Detection of Nucleoli Using ImageJ

Nico Hochberger

November 2014

Contents

1 Introduction

1.1 Preface

1.2 Objective

1.3 Motivation

Currently nucleoli are detected using an application called CellProfiler[1]. While this application yields reliable results, it also takes pretty long to complete the analysis. Runtimes up to 45 seconds are common. Due to the fact that CellProfiler is a very general approach, applicable to a large variety of tasks related to detecting nuclei and nucleoli, the results of its analysis have to be checked manually to reduce the amount of false-positives. In order to analyze the cells, they need to be taken out of the incubator. Yet, outside the incubator the cells can only be kept alive for a limited timespan. Considering this, time is a valuable resource and must not be wasted by using a too general approach. Consequently, this leads to a more specialized way of analyzing the cells, which does not do all the analysis performed by CellProfiler, but on the other hand is much faster and thus helps to prevent cells dying before the analysis is finished.

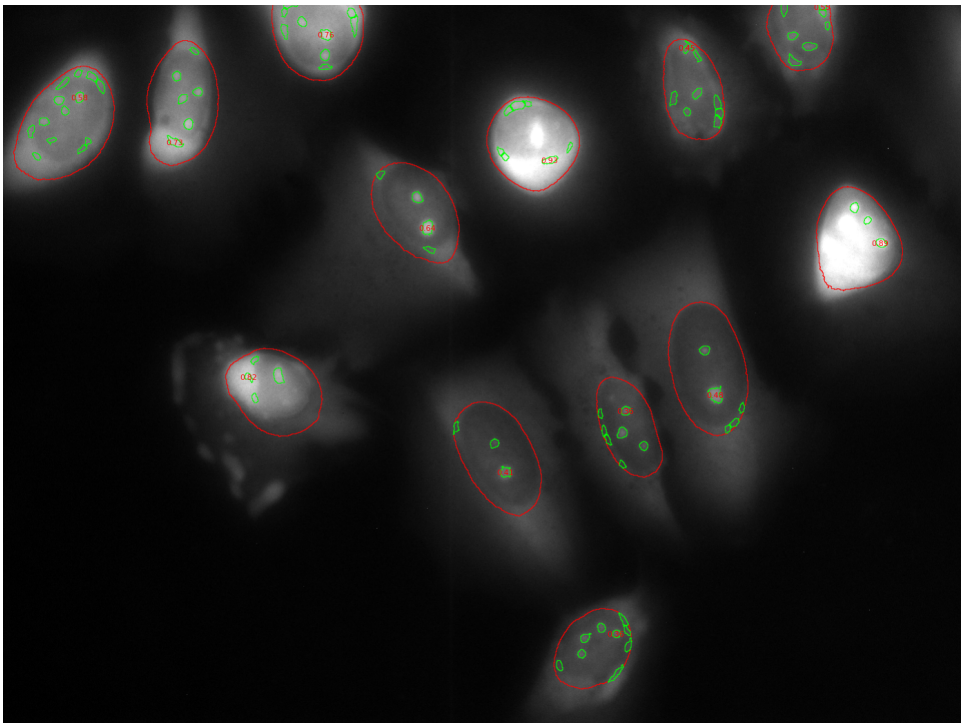

Figure 1: Example analysis performed with CellProfiler

[1] http://www.cellprofiler.org

2 Design and Implementation

2.1 Requirements

The application has to meet the following requirements:

- **Reliable nucleoli detection:** Stable and reliable detection of nucleoli is the main purpose of the application. Hence, it is supposed to detect at least 75% of the nucleoli CellProfiler can detect. This includes a certain degree of stability concerning fuzzy pictures or pictures with inequally distributed or diffuse brightnes.

- **Fast analysis:** As the application is tailored to this single task, it is expected to detect a suitable amount of nucleoli in only a small fraction of the time required by CellProfiler. The topmost time that the application may require to complete the analysis of one image is five seconds.

- **Fallback in case of empty nuclei:** Each nucleus is expected to containg at least one nucleolus. Yet, this expectation cannot always be fulfilled due to potentially damaged nuclei, fuzzy images, or other reasons. In this case, the center of the nucleus has to be provided as fallback target.

- **Visualizer:** In order to quickly check the results directly after running the analysis and to provide a way to quickly present the results to a potential audience, the application has to provide the possibility to be configured so that it shows the results as an image. This image has to contain all detected neuleoli targets, fallback targets and the regions of interest, e.g. the nuclei.

- **Versatility:** Since the appearance of different specimen can vary in various ways, all analysis parameters have to be configurable. Among others, this includes the minimum and maximum sizes of nuclei and nucleoli. The configuration is supposed to be achieved via an understandable, interchangeable, text-based file[2].

- **Statistics:** To determine the most suitable parameters for different kinds of specimen, another feature may be configured. This statistics feature has to include:

  - The amount of detected nuclei
  - The amount of detected nucleoli
  - Nuclei to nucleoli ratio as percentage

[2]Configurable parameters are explained in detail in the User's Manual section

  - The distance of each detected nucleolus to the center point of the containing nuclei and the average distance in pixels
  - The area of each detected nucleus and the average area in square pixels
  - The area of each detected nucleolus and the average area in square pixels

- **Serialization of the results:** All results have to be stored in their accordant files in a subfolder *results* of the folder containing the original data. In the following, the accordant formats and files are described.

  - **Targets:** Real targets and fallback targets sre to be saved in one txt-file named *targets_⟨timestamp⟩.txt* in the following format:

    Listing 1: Format of results txt-file

```
# nucleoli targets
<target number> : [<x-coord>, <y-coord>]
...
# targets in center of empty nuclei
<target number> : [<x-coord>, <y-coord>]
...
```

    Example:

    Listing 2: Example of results file

```
# nucleoli targets
1 : [468, 43]
2 : [1183, 14]
# targets in center of empty nuclei
3 : [87, 174]
4 : [769, 198]
```

  - **Statistics:** The statistics as mentioned above have to be stored in a txt-file named *statistics_⟨timestamp⟩.txt*.

    Example:

    Listing 3: Example of statistics file

```
Nuclei count:
        13

Target count:
        9
```

```
Nuclei to target ratio:
        69.23%

Mean distance:
        25.24 pixels

Distances [pixels]:
        12.419742348374221
        44.04543109109048
        ...

Mean area of detected nuclei:
        12070.77 pixel^2

Nucleus areas [pixels^2]:
        9503.0
        7386.0
        ...

Mean area of detected nucleoli:
        149.06 pixel^2

Nucleolus areas (of all detected nucleoli)

        240.0
        170.0
        ...
```

– **Result image:** The image as it would be displayed by the visualizer has to be stored to a file named *targets_⟨timestamp⟩.⟨original image filetype⟩*.

Example:

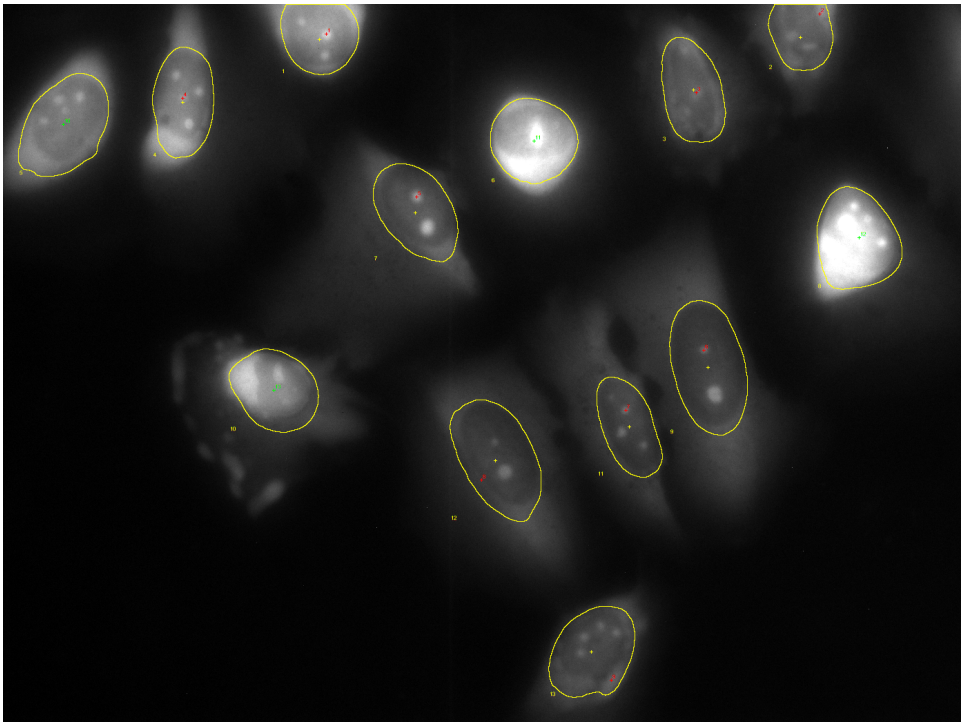

Figure 2: Example of the image containing the results

3 User's Manual

3.1 System Requirements

3.2 Starting the Application

3.3 Configuration

3.3.1 General Parameters

3.3.2 Improved Image Detection Parameters

3.3.3 Structure of Files and Folders

4 Conclusion and Prospect

4.1 Conclusion

4.2 Prospect

References