

**Wymagający złamania wierszy  
tytuł pracy w języku polskim**

(English title)

Artur Rosa

Praca licencjacka

**Promotor:** dr Andrzej Łukaszewski

Uniwersytet Wrocławski  
Wydział Matematyki i Informatyki  
Instytut Informatyki

23 lipca 2020



## Streszczenie

Polskie streszczenie

---

English abstract



# Spis treści

<b>1. Wprowadzenie</b>	<b>7</b>
1.1. Wstęp . . . . .	7
1.2. Background . . . . .	7
1.2.1. Jakich danych i do czego potrzebują biolodzy . . . . .	7
1.2.2. Pozyskiwanie danych . . . . .	7
<b>2. Wykrywanie i śledzenie komórek</b>	<b>9</b>
2.1. Opis problemu . . . . .	9
2.1.1. Obrazy wejściowe . . . . .	9
2.1.2. Pożądaný efekt . . . . .	9
2.2. Powiązane prace . . . . .	9
2.3. Wykrywanie komórek . . . . .	9
2.3.1. Wstęp . . . . .	9
2.3.2. Dane wejściowe . . . . .	9
2.3.3. Wybór kanału i wstępne przetwarzanie obrazu . . . . .	10
2.3.4. Szkieletyzacja i wstępna detekcja komórek . . . . .	10
2.3.5. Rozwiązywanie konfliktów . . . . .	10
2.3.6. Korekta końcówek . . . . .	10
2.4. Śledzenie komórek w czasie . . . . .	10
2.5. Interakcja ze strony użytkownika . . . . .	10
<b>3. Opis implementacji</b>	<b>11</b>
3.1. Kod źródłowy . . . . .	11

3.2. Kompilacja i uruchomienie . . . . .	11
3.3. Struktury danych . . . . .	11
3.4. Architektura . . . . .	11
3.5. Opis wykorzystanego API ImageJ . . . . .	11
3.6. Błędy w implementacji ImageJ i sposoby na ich obejście . . . . .	11
3.7. Opis sposobu dalszego rozwoju, interfejsy . . . . .	11
<b>4. Zakończenie</b>	<b>13</b>
4.1. Podsumowanie . . . . .	13
4.2. Ograniczenia wynikające z zastosowanych metod . . . . .	13
4.3. Dalszy rozwój . . . . .	13
<b>Bibliografia</b>	<b>15</b>

## Rozdział 1.

# Wprowadzenie

### 1.1. Wstęp

### 1.2. Background

#### 1.2.1. Jakich danych i do czego potrzebują biolodzy

#### 1.2.2. Pozyskiwanie danych





## Rozdział 2.

# Wykrywanie i śledzenie komórek

### 2.1. Opis problemu

#### 2.1.1. Obrazy wejściowe

#### 2.1.2. Pożądany efekt

### 2.2. Powiązane prace

### 2.3. Wykrywanie komórek

#### 2.3.1. Wstęp

Problem opisany w sekcji 2.1. zdefiniowany jest dla nagrań spod mikroskopu. Postanowiłem jednak najpierw rozwiązać podobny problem, ale zdefiniowany dla pojedynczego obrazu. Rozwiązanie tego problemu mogłoby z łatwością zostać uogólnione na stos obrazów (nagranie). W tym rozdziale opiszę rozwiązanie uproszczonego problemu: oznaczanie komórek widocznych na pojedynczym obrazie.

#### 2.3.2. Dane wejściowe

Oznaczenie wszystkich komórek widocznych na obrazie można rozłożyć na dwa osobne problemy:

1. Określenie liczby oraz lokalizacji poszczególnych komórek
2. Oznaczenie kręgosłupów komórek.

W niniejszej pracy zdecydowałem się nie rozwiązywać automatycznie pierwszego problemu. Zamiast tego użytkownik zobowiązany jest ręcznie zaznaczyć dokładnie jeden punkt wewnątrz każdej komórki widocznej na obrazie. Wymóg ten dotyczy tylko pierwszej klatki nagrania, co opiszę dokładniej w dalszej części pracy (2.4.).

Danymi wejściowymi są zatem obraz **I** oraz zbiór punktów **P** lokalizujących komórki.

### 2.3.3. Wybór kanału i wstępne przetwarzanie obrazu

Obraz wejściowy składa się z dwóch podstawowych kanałów, które opisałem wcześniej. Chcąc jak najdokładniej oznaczyć początek i koniec komórki, a także miejsca ich podziału, postanowiłem wybrać kanał, który zawiera wyraźną informację o krawędziach w tych miejscach. O ile kanał z fluorescencją mógłby bardzo dobrze sprawdzić się do określenia liczby oraz lokalizacji poszczególnych komórek (w przypadku ich niewielkiej liczby), o tyle drugi kanał zawiera dużo dokładniejszą informację na temat krawędzi komórek.

Preprocessing (shape index), ale też zalety i wady różnych kanałów obrazów wejściowych

### 2.3.4. Szkieletyzacja i wstępna detekcja komórek

Wydzielanie kręgosłupów jako komórek

### 2.3.5. Rozwiązywanie konfliktów

### 2.3.6. Korekta końcówek

Poprawianie końcówek (dociąganie do krawędzi)

## 2.4. Śledzenie komórek w czasie

Opis jednej iteracji + sprawdzenia czy komórka nie powinna zostać podzielona

## 2.5. Interakcja ze strony użytkownika

## Rozdział 3.

# Opis implementacji

3.1. Kod źródłowy

3.2. Kompilacja i uruchomienie

3.3. Struktury danych

3.4. Architektura

3.5. Opis wykorzystanego API ImageJ

3.6. Błędy w implementacji ImageJ i sposoby na ich obejście

3.7. Opis sposobu dalszego rozwoju, interfejsy



## Rozdział 4.

# Zakończenie

4.1. Podsumowanie

4.2. Ograniczenia wynikające z zastosowanych metod

4.3. Dalszy rozwój



# Bibliografia

- [1] Example Women, *A Document Preparation System*. Addison Wesley, Massachusetts, 2nd Edition, 1994.