Developing an R package for quantitative analyses of T cell/tumour cell contacts

Masterarbeit
and der Medizinischen Fakultät
der Eberhard Karls Universität Tübingen

vorgelegt von

Quach, Julia

Tübingen, DATUM

Statement

I hereby declare:

- that this thesis is my own work and that I have not made use of any other sources or aids than those referenced.

- that all statements appropriated from other works in letter or in substance have been referenced as such.

- that the thesis submitted has not been the subject of any other examination in essential parts or in full

- that the thesis submitted has not been published in essential parts or in full.

I affirm that I have written the dissertation myself and have not used any sources and aids other than those indicated. The references are presented accorded to standard rules for publication and standard citation guidelines.

Tübingen, Date, Signature

Contents

1. Summary

To do:

- Edit description of R package

- Add results and discussion part to summary

With nearly 10 million deaths in 2020, cancer is one of the leading causes of death worldwide [1][2]. Its high mortality is predominantly caused by resistance against available cancer therapies [c]. To combat this resistance, new therapies have been developed from which immunotherapeutic approaches have demonstrated promising results [c]. Those immunotherapies make use of engineered or innate immune cells, especially T cells, to eliminate malignancies [c]. Nevertheless, solid cancer types often remain resistant due to their immunosuppressive tumour microenvironment [c]. To overcome this hurdle, it is essential to understand the crucial factors promoting successful T cell killing.

To deepen our understanding of T cell killing, a very intuitive method is to *watch* the T cells while interacting with tumour cells by using live-cell imaging methods combined with a range of labelling techniques. For this, the OVA (ovalbumin)-tumour model comes handy as it provides tumour cell lines expressing the OVA antigen and transgenic mice expressing OVA-targeting T cells.

The interaction between T cells and tumour cells is essential for T cell killing. To quantitatively analyse this cell-cell interaction, we developed a work flow which includes the segmentation and tracking of tumour and T cells and the detection of cell-cell contacts. For the latter, we built an R package *cellcontacts* to identify cell-cell contacts and compute their frequency and duration. Furthermore, the R package is capable of mapping the interaction analyses to immunological staining results.

To put it in a nutshell, this thesis' aim is to widen the explanatory power of available live-cell imaging techniques by enabling large-scale analyses of cell interaction and correlation to immunological staining methods. We show that [...]

2. Introduction

In this section, we will introduce the essential role of T cells for tumour cell elimination and afterwards, we will focus on the crucial factors for efficient T cell killing. We will explain the necessity to understand cell dynamics for unravelling T cell killing mechanisms. Eventually, we will address available tools to analyse cell movement.

2.1 T cells are crucial for tumour elimination

To do:

- Mention anti-tumour activity of NK cells

- Compare NK cells vs. T cells

Cancer is a disease in which body cells grow uncontrollably because of inherited or acquired mutations [c]. Those cells form tissue mass, called malignant tumours, that can spread over the body, impede normal body cells and cause disease [c]. Naturally, the immune system is capable to track down and eliminate such abnormal cells by inducing their cell death. For this, T cells, a type of immune cell, use their surface antigen receptors to find irregularities within the surface antigens of abnormal cells. If they find suspicious cells, T cells induce their apoptosis, a well-regulated cell death. In accordance, malfunction of the immune system is strongly correlated to a higher incidence of cancer [ccc][3]. In particular, a decline in T cell mediated immune response can lead to a reduced response against malignant tumour cells and cause disease [3]. In conclusion, if the mechanisms to eliminate abnormal cells are suppressed or fail, cancer cells can form malignant tumours. To prevent the formation of malignant tumours, T cells play a key role as they are capable of eliminating tumour cells.

2.2 CAR T cell therapy exploits the killing activity of T cells

To-do:

- Add some statistics and citations on antibody and CAR T cell therapy

Because T cells are capable of eliminating tumour cells, it is clear that one would like to enhance their killing activity to prevent and treat cancer. To achieve this goal, we need to understand why natural T cells are not always capable of full tumour elimination: Researchers found that tumour cells can escape immuno-surveillance through inhibiting T cell activity, down-regulating their surface antigens, and inducing T cell exhaustion [ccc].
Understanding these immune escape mechanisms, allowed two major breakthroughs in cancer therapy: Check-point inhibitors were discovered which can reverse an inhibition on T cell activity. The release of this brake allows T cells to become active and track down and kill cancer cells [ccc]. This finding

was awarded with a Nobel Prize in 2018 and the consequent therapeutic approaches, called antibody therapies, became a standard in the clinics [c].

Even with a released brake, T cells are not able to track down tumour cells that hide by down-regulating their surface-antigens. To tackle this obstacle, researchers have genetically engineered T cells that carry modified antigen receptors such that they can identify and kill hiding tumour cells [c]. This approach, called chimeric antigen receptor (CAR) T cell therapy, was successful in treating leukaemia cases that were incurable before [cc]. Hence, it was a second major breakthrough in cancer therapy [cc].

Following the great success of CAR T cell therapy to treat haematological malignancies, efforts have sparked to translate this approach to solid tumour entities [ccc]. But in contrary to leukaemia, solid tumours have a often times stiff tumour micro-environment which consists of an extracellular matrix filled different non-cancerous cell types [ccc]. This makes it difficult for T cells and CAR T cells to even enter the tumour tissue before they have a chance to identify and kill the tumour cells [ccc]. Once T cells got through the tumour micro-environment and arrived at the tumour cells, they are usually exhausted and their killing activity is drastically reduced [cc].

To increase the success of CAR T cell therapy in solid malignancies, one could either tackle the tumour micro-environment or further modify CAR T cells such that they become more effective even in an exhaustive environment. There is much research approaching both ideas and adequate methods are required to evaluate the success of those modifications [cc]. In the following section, we will put our focus on the methods to evaluate the success of tumour treatments. Furthermore, we will address why the investigation of cell dynamics is pivotal to evaluate the success of T cell killing.

2.3   Why are cell dynamics relevant?

- Describe tools to evaluate CAR T cell efficiency

- Name the advantages and also shortcomings of static methods (FACS, sequencing, immuno-stainings)

- Name the benefit of cell dynamics

- In particular:

  – Explain why understanding cell-cell interactions are essential for tumour killing

  – What is known about cell-cell interaction and what we would like to know about cell-cell interaction

To observe the success of T cell killing, the most straightforward approach would be to measure the amount of surviving tumour cells before and after therapy. But this is neither sufficient nor does this method reveal the full picture: It does not give an insight into [name more reasons]

Hence, the interaction between T cells, tumour cells and their tumour micro-environment should be observed from different angles which we can roughly classify into static and dynamic methods: Static methods give a detailed insight into the cellular processes at a specific point, including the gene and protein expression profile of cells on a single cell-level, the distribution of cells in tissue and the cells' state in the cell cycle [ccc]. On the other hand, dynamical methods are able to capture immune and tumour cells on a temporal scale. They widen our understanding of cell-cell-interaction, cell growth and killing activity [ccc]. For example, *in vitro* live-cell imaging can visualize how tumour cells move

during tumour growth and invasion and with additional labelling of Ca^{2+}-signalling, we can investigate *when and how* the T cells try to kill tumour cells. [Mention additive cytotoxiciy and maybe also other papers]

Given the different aspects that static and dynamic methods elucidate, it is beneficial to join forces to understand and modify existing tumour therapies. For this, we need comprehensive analysis work flows that combine results from static and dynamic experiments.

In the following section, we will give an overview of existing tools to investigate cell dynamics in live-cell imaging films. In the end, we will give an insight into available tools to combine static with dynamic methods to understand T cell killing mechanisms.

2.4 Tools to investigate cell dynamics

Cell dynamics are equally important as static cell information. In this project, we will first put our focus on methods to quantitatively analyse cell dynamics and cell-cell interaction. Following, we will address methods to connect results on cell dynamics to the findings from static experiments.

Live-cell imaging techniques usually produce a file of around 35 GB file size with around 3000 frames capturing a region of interest of around $x \times x$ μm with around x cells for x hours. After acquiring such a data set, it is evident that it is not feasible to analyse the whole data set manually. Therefore, tools which can automatically detect cells and track their movement are indispensable. In this section, we will introduce available tools for automated cell segmentation and tracking.

2.4.1 Cell segmentation

- Stardist

- CellPose

Before analysing cell movement, cells need to be automatically identified and their boundary accurately delineated, called segmented. For this purpose, several cell segmentation tools have been published [ccc]. Within these tools, StarDist has proved to be reliable for an automatic detection of cell nuclei in a regular roundish shape. The core of StarDist is a convolutional network which has been trained to predict the occurrence of a polygon representing the cell nucleus shape for every pixel. This approach facilitates the correct segmentation of crowded cells which is a typical phenomenon in live-cell imaging although it does not allow the segmentation of ring- or in general odd-shaped cell nuclei [4].

2.4.2 Cell tracking

- Overview and classification

- LAP tracker

- Kalman tracker

Regarding the tracking of cell movement, there exist a broad range of algorithms [cc]. A comprehensive analysis of available cell-tracking algorithms was given by Ulman *et al* in which 21 cell-tracking algorithms were classified regarding their tracking strategy and evaluated based on their performance on up to 13 data sets. In terms of the tracking strategy, the algorithms either used cell contour evolution techniques or cell detection methods. The former requires a spatio-temporal overlap between the corresponding cell regions, thus performing segmentation and tracking in one task, whereas the

latter firstly segments the cells in all frames and afterwards, builds a probabilistic framework to propose temporal associations between the cells. While some tracking algorithms achieved good results, none of them achieved fully correct solutions for all data sets. Hence, a major implication of this report is that the right choice of tracking algorithm heavily depends on the applied imaging modality and the respective data set [5].

[More information on T cell tracking, segmentation + detection and not cell contour evolution]

To facilitate the usage of cell-tracking algorithms, the extensible, open-source Fiji plugin *TrackMate* was published by Tinevez *et al* [6]. Via a graphical user interface, it allows the user to choose a segmentation algorithm and a tracking algorithm for their specific data set in any combination. The available segmentation algorithms included ilastik, Weka, cellpose, MorphoLibJ and StarDist. Moreover, TrackMate allows the import of segmentation results as mask or label images. For the tracking algorithm, TrackMate offers the LAP framework by Jaqaman *et al* for Brownian motion [c] and the Kalman filter for linear motion [c]. Overall, TrackMate offers an open-source framework to combine any segmentation and tracking algorithm for a customized cell-tracking work flow.

2.4.3   Cell-cell contacts

- Available tools

- Our needs -> quantitative information from microscopy images of cultured cells

To gain biological insights from time-lapse microscopy recordings of cell behaviour, ...

How do the dynamics of T cell behaviour correlate to tumour killing efficiency? This is the question central to this thesis.

3. Data sets

We used five live-cell imaging datasets to build and to challenge the R package. All data sets were acquired using a Widefield Microscope DMI8 by Leica Microsystems (Wetzlar, GER) as imaging device equipped with a [...] objective with a resolution of [...] at 5% CO_2.

To set the distance and duration thresholds for cell-cell contact computation, we used three data sets of live-cell imaging films with tumour and T cells. An overview of the data sets is given in Table 3.1.

Table 3.1: Overview of data sets that were used for building the R package

Tumour cell line	T cell line	Frame rate (sec/frame)	Frame size (µm × µm)	Duration (h)	Temperature (°C)
YUMM-OVA (add cell concentration) YUMM					
Ovarian					

To challenge the work flow for cell-cell contact detection, we applied the image analysis pipeline on three additional data sets (see Table 3.2).

Table 3.2: Overview of data sets that were used for challenging the R package

Tumour cell line	T cell line	Frame rate (sec/frame)	Frame size (µm × µm)	Duration (h)	Temperature (°C)
YUMM					
YUMM-OVA					
B16F10					

4. Method

The R package *cellcontacts* was built to process cell tracks acquired from live-cell imaging films of tumour cells with T cells. The package is meant to detect and characterise cell-cell contacts and cell movement for large data sets. Moreover, it relates cell-cell contacts and cell movement to findings from immunological staining techniques. The package requires preprocessing of live-cell imaging films with a cell tracking tool, e.g., TrackMate or Imaris. In this chapter, the package's algorithm to compute cell-cell contacts is described and an analysis pipeline incorporating TrackMate and the R package *cellcontacts* is laid out.

4.1   Algorithm to compute cell-cell contacts

In this section, we formally define cell-cell contacts. Afterwards, we describe the algorithm to compute cell-cell contacts.

Formalizing the data set and defining cell-cell contacts   Let $n \in \mathbb{N}$ be the the the last time point in the live-cell imaging film, equivalent to the maximum frame number. Let M_i be the set of all cells at one time point $i \in \{1, \ldots, n\}$ with $0 \in M_i$ representing a non-registered cell. Let $U_i \subset M_i$ be the set of tumour cells and $V_i \subset M_i$ the set of T cells at a time point $i \in \{1, \ldots, n\}$ such that $M_i = U_i \cup V_i$.

A tumour cell shall be represented as $u = (u_1, \ldots, u_n) \in U_1 \times \cdots \times U_n$ such that for each time point $i \in \{1, \ldots, n\}$, the element u_i either represents the xy-position of the tumour cell at this time point or equals 0 if the respective tumour cell was not registered at this time point. Analogously, we define T cells as $v = (v_1, \ldots, v_n) \in V_1 \times \cdots \times V_n$.

Let $d_i : U_i \times V_i \to \mathbb{R}$ be a metric that describes the euclidean distances between the segmented area of tumour cell and a T cell for a given time point $i \in \{1, \ldots, n\}$ and if either the tumour cell or the T cell was not registered during that time point, we define $d_i(u_i, v_i) = \infty$.

Let $x \in \mathbb{R}$ be a distance threshold in µm and $m \in \mathbb{N}$ a duration threshold given as number of frames. We define that a tumour cell $u \in U_1 \times \cdots \times U_n$ and a T cell $v \in V_1 \times \cdots \times V_n$ are *in contact* if and only if their distance is below the distance threshold for m consecutive frames:

$$u \text{ and } v \text{ in contact} \quad \Leftrightarrow \quad \exists\, j \in \{1, \ldots, n\}\ \forall i \in \{j, \ldots, j+m\} : d_i(u_i, v_i) \leq x$$

Algorithm (cell-cell contact computation)   To compute cell-cell contacts, we need to find all tumour cell/T cell pairs (u, v) that maintain a distance below threshold for m consecutive frames. As the number of distance calculations grow quadratically with an increasing frame size of the film, we added preprocessing steps to fasten the computation time:

Algorithm to compute cell-cell contacts

Input:

- Cell tracks and cell areas of tumour cells and T cells at all time points.

- A distance threshold x.

- A minimum duration threshold expressed as a number of frames m.

Output: Data frame with columns

- Names of cell pairs that maintain a distance below threshold for at least m consecutive frames.

- Time points during a contact.

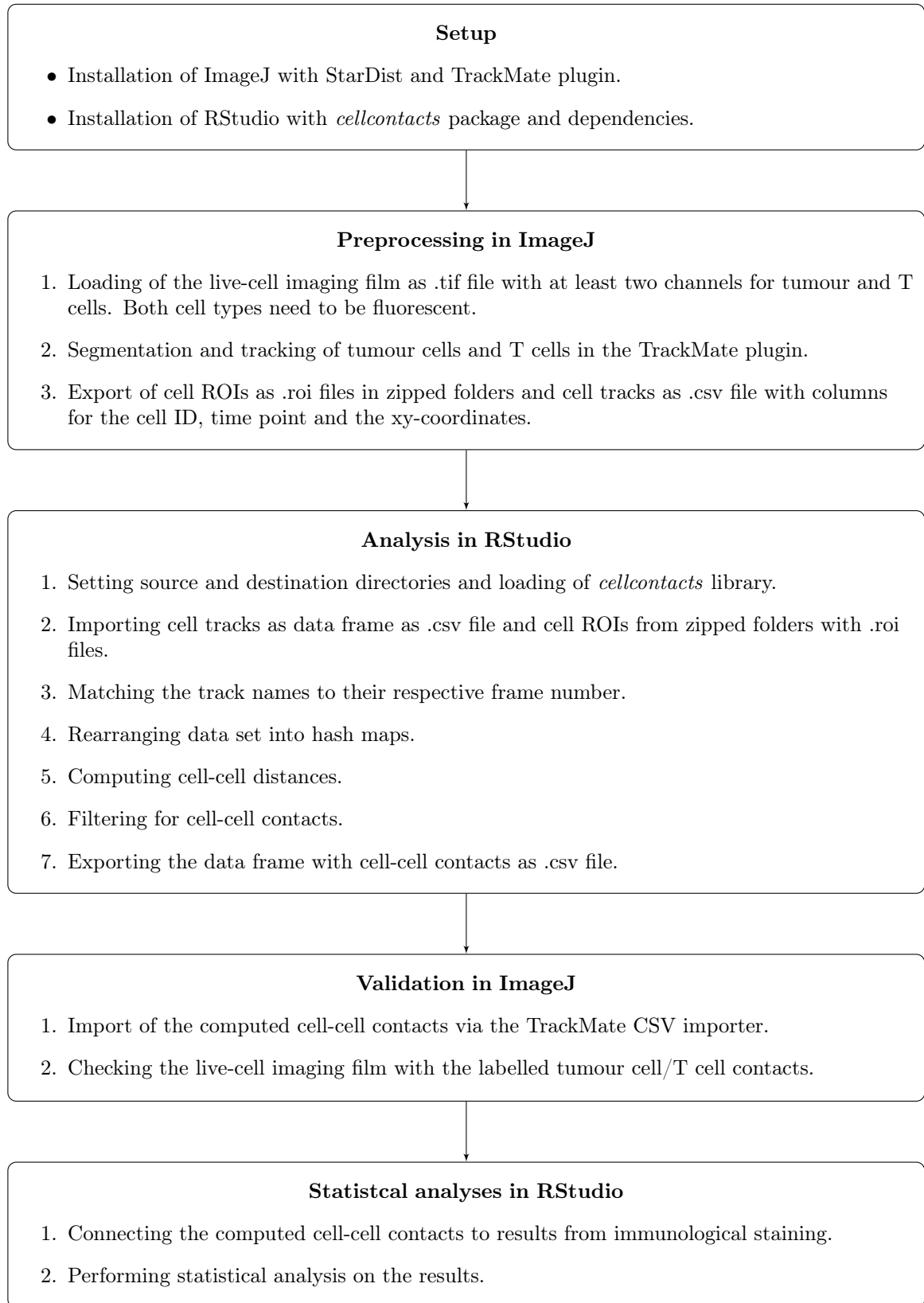- Cell-cell distances during a contact.

Instructions:

1. Preprocessing

    1.1 Compute the center points of all tumour and T cells.

    1.2 Create a grid in which columns and rows have a distance of 100 μm.

    1.3 Sort the computed cell center points into the grid.

2. Distance computation

    2.1 Compute all possible tumour cell/T cell pairs for each time point.

    2.2 For each pair, save the information whether their grid coordinates are farther away than two rows or columns.

    2.3 If the grid coordinates are at least in neighbouring rows and columns, compute and save the distance of the segmented tumour and T cell area.

3. Filtering for contacts

    3.1 For all time points, discard cell pairs that have a distance above the threshold x.

    3.2 For the remaining pairs, discard cell pairs that maintain a distance below threshold for less than m frames.

    3.3 Return the pair names, their distance and their time point information as a data frame.

4.2   Proposed image analysis pipeline

After describing the algorithm, this section is a detailed guide through the image analysis work flow. An overview of the steps in the work flow is given in Figure 4.1. An exemplary R script is provided on the GitHub page of the R package and a detailed tutorial of the workflow is provided in the supplement (add link).

Figure 4.1: Overview of work flow to detect cell-cell contacts in live-cell imaging films using the R package *cellcontacts*.

Setup

- Installation of ImageJ with StarDist and TrackMate plugin.

- Installation of RStudio with *cellcontacts* package and dependencies.

Preprocessing in ImageJ

1. Loading of the live-cell imaging film as .tif file with at least two channels for tumour and T cells. Both cell types need to be fluorescent.

2. Segmentation and tracking of tumour cells and T cells in the TrackMate plugin.

3. Export of cell ROIs as .roi files in zipped folders and cell tracks as .csv file with columns for the cell ID, time point and the xy-coordinates.

Analysis in RStudio

1. Setting source and destination directories and loading of *cellcontacts* library.

2. Importing cell tracks as data frame as .csv file and cell ROIs from zipped folders with .roi files.

3. Matching the track names to their respective frame number.

4. Rearranging data set into hash maps.

5. Computing cell-cell distances.

6. Filtering for cell-cell contacts.

7. Exporting the data frame with cell-cell contacts as .csv file.

Validation in ImageJ

1. Import of the computed cell-cell contacts via the TrackMate CSV importer.

2. Checking the live-cell imaging film with the labelled tumour cell/T cell contacts.

Statistcal analyses in RStudio

1. Connecting the computed cell-cell contacts to results from immunological staining.

2. Performing statistical analysis on the results.

Requirements   To use this work flow on a live-cell imaging film, the imaged cells need to be fluorescent to enable their correct segmentation. The frame rate usually needs to be 90 s/frame or faster to ensure correct tracking of fast-moving T cells.

To process the live-cell imaging films, installation of the open source package Fiji (also known as ImageJ) with the plugins StarDist for cell segmentation and TrackMate for cell tracking is required. To use the *cellcontacts* package, the programming language R is needed, preferably in the RStudio environment.

Installation   The latest development version of *cellcontacts* can be installed in the R environment after getting access to the *cellcontacts* project on GitHub:

```
1   devtools::install_github( "juliaquach02/cellcontacts" )
```

The package can be loaded via:

```
1   library( cellcontacts )
```

Preprocessing in ImageJ   To extract the image features, we load the live-cell imaging film as .tif-file into Fiji where we can use the TrackMate plugin for segmentation and tracking. Two separate TrackMate session are needed to analyse T cells and tumour cells. During segmentation and tracking, the plugin offers the possibility to adjust parameters and to manually correct segmented cells or cell tracks if necessary. After segmentation and tracking, the cell areas of both data sets are renamed in the TrackMate GUI by their assigned track name and an unique index for the respective track.

We export cell tracks as .csv file with columns indicating the cell ID, time and coordinates at each measured time point. The renamed cells region of interests (ROIs) are exported as several zipped folders with around thousand .roi-files per folder. Saving the ROIs into several zipped folders will enable parallel, hence faster, import to RStudio. For the saving process, we provide an ImageJ macro on the package's GitHub page.

Apart from exporting cell tracks and cell ROIs, the TrackMate sessions can be saved as .xml file. This allows reloading the session with its segmentation and tracking parameters. An overview of the data structure after this preprocessing step is given in table 4.1.

Table 4.1: This is an overview of files that should be available for both, tumour and T cells, after preprocessing a live-cell imaging film.

Folder	File type	Content
Tumour/T cells	.zip-file or folder with .zip-files	.roi-files that represent the area of the segmented tumour/T cells with file names in the format Track_[id]_[index].roi
	.csv file	Data frame in which each row represents one cell in one frame. The data frame should contain columns for the track names (in the format Track_[id]_[index].roi), the x-, and y-coordinate and the frame number.
	.xml file	This file is optional. It contains information about the TrackMate session such that the session can be reloaded.

Setting directories and loading the track data   After acquiring all necessary files, we switch to RStudio to use the *cellcontacts* package. For this, we set the directories to import the data. Cell

tracks saved as .csv files are loaded as data frame using the base R function `read.table()`. To load the zip-files containing the ROIs, we use the `read.ijzip()` function from the `RImageJROI` package.

Inserting data into hash maps We provide functions to add the data set into hash maps for an efficient retrieval of our data during computation (see supplement). To decrease the computation time of cell-cell distances, we sort the cell ROIs into a 2D grid to get rough estimates of their position (add a schematic figure). To create such a grid, we provide the function `create_grid()`.

Using the hash maps, we can easily access the track names of all cells at a given time point to compute possible tumour cell/T cell pairs. Moreover, we can retrieve the center points and position of the cells in the 2D grid before computing cell-cell distances. Calling the approximate cell position in the grid avoids the computation of euclidean distances for cell pairs that are clearly farther away than the distance threshold.

Computing cell-cell distances We compute cell-cell distances to filter for cell-cell contacts. For this, we provide the function `compute_distTimepoint_wGrid()`. This function computes all possible tumour cell/T cell pairs for a given time point. For each pair, the function checks the rough position of both cells using their position in the 2D grid. If the cells are at least in neighbouring columns and rows of the grid, the euclidean distance between the cell ROI is computed (add a schematic figure). The output of the function is a data frame listing all possible cell pairs with either the distance of the pair or a remark that the distance is substantially larger than the distance threshold for a cell-cell contact.

Filtering for cell-cell contacts To filter the cell pairs involved in a cell-cell contact, we only keep pairs that maintain a distance below a distance threshold for a minimum duration of frames. The distance and duration threshold can be set manually. As a result, we obtain a list of tumor cell/T cell pairs and each list entry represents one pair and contains the columns *time point* and *distances*.
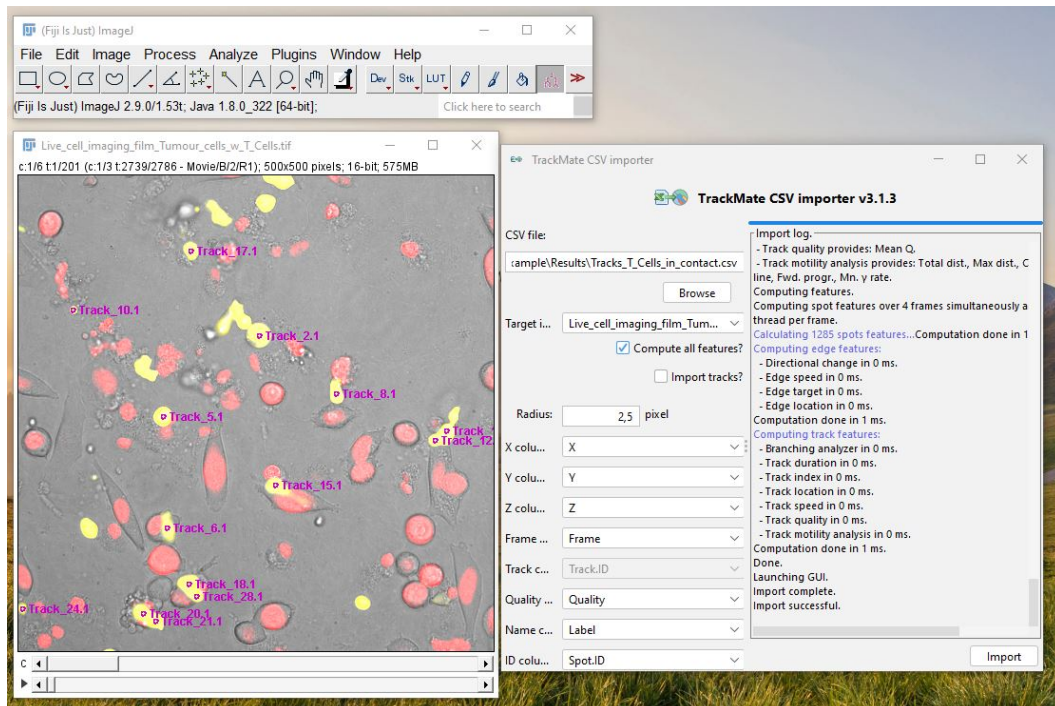

Figure 4.2: Via the TrackMate CSV Importer GUI, we can load our computed cell-cell contacts back into a TrackMate session.

Correlating results to immunological staining results   We often want to connect the dynamic information of the cell tracks to the static information of the same cells from immunological staining. For this, we provide the function `match_to_endpoint_ROIs()` to match the cell tracks to the corresponding cell ROIs at the end point. The function `add_meas_to_matches()` can add the measurements of the signal intensity to the mapping of cell tracks to endpoint ROIs (add a schematic figure?).

Computing characteristics of cell tracks   To get a thorough picture of the mechanisms behind T cell killing, we observe additional aspects of T cell dynamics including speed, directionality and persistence of the cells. To compute and to cluster these cell track characteristics, we use the package *celltrackR* by Ingel Wortel et al [7]. The documentation of this package can be found at `https://ingewortel.github.io/celltrackR/`.

Exporting and visualization of results   To check of the computed cell-cell contacts, we export the track information of the T cells that were in contact as .csv-file using `prepareExportContacts()` and `write.csv()`. The exported files are loaded into TrackMate via ImageJ > Plugins > Tracking > TrackMate CSV Importer. Using the .csv file, TrackMate labels the T cells in the .tif-file only during a contact (see Figure 4.2). This allows us to revisit the live-cell imaging film and to check whether cell-cell contacts are correctly computed.

4.3   Quality control

Segmentation and track quality   To identify cell-cell contacts, correct segmentation and tracking data are pivotal. Already during the analysis with TrackMate, we filter the segmented cells and their corresponding tracks by the spot size, the mean fluorescence intensity, the track length and other criteria [6]. After importing the data into R, we detect and correct for artefacts, e.g., double tracking or drift, via track angle analyses from the package *celltrackR* by Ingel Wortel et al [7].

Evaluation metric   To assess the reliability of cell contact detection, we manually observe cell-cell contacts in the films and afterwards, we evaluate the accordance of the computed cell-cell contacts to the manually obtained results. In addition, we import the detected cell-cell contacts back into TrackMate such that cells are labelled in the film during a cell-cell contact.

5. Results

5.1 Accuracy of cell segmentation and cell tracking

Discuss

- Accuracy of StarDist (especially on mitotic cells and T cells in a non-regular shape)

- Accuracy of LAP tracker (include photos)?

5.2 Accuracy of cell-cell contact detection

- Detection based on duration of proximity (not consideration of cell shape)

- Allows simultaneous contact of one T cell to multiple tumour cells?

- Figure: Boxplot to demonstrate deviation from ground truth?

5.3 Evaluating computation time

- Compute complexity for the approach with and without grid

- Include benchmarking results

5.4 Detecting tumour cell/T cell contacts

- Olis Data with ground truth

- Olis Data on large scale

- Olis Data - Compare with and without OVA

- Olis Data - Connect to Fix-While-Filming

- Steffens Data with ground truth

- Steffens Data on large scale

6. Discussion

6.1   Reliability and potential

- Discuss results (comparison to ground truth)

6.2   Compared methods

- Imaris

- CellProfiler

- Napari

6.3   Biological interpretation and perspective

- Interpretation of results

- Perspective of potential experiments/analyses

7. Conclusion

In this thesis, we set up a framework for an automatic detection of cell-cell contacts in live-cell imaging films. This enables a quantitative analysis of dynamic cell data. The framework relies on previous segmentation and tracking of cells using well-established open-source tools. If the segmentation and tracking is accurate, the cell-cell contact detection based on the duration of cell proximity proved to be

- Created a framework

- R package automatizes cell-cell contact detection -> comparability (observer bias avoided)

Supplement

- Link to R package on GitHub

- Link to movies?

References

[1] WHO's Global Health Estimates. The top 10 causes of death, 2020-12-09. URL `https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death`. [Last access 2023-04-25].

[2] WHO Fact Sheets. Cancer, 2022-02-03. URL `https://www.who.int/news-room/fact-sheets/detail/cancer`. [Last access 2023-04-25].

[3] Helmout Modjtahedi and Ailsa Clarke. The Immune System, pages 79–98. 2007. ISBN 9780470988121. doi: https://doi.org/10.1002/9780470988121.ch8. URL `https://doi.org/10.1002/9780470988121.ch8`.

[4] Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. Cell detection with star-convex polygons. In Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger, editors, Medical Image Computing and Computer Assisted Intervention – MICCAI 2018, pages 265–273. Springer International Publishing. ISBN 978-3-030-00934-2.

[5] Vladimír Ulman, Martin Maška, Klas E. G. Magnusson, Olaf Ronneberger, Carsten Haubold, Nathalie Harder, Pavel Matula, Petr Matula, David Svoboda, Miroslav Radojevic, Ihor Smal, Karl Rohr, Joakim Jaldén, Helen M. Blau, Oleh Dzyubachyk, Boudewijn Lelieveldt, Pengdong Xiao, Yuexiang Li, Siu-Yeung Cho, Alexandre C. Dufour, Jean-Christophe Olivo-Marin, Constantino C. Reyes-Aldasoro, Jose A. Solis-Lemus, Robert Bensch, Thomas Brox, Johannes Stegmaier, Ralf Mikut, Steffen Wolf, Fred A. Hamprecht, Tiago Esteves, Pedro Quelhas, Ömer Demirel, Lars Malmström, Florian Jug, Pavel Tomancak, Erik Meijering, Arrate Muñoz-Barrutia, Michal Kozubek, and Carlos Ortiz-de Solorzano. An objective comparison of cell-tracking algorithms. Nature Methods, 14(12):1141–1152, 2017. ISSN 1548-7105. doi: 10.1038/nmeth.4473. URL `https://doi.org/10.1038/nmeth.4473`.

[6] Jean-Yves Tinevez, Nick Perry, Johannes Schindelin, Genevieve M. Hoopes, Gregory D. Reynolds, Emmanuel Laplantine, Sebastian Y. Bednarek, Spencer L. Shorte, and Kevin W. Eliceiri. Trackmate: An open and extensible platform for single-particle tracking. Methods, 115:80–90, 2017. ISSN 1046-2023. doi: https://doi.org/10.1016/j.ymeth.2016.09.016. URL `https://www.sciencedirect.com/science/article/pii/S1046202316303346`.

[7] Inge M. N. Wortel, Annie Y. Liu, Katharina Dannenberg, Jeffrey C. Berry, Mark J. Miller, and Johannes Textor. Celltrackr: An r package for fast and flexible analysis of immune cell migration data. ImmunoInformatics, 1-2:100003, 2021. ISSN 2667-1190. doi: https://doi.org/10.1016/j.immuno.2021.100003. URL `https://www.sciencedirect.com/science/article/pii/S2667119021000033`.