schwartz (see Table 4.4). Both VSL and VAP differences were within acceptable margins according to Veronika Magdanz, but the VCL calculation was clearly incorrect. Mrs. Magdanz suggested that the path estimated by my program seemed smoothed out (see Figure 4.3). Two potential reasons for this have been discovered:

- My system measured sperm cell position at the point where flagella met the sperm cell head and not at the center of the sperm cell head.
- The way the Kalman filter was set up resulted in low trust in measurements, leading to path smoothing.

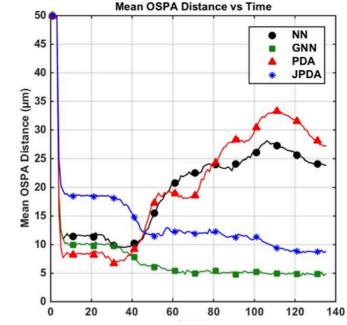
Both of these problems have been addressed (see 3.2.5.1) and I have conducted a second test. The results of the second test can be seen in Table 4.4. I was able to reduce the difference between the VCL estimated by my program and the VCL obtained manually, but VSL and VAP differences grew. These test results led me to the final solution where I separated the path for calculating VCL from the one that I used to calculate the VSL and VAP. For the final results,



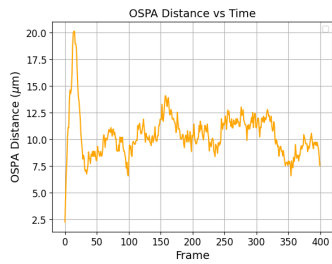
(a) Ideal paths: sperm cells do not cross paths and do not get close to each other.



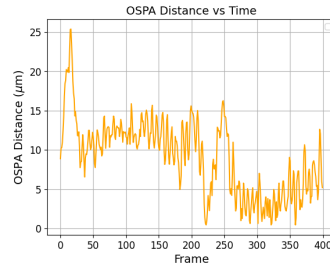
(b) Sperm cells cross paths and measurements are less accurate or lost.



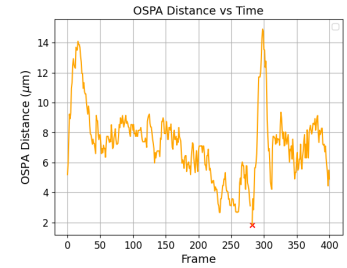
(c) Most problematic scenario when target tracking. Sperm cells that have a minimum separation of $10 \mu\text{m}$ but do not cross each other's paths.



(d) Ideal path OSPA distance.

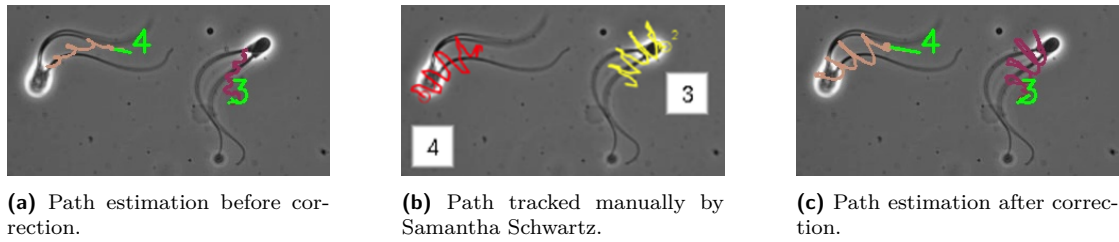


(e) Path OSPA distance when sperm cells nearly collide.



(f) Path OSPA distance during bundle formation.

Figure 4.2 In this figure, I compare my target tracking accuracy to those reached in the [54] study. OSPA metric returns distance in μm . The algorithms used in the study are: NN = Nearest-neighbor (black), GNN = Global nearest neighbor (green), PDA = Probabilistic data association (red), and JPDA = Joint probabilistic data association (blue). The first image 4.2a shows the distances in a video where sperm cells do not cross paths and do not get close to each other; the second 4.2b comparison is over a video where sperm cells cross paths; the third video 4.2c shows sperm cells that have a minimum separation of $10 \mu\text{m}$ but do not cross each other's paths. In the [54] study, multiple targets are tracked at the same time, so mean OSPA is calculated. In my program, I track each sperm cell individually using a Kalman filter (orange). The first path (4.2d) represents the path of an uninterrupted sperm cell. The second path (4.2e) shows the path of a sperm cell that crosses paths with another sperm cell. The last path (4.2f) depicts the path of a sperm cell that loses measurement in frame 284 (marked with a red "x") and forms a bundle with a different sperm cell.

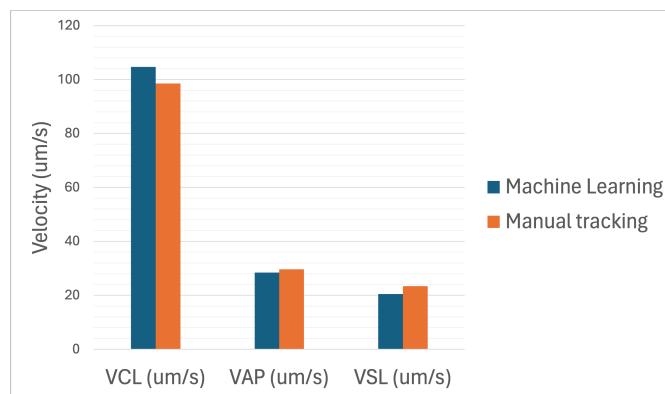


■ **Figure 4.3** Test of path reconstruction accuracy

■ **Table 4.4** Calculated differences

Velocity type / Difference	original	first correction	final
VCL	78.7	45.3	34.8
VSL	22.1	64.1	15.6
VAP	20.6	30.7	15.2

see the fourth column of Table 4.4. In Graph 4.4, you can see the comparison of average VCL, VSL, and VAP calculated by my program and those calculated manually by Mrs. Schwartz.



■ **Figure 4.4** Comparison of my Machine learning velocities to manual tracking over the same sample

According to field expert Mrs. Veronika Magdanz, the final test showed that my program was able to achieve acceptable results for her to consider it trustworthy and for it to be used in practice.

Chapter 5

User Interface

The user interface (UI) is in the form of a web application publicly available¹. Users have two options: either process one video at a time or upload a compressed folder with up to 50 videos that do not exceed the 100 MB threshold. This limitation has been placed because of the potential size of these videos and also because processing 50 videos can take anywhere from 10 to 90 minutes based on the complexity of the videos. Users, in both cases, also have to upload the scale of the video in pixels/ μm ; this is required so that the velocities can be calculated. This brings one more restriction on the option of uploading multiple videos: all the videos in one folder have to have the same scale, as there is no option to specify the scale for each video separately. Accepted video formats are .avi and .mp4. The webpage layout can be seen in Figure 5.1.

Sperm cell tracking

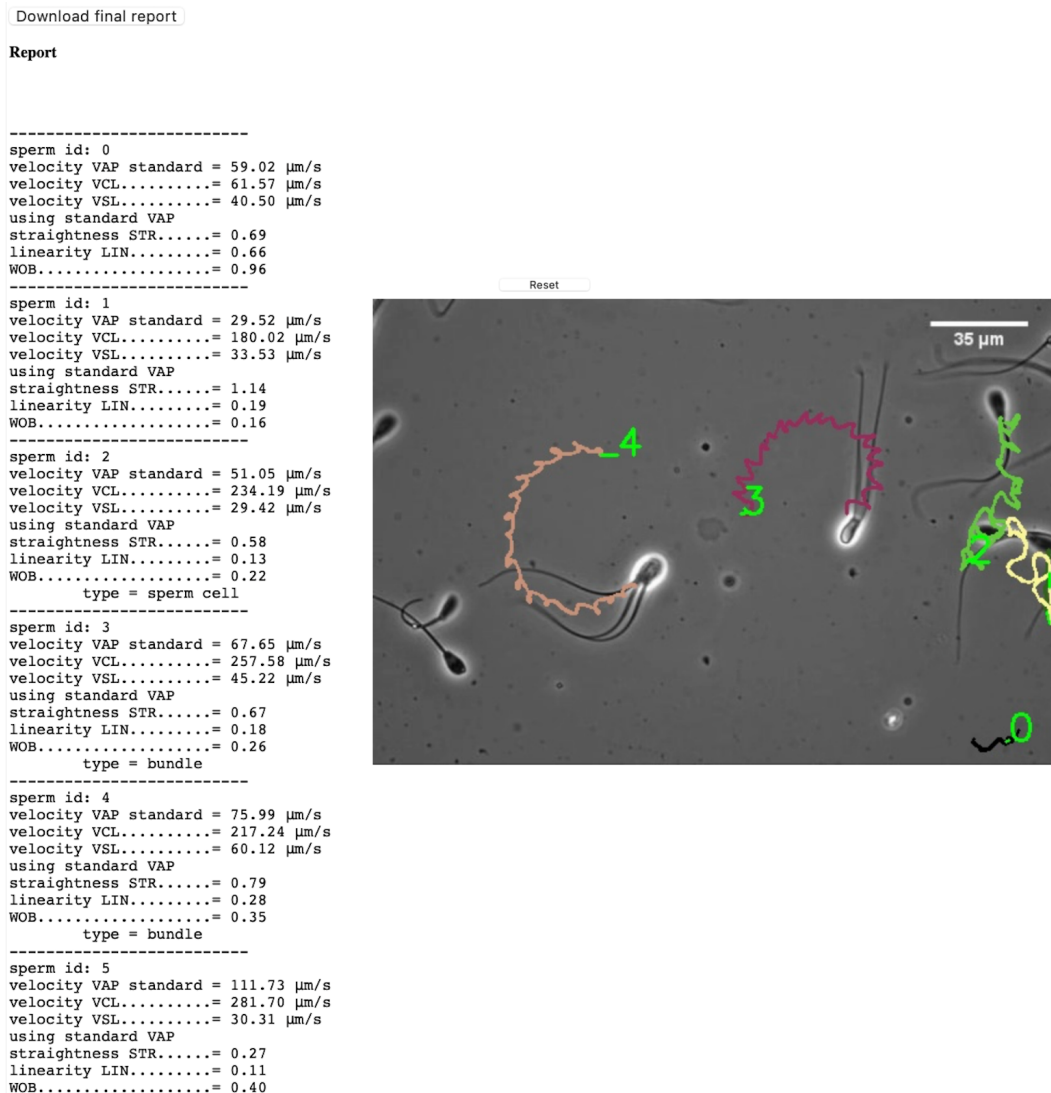
The screenshot shows the 'Sperm cell tracking' web application interface. At the top, there is a dashed box containing the text 'Select Video or Compressed folder with videos (Max size 100MB)'. Below this is a text input field labeled 'Video Scale (pixel/ μm): 1'. Underneath is a grey progress bar labeled 'Process uploaded video/s'. Below the progress bar is a blue progress bar labeled 'Number of videos processed' with a scale from 0 to 1. Below that is another blue progress bar labeled 'Number of frames processed in current video' with a scale from 0 to 700. At the bottom left, there is a button labeled 'Download final report'. At the bottom right, there is a button labeled 'Reset'.

■ **Figure 5.1** Web application layout

If users upload a single video, the program for processing it is started, and after it finishes, an image with detected objects and their paths is displayed to the user, along with the final report (see Figure 5.2). The UI also includes a progress bar so users can see how many frames have been processed so far, as each video takes several seconds to process. Users can download the image with paths, the final report, a CSV file with each sperm cell, its classifications, and velocities,

¹https://apps.datalab.fit.cvut.cz/sperm_tracking/

and in addition to that, the original video is downloaded with the addition of bounding boxes and their classifications.



■ **Figure 5.2** In this image, you can see the results shown after the video is processed.

If users decide to upload the compressed folder with videos, I first check if it contains only videos. Then the program for processing videos is run for each video, and its results are stored in one folder. For this option, the results are not displayed in the UI. The only way to review the results is once all videos are processed, which is signaled through the progress bar showing how many videos have been processed so far, to download the folder with all the results for each video. Same as previously, for each video, there will be an image with paths, a final report with statistics and velocities, a CSV file with simple information about sperm cell calcification and its velocities, and a video enhanced with bounding boxes of detected sperm cells and their classifications. In Figure 5.3, you can see how the user interface looks while processing a zip file that contains 3 videos.

The whole UI is made using Plotly Dash, a low-code framework for rapidly building data apps in Python [58].



■ **Figure 5.3** In this image, you can see that the currently processed video is named "video_14 copy," which is the second video out of 3. The first video has already been processed, and the program is currently processing around the 280th frame of the second video out of a total of 400. You can also see that all the videos are scaled to 2.2556 pixels/ μm .

Discussion

In this chapter, I will give my final opinion on both of the tried approaches, discussing their flaws and advantages, and explain which one I have decided to choose. I will present the first hypothesis that my CASA tool will be used to verify. Finally, I will discuss the future direction that this tool development will continue in.

6.1 Final Approach Selection

The final approach that has been fully developed to meet all the requirements, calibrated, and put into use is the approach using background subtraction for detection and Kalman filter for path reconstruction. At one point, following the conduction of experiments discussed in section 4.1, the approach using YOLO has been fully abandoned. That is the reason that some of the features have not been implemented at all; therefore, they could not be discussed in Section 3.1. Therefore, a one-to-one comparison is not possible, and the main reason why the approach using background subtraction has been selected is due to how the approach using YOLO underperformed when comparing based on sperm cell detection.

The struggle with the quality of detection by YOLO was significant (see Section 4.1). Often, sperm cells were not detected, and in some instances, one sperm cell was detected multiple times within a single frame. Although the issue of multiple detections was addressed and mitigated, owing to the inconsistency of multi-detection, allowing me to flag and confidently remove duplicates, the path reconstruction process presented numerous challenges. Initially, the rapid improvements made the issues seem manageable, but in hindsight, especially when compared to the outcomes achieved with the approach using background subtraction, it is clear now that the results were subpar. A major complication was also the variance in calibration of video characteristics, such as quality, scale, and sample density. While the program performed adequately for certain video subgroups, it yielded poor results for others. This variance persisted, but the other approach is much better at handling these changes.

In retrospect, the initial approach was laden with misguided decisions regarding technology use. My intention to build upon previously conducted work by Daniil Pastukhov[53] and using a standard tool for object detection not taking into account the unique data that I was working with, aiming to simplify my task, inadvertently led me astray. YOLO's ineffectiveness in detecting small objects (see Section 2.6.1) and the overkill of using a neural network in a setting where I clearly proved later that a much simpler and faster tool for object detection could be used, which also has good explainability, therefore, can be better calibrated, were clear missteps. However, this journey provided a profound understanding of the data at hand, highlighted the challenges to be addressed, and through learning from these mistakes, I developed a second so-

lution. This new solution removed the deficiencies of the first approach, delivering results that were significantly better.

The primary drawback of the approach using background subtraction is its inability to detect stationary objects, but for the use case that the application is being developed, this is not required. Also, in none of the videos have I encountered a case where a sperm cell would stop moving and then start moving again, which would definitely cause problems when using this approach. In every other way, it is better than the approach I have implemented using YOLO. I need to stress again that this is not a fully fair comparison as the approach using YOLO was my initial way of solving the given task, and in the approach using background subtraction, I learned from my mistakes so if I were to replace background subtraction with YOLO but keep Kalman Filter and the rest, it would probably also be a working solution even though not as viable as one that I was able to get with background subtraction.

6.2 Program Use

The first use of my application will be for checking videos where sperm cells have been exposed to Heparin and control videos without Heparin to confirm the hypothesis that if Heparin is used, sperm cells create more bundles. My program will calculate the velocities of all sperm cells to see if sperm cells in the control group have similar speeds as sperm cells exposed to Heparin. If my program were to show that sperm cells in the sample exposed to Heparin have significantly higher velocities, it could be that more bundles are created just because of the higher activity and not due to the fact that Heparin spurs bundle formation.

As the tool is still in development, it is being slightly modified to only focus on calculating velocities, and the rest of the rules discussed in Section 3.2.6 are being turned off. Also, the output is modified to return Comma-separated values (CSV) files that can be easily exported to Excel and analyzed by a field expert.

This is also the first major test of the user interface and its usability. Based on the feedback given, I am expecting to modify it accordingly.

6.3 Future Development

I believe that currently, the program is still not in its final form. I am expecting to continually work on it and improve the user interface based on feedback I will be getting from the research team at UW as they are going to be using my program for a wide range of tests that they are currently preparing. I also want to add features that would give more power to the researchers to modify certain variables of the program without the requirement of having programming knowledge, so my program can work in an ideal way for their type of use case as there is great variance in the types of recordings of a spermatozoa sample that can be made, and there is a limit to how much the program can be universal.

Conclusion

I was tasked with learning about and introducing concurrent research of spermatozoa, specifically of sperm bundles and their role in nanorobotics research. And with the development of a tool that could help with the very specific research in this untapped territory where modern CASA systems are not fully capable of supporting researchers. This tool needed to be able to identify individual moving objects in provided recordings of spermatozoa, classify these objects into categories of interest, and place significant emphasis on accurately distinguishing between single sperm cells and conglomerations of sperm cells, referred to here as "bundles." Subsequently, reconstructing their paths so they could be used to calculate their velocities, namely the Straight Line Velocity (VSL), Average Path Velocity (VAP), and curvilinear velocity (VCL).

As a result, I introduce a novel, multistep methodology employing machine learning techniques to track sperm cells within video recordings. My approach demonstrates the capability to accurately identify and track individual sperm cells, as well as to distinguish these single entities. Through extensive experimentation and analysis, my results affirm the high precision and reliability of my machine learning-based tracking solution in the context of sperm cell observation.

Presented within the Testing/Evaluation section 4 of this thesis, my findings elucidate that the proposed methodology for individual sperm cell detection achieves a remarkable detection accuracy of 95.99% for live sperm cells. This technique effectively diminishes the interference caused by stationary background elements and non-viable cells, thereby enhancing both accuracy and performance metrics. Utilizing a combination of bounding box intersections and the application of the Kalman filter, my approach mitigates inaccuracies inherent in bounding box detection methodologies. The empirical evidence demonstrates that, under optimal conditions where measurement data remain unlost, the deviation between the path identified by my system and a path annotated manually is quantified at a distance of 5378 units. This identified path forms the basis for subsequent estimations of key motility parameters, offering insights into the dynamic behavior of sperm cells.

Concurrently with the process of path identification, my study also focuses on determining the nature of the object enclosed within each bounding box. For this purpose, I employ the ResNet architecture, a convolutional neural network renowned for its efficacy in image recognition tasks. The implementation of ResNet in my methodology is underpinned by a foundation of manually annotated data, ensuring the training phase is robust and tailored to the specific nuances of this domain. The F1 scores for each category are as follows:

- Single Sperm Cell: 0.940
- Other: 0.876
- Sperm Bundle: 0.828

- More Sperm Cells: 0.729

In the concluding phase of my analysis, the classification of objects on a per-frame basis is aggregated to facilitate the classification of entire paths. This aggregation process also involves the identification of pivotal points of transition within the paths.

I have implemented my methodological approach within a web-based application, accessible via the following URL: https://apps.datalab.fit.cvut.cz/sperm_tracking/. Additionally, the source code underpinning our application has been made publicly available for scholarly and development purposes on GitLab, which can be accessed at https://gitlab.fit.cvut.cz/horenjak/sperm_cell_tracking_app/.

Regarding the enhancement of path identification capabilities, I assume that transitioning to a non-linear modeling approach would more accurately encapsulate the dynamics of sperm cell motion. In parallel, the adoption of the Unscented Kalman Filter, or the exploration of more sophisticated methodologies such as the Joint Probabilistic Data Association Filter, which facilitates the simultaneous tracking of multiple targets, could offer improvements in path reconstruction accuracy. These proposed enhancements are predicated on existing literature and methodologies that have demonstrated efficacy in similar contexts [36, 59, 3, 54].

Appendix A

Additional Video Analysis Results

```

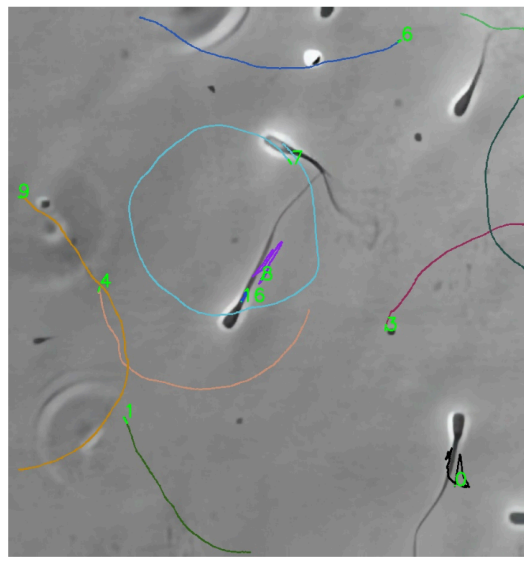
-----
sperm id: 0
velocity VAP standard = 23.40 μm/s
velocity VSL.....= 8.40 μm/s
type = sperm cell

    object disappeared
    on frame: 362
    on coordinates: [729, 737]
-----
sperm id: 1
velocity VAP standard = 100.15 μm/s
velocity VSL.....= 100.07 μm/s
-----
sperm id: 3
velocity VAP standard = 84.61 μm/s
velocity VSL.....= 94.12 μm/s
type = sperm cell
-----
sperm id: 4
velocity VAP standard = 95.72 μm/s
velocity VSL.....= 72.97 μm/s
type = sperm cell

    object disappeared
    on frame: 281
    on coordinates: [498, 502]
-----
sperm id: 6
velocity VAP standard = 86.45 μm/s
velocity VSL.....= 110.05 μm/s
type = sperm cell
-----
sperm id: 7
velocity VAP standard = 122.94 μm/s
velocity VSL.....= 74.61 μm/s
bundle might have formed
    frame: 399
    on coordinates: [[455 226]]
-----
sperm id: 8
velocity VAP standard = 39.60 μm/s
velocity VSL.....= 9.01 μm/s
type = sperm cell

    object disappeared
    on frame: 261
    on coordinates: [438, 402]
-----
sperm id: 9
velocity VAP standard = 88.93 μm/s
velocity VSL.....= 89.34 μm/s
type = sperm cell
-----
sperm id: 11
velocity VAP standard = 140.73 μm/s
velocity VSL.....= 130.03 μm/s
type = sperm cell
-----
sperm id: 12
velocity VAP standard = 101.11 μm/s
velocity VSL.....= 96.66 μm/s
type = sperm cell
-----
sperm id: 16
velocity VAP standard = 29.51 μm/s
velocity VSL.....= 8.74 μm/s
type = sperm cell

```

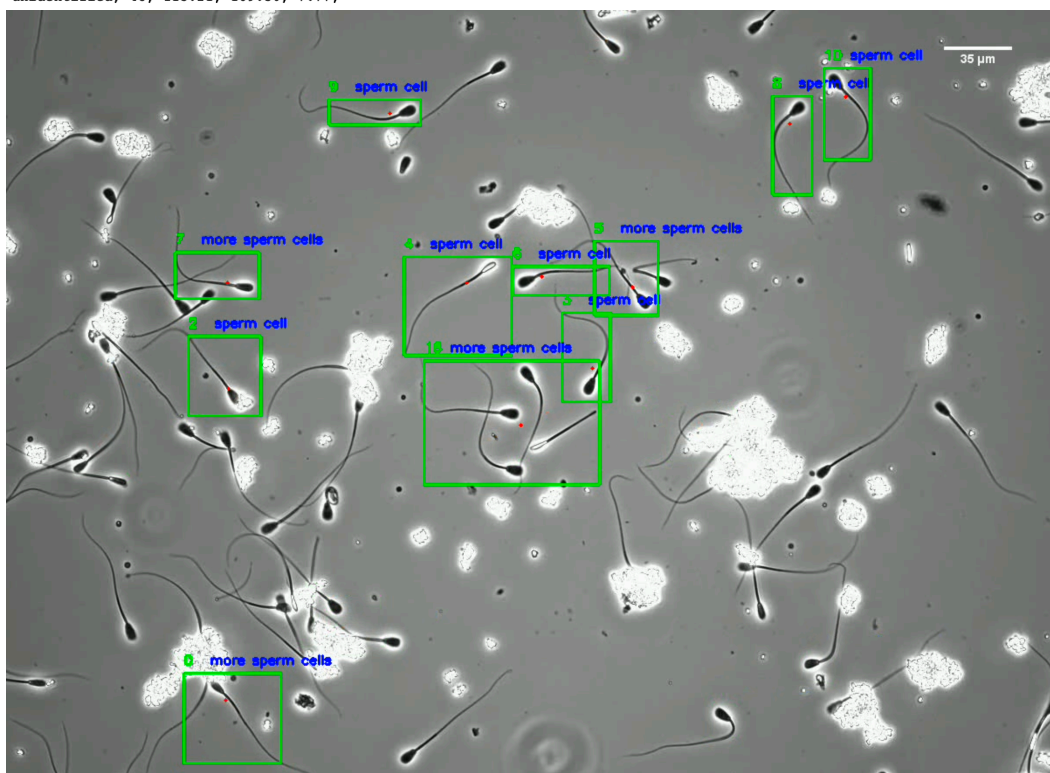
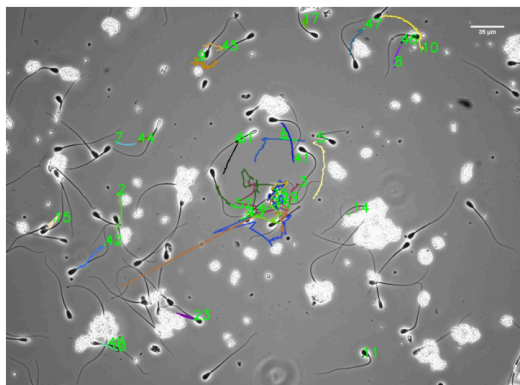


■ **Figure A.1** In this figure, you can see the results of the analysis of a video where bundle formation has occurred between sperm cell **4** and sperm cell **7** and has been correctly flagged.

```

type, id, VAP, VCL, VSL
single, 6, 130.78, 243.33, 103.43,
single, 8, 28.75, 70.66, 23.42,
single, 11, 1.43, 14.67, 2.12,
single, 17, 13.87, 195.78, 7.07,
single, 23, 57.74, 330.21, 2.06,
single, 44, 62.08, 137.28, 36.74,
single, 47, 90.22, 389.18, 60.04,
single, 51, 163.15, 294.66, 64.76,
single, 52, 131.56, 395.89, 70.00,
unidentified, 0, 20.53, 98.78, 7.65,
unidentified, 1, 142.36, 624.07, 35.11,
unidentified, 2, 34.68, 359.09, 38.00,
unidentified, 3, 174.53, 520.42, 116.33,
unidentified, 4, 9.76, 231.03, 1.00,
unidentified, 5, 76.50, 212.99, 67.23,
unidentified, 7, 22.11, 51.64, 21.10,
unidentified, 9, 44.25, 67.48, 23.02,
unidentified, 10, 73.83, 165.40, 57.64,
unidentified, 14, 37.14, 96.62, 12.38,
unidentified, 15, 35.28, 161.24, 14.21,
unidentified, 16, 125.86, 1069.68, 20.80,
unidentified, 19, 284.52, 327.57, 23.05,
unidentified, 35, 123.88, 235.25, 71.78,
unidentified, 41, 88.09, 174.99, 80.23,
unidentified, 42, 100.48, 507.67, 72.12,
unidentified, 43, 126.36, 626.05, 51.84,
unidentified, 45, 99.24, 215.34, 26.10,
unidentified, 46, 26.50, 94.49, 19.89,
unidentified, 48, 113.21, 169.50, 7.77,

```



■ **Figure A.2** The program encounters problems with path reconstruction and classification due to the high density of sperm cells and clutter. Bounding boxes often encompass multiple sperm cells, leading to insufficient confidence in classification and distortion of position measurements.

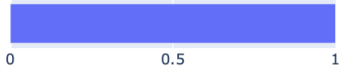
Sperm cell tracking

Select Video or Compressed folder with videos (Max size 100MB)

Video Scale (pixel/ μ m): 2.2762

Process uploaded video/s

Number of videos processed



Number of frames processed in current video

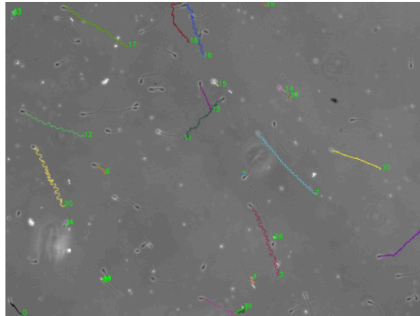


Download final report

Report

```
type, id, VAP, VCL, VSL
single, 2, 6.65, 4.05, 6.40,
single, 7, 28.68, 80.93, 26.35,
single, 10, 18.89, 70.44, 16.55,
single, 12, 22.78, 66.21, 22.30,
single, 13, 10.97, 55.27, 9.19,
single, 14, 11.15, 19.96, 4.96,
single, 18, 19.22, 73.35, 15.59,
single, 19, 12.44, 56.51, 9.75,
single, 23, 17.39, 77.08, 12.73,
single, 24, 5.35, 102.01, 3.04,
single, 28, 20.33, 163.62, 1.77,
single, 30, 3.49, 19.10, 1.60,
single, 34, 8.28, 17.73, 4.61,
single, 39, 75.97, 164.11, 29.88,
single, 44, 3.36, 12.41, 1.60,
bundle, 0, 7.53, 19.22, 11.34,
bundle, 1, 6.85, 211.98, 2.80,
bundle, 15, 2.72, 38.44, 1.60,
bundle, 33, 50.80, 97.28, 4.37,
bundle, 47, 13.43, 127.29, 1.03,
unidentified, 3, 25.10, 105.86, 20.45,
unidentified, 4, 8.03, 35.95, 3.25,
unidentified, 8, 4.27, 40.87, 1.41,
unidentified, 9, 5.56, 9.38, 2.50,
unidentified, 11, 20.31, 68.14, 22.63,
unidentified, 16, 24.46, 71.79, 22.22,
unidentified, 17, 24.44, 83.44, 19.56,
unidentified, 20, 24.81, 101.35, 22.33,
```

Reset



■ **Figure A.3** Full User interface with high density video and CSV output.

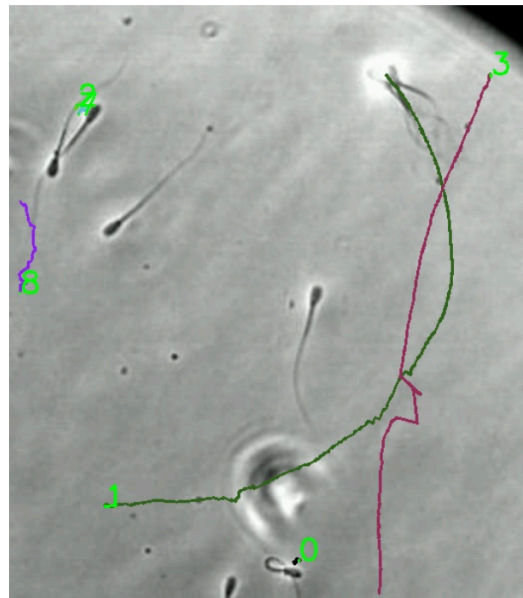
```

-----
sperm id: 0
velocity VAP standard = 3.94 μm/s
velocity VSL.....= 2.36 μm/s
type = sperm cell

object disappeared
on frame: 154
on coordinates: [288, 564]
-----
sperm id: 1
velocity VAP standard = 81.61 μm/s
velocity VSL.....= 84.93 μm/s
type = bundle
-----
sperm id: 2
velocity VAP standard = 3.39 μm/s
velocity VSL.....= 1.03 μm/s
type = sperm cell

object disappeared
on frame: 85
on coordinates: [68, 103]
-----
sperm id: 3
velocity VAP standard = 89.99 μm/s
velocity VSL.....= 90.32 μm/s
type = sperm cell
-----
sperm id: 7
velocity VAP standard = 6.14 μm/s
velocity VSL.....= 2.80 μm/s
-----
sperm id: 8
velocity VAP standard = 91.14 μm/s
velocity VSL.....= 81.64 μm/s
type = sperm cell

```



■ **Figure A.4** In this figure, you can see a video where two sperm cells, **1** and **3**, have collided, leading to less accurate path reconstruction, especially for sperm cell **3**.

Bibliography

1. Computer-assisted sperm analysis (CASA): Capabilities and potential developments. *Theriogenology*. 2014, vol. 81, no. 1, pp. 5–17. Available from DOI: 10.1016/j.theriogenology.2013.09.004.
2. MORTIMER, Sharon T; VAN DER HORST, Gerhard; MORTIMER, David. The future of computer-aided sperm analysis. *Asian journal of andrology*. 2015, vol. 17, no. 4, p. 545.
3. CHOI, Ji-won; ALKHOORY, Ludvik; URBANO, Leonardo F.; MASSON, Puneet; VERMILYEA, Matthew; KAM, Moshe. An assessment tool for computer-assisted semen analysis (CASA) algorithms. *Scientific Reports*. 2022, vol. 12, no. 1, p. 16830. Available from DOI: 10.1038/s41598-022-20943-9.
4. VIRKKI, Niilo. Sperm bundles and phylogenesis. *Zeitschrift für Zellforschung und mikroskopische Anatomie*. 1969, vol. 101, no. 1, pp. 13–27.
5. SCHOELLER, Simon F; HOLT, William V; KEAVENY, Eric E. Collective dynamics of sperm cells. *Philosophical Transactions of the Royal Society B*. 2020, vol. 375, no. 1807, p. 20190384. Available from DOI: 10.1098/rstb.2019.0384.
6. ZHANG, Kaixuan; KLINGNER, Anke; LE GARS, Yohan; MISRA, Sarthak; MAGDANZ, Veronika; KHALIL, Islam S. M. Locomotion of bovine spermatozoa during the transition from individual cells to bundles. *Proceedings of the National Academy of Sciences*. 2023, vol. 120, no. 3, e2211911120. Available from DOI: 10.1073/pnas.2211911120.
7. MORCILLO I SOLER, Paula; HIDALGO, Carlos; FEKETE, Zoltán; ZALANYI, Laszlo; KHALIL, Islam S. M.; YESTE, Marc; MAGDANZ, Veronika. Bundle formation of sperm: Influence of environmental factors. *Frontiers in Endocrinology*. 2022, vol. 13. ISSN 1664-2392. Available from DOI: 10.3389/fendo.2022.957684.
8. JOHNSTON, Steve D; SMITH, Brett; PYNE, Michael; STENZEL, Deborah; HOLT, William V. One-sided ejaculation of echidna sperm bundles. *The American Naturalist*. 2007, vol. 170, no. 6, E162–E164.
9. BURNETT, William E.; HEINZE, Jürgen. Sperm Bundles in the Seminal Vesicles of Sexually Mature *Lasius* Ant Males. *PLOS ONE*. 2014, vol. 9, no. 3, pp. 1–4. Available from DOI: 10.1371/journal.pone.0093383.
10. FISHER, Heidi S; GIOMI, Luca; HOEKSTRA, Hopi E; MAHADEVAN, Lakshminarayanan. The dynamics of sperm cooperation in a competitive environment. *Proceedings of the Royal Society B: Biological Sciences*. 2014, vol. 281, no. 1790, p. 20140296. Available from DOI: 10.1098/rspb.2014.0296.

11. KHALIL, Islam S. M.; DIJKSLAG, Herman C.; ABELMANN, Leon; MISRA, Sarthak. MagnetoSperm: A microrobot that navigates using weak magnetic fields. *Applied Physics Letters*. 2014, vol. 104, no. 22, p. 223701. ISSN 0003-6951. Available from DOI: 10.1063/1.4880035.
12. MAGDANZ, Veronika; KHALIL, Islam SM; SIMMCHEN, Juliane; FURTADO, Guilherme P; MOHANTY, Sumit; GEBAUER, Johannes; XU, Haifeng; KLINGNER, Anke; AZIZ, Azaam; MEDINA-SÁNCHEZ, Mariana, et al. IRONSperm: Sperm-templated soft magnetic microrobots. *Science advances*. 2020, vol. 6, no. 28, eaba5855.
13. MIDDELHOEK, Kaz INA; MAGDANZ, Veronika; ABELMANN, Leon; KHALIL, Islam SM. Drug-Loaded IRONSperm clusters: modeling, wireless actuation, and ultrasound imaging. *Biomedical Materials*. 2022, vol. 17, no. 6, p. 065001.
14. MOORE, Harry; DVORÁKOVÁ, Katerina; JENKINS, Nicholas; BREED, William. Exceptional sperm cooperation in the wood mouse. *Nature*. 2002, vol. 418, no. 6894, pp. 174–177. Available from DOI: 10.1038/nature00832.
15. KHALIL, Islam S. M.; MAGDANZ, Veronika; SIMMCHEN, Juliane; KLINGNER, Anke; MISRA, Sarthak. Resemblance between motile and magnetically actuated sperm cells. *Applied Physics Letters*. 2020, vol. 116, no. 6, p. 063702. ISSN 0003-6951. Available from DOI: 10.1063/1.5142470.
16. KHALIL, Islam S. M.; FATIH TABAK, Ahmet; KLINGNER, Anke; SITTI, Metin. Magnetic propulsion of robotic sperms at low-Reynolds number. *Applied Physics Letters*. 2016, vol. 109, no. 3, p. 033701. ISSN 0003-6951. Available from DOI: 10.1063/1.4958737.
17. ZOU, Zonghao; LIU, Yuexin; YOUNG, Y. -N.; PAK, On Shun; TSANG, Alan C. H. Gait switching and targeted navigation of microswimmers via deep reinforcement learning. *Communications Physics*. 2022, vol. 5, no. 1, p. 158. Available from DOI: 10.1038/s42005-022-00935-x.
18. RIJSSELAERE, Tom; VAN SOOM, Ann; MAES, Dominiek; KRUIF, Aart de. Effect of technical settings on canine semen motility parameters measured by the Hamilton-Thorne analyzer. *Theriogenology*. 2003, vol. 60, no. 8, pp. 1553–1568.
19. DORADO, J; RIJSSELAERE, Tom; MUÑOZ-SERRANO, A; HIDALGO, M. Influence of sampling factors on canine sperm motility parameters measured by the Sperm Class Analyzer. *Systems Biology in Reproductive Medicine*. 2011, vol. 57, no. 6, pp. 318–325.
20. SURMACZ, Paulina; NIWINSKA, Anna; KAUTZ, Ewa; GIZINSKI, Slawomir; FAUNDEZ, Ricardo. Comparison of two staining techniques on the manual and automated canine sperm morphology analysis. *Reproduction in Domestic Animals*. 2022, vol. 57, no. 6, pp. 678–684.
21. RURANGWA, E; KIME, D.E; OLLEVIER, F; NASH, J.P. The measurement of sperm motility and factors affecting sperm quality in cultured fish. *Aquaculture*. 2004, vol. 234, no. 1, pp. 1–28. ISSN 0044-8486. Available from DOI: 10.1016/j.aquaculture.2003.12.006.
22. DCUNHA, Reyon; HUSSEIN, Reda S.; ANANDA, Hanumappa; KUMARI, Sandhya; ADIGA, Satish Kumar; KANNAN, Nagarajan; ZHAO, Yulian; KALTHUR, Guruprasad. Current Insights and Latest Updates in Sperm Motility and Associated Applications in Assisted Reproduction. *Reproductive Sciences*. 2022, vol. 29, no. 1, pp. 7–25. Available from DOI: 10.1007/s43032-020-00408-y.
23. RAEF, Behnaz; FERDOUSI, Reza. A review of machine learning approaches in assisted reproductive technologies. *Acta Informatica Medica*. 2019, vol. 27, no. 3, p. 205.
24. HIDAYATULLAH, Priyanto; AWALUDIN, Iwan; KUSUMO, Reyhan Damar; NURIYADI, Muhammad. Automatic sperm motility measurement. 2015, pp. 1–5. Available from DOI: 10.1109/ICITSI.2015.7437674.

25. OKUMUŞ, Fatih; KOCAMAZ, Fatih; ÖZGÜR, Mustafa Erkan. Using polynomial modeling for calculation of quality parameters in computer assisted sperm analysis. *Computer Science*. 2021, vol. 6, no. 3, pp. 152–165. Available from DOI: 10.53070/bbd.999296.
26. VASILESCU, Steven A.; DING, Lin; PARAST, Farin Yazdan; NOSRATI, Reza; WARKIANI, Majid Ebrahimi. Sperm quality metrics were improved by a biomimetic microfluidic selection platform compared to swim-up methods. *Microsystems & Nanoengineering*. 2023, vol. 9, no. 1, p. 37. Available from DOI: 10.1038/s41378-023-00501-7.
27. REDMON, Joseph; DIVVALA, Santosh Kumar; GIRSHICK, Ross B.; FARHADI, Ali. You Only Look Once: Unified, Real-Time Object Detection. *CoRR*. 2015, vol. abs/1506.02640. Available from DOI: 10.48550/arXiv.1506.02640.
28. JIANG, Peiyuan; ERGU, Daji; LIU, Fangyao; CAI, Ying; MA, Bo. A Review of Yolo Algorithm Developments. *Procedia Computer Science*. 2022, vol. 199, pp. 1066–1073. ISSN 1877-0509. Available from DOI: 10.1016/j.procs.2022.01.135.
29. LEDIG, Christian; THEIS, Lucas; HUSZAR, Ferenc; CABALLERO, Jose; CUNNINGHAM, Andrew; ACOSTA, Alejandro; AITKEN, Andrew; TEJANI, Alykhan; TOTZ, Johannes; WANG, Zehan; SHI, Wenzhe. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. 2017, pp. 105–114. Available from DOI: 10.48550/arXiv.1609.04802.
30. WANG, Xintao; YU, Ke; WU, Shixiang; GU, Jinjin; LIU, Yihao; DONG, Chao; QIAO, Yu; CHANGE LOY, Chen. *Computer Vision – ECCV 2018 Workshops: Munich, Germany, September 8-14, 2018, Proceedings, Part V*. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. Springer-Verlag, 2018. ISBN 978-3-030-11020-8. Available from DOI: 10.1007/978-3-030-11021-5_5.
31. WANG, Xintao; XIE, Liangbin; DONG, Chao; SHAN, Ying. Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data. *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. 2021, pp. 1905–1914. Available from DOI: 10.48550/arXiv.2107.10833.
32. WELCH, Greg; BISHOP, Gary, et al. An introduction to the Kalman Filter. *Nan*. 1995.
33. GOVAERS, Felix. *Introduction and Implementations of the Kalman Filter*. Rijeka: IntechOpen, 2019. ISBN 978-1-83880-537-1. Available from DOI: 10.5772/intechopen.75731.
34. SIMON, Dan. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. USA: John Wiley & Sons, 2006. ISBN 0471708585.
35. JULIER, Simon J.; UHLMANN, Jeffrey K. New extension of the Kalman filter to nonlinear systems. In: KADAR, Ivan (ed.). *Signal Processing, Sensor Fusion, and Target Recognition VI*. SPIE, International Society for Optics and Photonics, 1997, vol. 3068, pp. 182–193. Available from DOI: 10.1117/12.280797.
36. WAN, E. A.; MERWE, R. Van Der. The unscented Kalman filter for nonlinear estimation. *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*. 2000, pp. 153–158. Available from DOI: 10.1109/ASSPCC.2000.882463.
37. SARWINDA, Devvi; PARADISA, Radifa Hilya; BUSTAMAM, Alhadi; ANGGIA, Pinkie. Deep Learning in Image Classification using Residual Network (ResNet) Variants for Detection of Colorectal Cancer. *Procedia Computer Science*. 2021, vol. 179, pp. 423–431. ISSN 1877-0509. Available from DOI: 10.1016/j.procs.2021.01.025. 5th International Conference on Computer Science and Computational Intelligence 2020.
38. JIANG, Yun; CHEN, Li; ZHANG, Hai; XIAO, Xiao. Breast cancer histopathological image classification using convolutional neural networks with small SE-ResNet module. *PLOS ONE*. 2019, vol. 14, no. 3, pp. 1–21. Available from DOI: 10.1371/journal.pone.0214587.

39. HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. abs/1512.03385, Deep Residual Learning for Image Recognition. 2015. Available from DOI: 10.48550/arXiv.1512.03385.
40. ODUSAMI, Modupe; MASKELIŪNAS, Rytis; DAMAŠEVIČIUS, Robertas; KRILAVIČIUS, Tomas. Analysis of Features of Alzheimer's Disease: Detection of Early Stage from Functional Brain Changes in Magnetic Resonance Images Using a Finetuned ResNet18 Network. *Diagnostics*. 2021, vol. 11, no. 6. ISSN 2075-4418. Available from DOI: 10.3390/diagnostics11061071.
41. OTSU, Nobuyuki. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*. 1979, vol. 9, no. 1, pp. 62–66.
42. BRUNELLI, Roberto. *Template Matching Techniques in Computer Vision*. Template Matching as Testing. John Wiley & Sons, Ltd, 2009. ISBN 9780470744055. Available from DOI: 10.1002/9780470744055.ch3.
43. LEWIS, J.P. Fast Template Matching. *Vis. Interface*. 1994, vol. 95.
44. HERRERO, Sonsoles; BESCÓS, Jesús. *Advanced Concepts for Intelligent Vision Systems*. Background Subtraction Techniques: Systematic Evaluation and Comparative Analysis. Ed. by BLANC-TALON, Jacques; PHILIPS, Wilfried; POPESCU, Dan; SCHEUNDERS, Paul. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. ISBN 978-3-642-04697-1.
45. CHEUNG, Sen-ching S.; KAMATH, Chandrika. Robust techniques for background subtraction in urban traffic video. 2004, vol. 5308, pp. 881–892. Available from DOI: 10.1117/12.526886.
46. MÜLLER, Meinard. *Dynamic time warping*. Springer, 2007. ISBN 978-3-540-74047-6. Available from DOI: 10.1007/978-3-540-74048-3_4.
47. BERNDT, Donald J.; CLIFFORD, James. Using dynamic time warping to find patterns in time series. 1994, pp. 359–370. Available from DOI: 10.5555/3000850.3000887.
48. SCHUHMACHER, Dominic; VO, Ba-Tuong; VO, Ba-Ngu. A Consistent Metric for Performance Evaluation of Multi-Object Filters. *IEEE Transactions on Signal Processing*. 2008, vol. 56, no. 8, pp. 3447–3457. Available from DOI: 10.1109/TSP.2008.920469.
49. BEARD, Michael; VO, Ba Tuong; VO, Ba-Ngu. OSPA(2): Using the OSPA metric to evaluate multi-target tracking performance. 2017, pp. 86–91. Available from DOI: 10.1109/ICCAIS.2017.8217598.
50. VU, Tuyet. A New Optimal Subpattern Assignment (OSPA) Metric for Multi-target Filtering. In: 2023. Available from DOI: 10.48550/arXiv.2306.14380.
51. VU, Tuyet; EVANS, Rob. Optimal Subpattern Assignment Metric for Multiple Tracks (OSPAMT Metric). 2019. Available from DOI: 10.48550/arXiv.1808.02242.
52. HARALICK, Robert M.; STERNBERG, Stanley R.; ZHUANG, Xinhua. Image Analysis Using Mathematical Morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1987, vol. PAMI-9, no. 4, pp. 532–550. Available from DOI: 10.1109/TPAMI.1987.4767941.
53. DANIL, Pastukhov. *Advancing Microrobotics for Biomedical Applications through Machine Learning*. 2024. MA thesis. České vysoké učení technické v Praze. Vypočetní a informační centrum.
54. URBANO, Leonardo F.; MASSON, Puneet; VERMILYEA, Matthew; KAM, Moshe. Automatic Tracking and Motility Analysis of Human Sperm in Time-Lapse Images. *IEEE Transactions on Medical Imaging*. 2017, vol. 36, no. 3, pp. 792–801. Available from DOI: 10.1109/TMI.2016.2630720.

55. RONG LI, X.; BAR-SHALOM, Y. Tracking in clutter with nearest neighbor filters: analysis and performance. *IEEE Transactions on Aerospace and Electronic Systems*. 1996, vol. 32, no. 3, pp. 995–1010. Available from DOI: 10.1109/7.532259.
56. KONSTANTINOVA, Pavlina; UDVAREV, Alexander; SEMERDJIEV, Tzvetan. A study of a target tracking algorithm using global nearest neighbor approach. 2003, pp. 290–295. ISBN 9549641333. Available from DOI: 10.1145/973620.973668.
57. BAR-SHALOM, Yaakov; DAUM, Fred; HUANG, Jim. The probabilistic data association filter. *IEEE Control Systems Magazine*. 2009, vol. 29, no. 6, pp. 82–100. Available from DOI: 10.1109/MCS.2009.934469.
58. *Dash Documentation & User Guide / Plotly* — *dash.plotly.com* [<https://dash.plotly.com>]. [N.d.]. [Accessed 18-01-2024].
59. BAR-SHALOM, Yaakov; DAUM, Fred; HUANG, Jim. The probabilistic data association filter. *IEEE Control Systems Magazine*. 2009, vol. 29, no. 6, pp. 82–100. Available from DOI: 10.1109/MCS.2009.934469.

Contents of the attached media

thesis	
├ thesis.pdf	The thesis in a pdf format
├ images	All figures used in this thesis
├ text	Folder containing thesis test files
├ bib-database.bib	Bibliography database
├ ctufit-thesis.cls	L ^A T _E X support file
├ ctufit-thesis.tex	Main thesis L ^A T _E X file
├ horenjak-assignment.pdf	Thesis assignment
package	
├ spermatozoa_tracking_horenjak-0.0.1.tar.gz	Source code of my project
├ spermatozoa_tracking_horenjak-0.0.1-py3-none-any.whl ..	Built distribution of my program
└ videos	Example videos after full analysis