A Novel Method for Classification and Characterization of Urothelium Cell Culture Exposed to the Different PPARg Agonists

Yuanxiang WU

MSc by Research

University of York

Electronic Engineering

August 2019

Acknowledgements

First, I would like to express my foremost gratitude to my first supervisor, Prof. Stephen Smith. He has been very supportive and provide lots of helpful advices since the first day I started my MSc by research study. I could not have imagined having a better advisor and mentor for my MSc by research study. I would also like to thank my second supervisor, Dr. Steven Johnson for the many useful suggestions and assistance that he has given on my academic researches.

My thanks also go to Prof. Jenny Southgate and Mr. Zhen Liu (a PhD student under Jenny’s guidance) for providing me a lot of support and many insightful comments and encouragement.

Finally, I must express my very profound gratitude to my parents. This accomplishment would not have been possible without them.

Thank you all!

Declaration of Authorship

I declare that this thesis is a presentation of original work and I am the sole author. This work has not previously been presented for an award at this, or any other, University. All sources are acknowledged as references. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Abstract

The main purpose of the thesis is to classify and characterize urothelium cell cultures under PPARg activator/inhibitor. In this project, the raw data are obtained from videos of three different cell cultures: TZ (PPARg activator), T0070907 (PPARg inhibitor) and a control culture. A cell tracking program based on the OpenCV computer vision library is applied to the videos to generate a dataset consisting of x,y coordinates of the tracked cells. The numerical computing environment MATLAB® is subsequently used to filter the data and extract features, which were applied to machine learning algorithms to classify the cell cultures. Results obtained indicate that the TZ/T0070907 addition can cause a change in the average behavior of cells, such as the number of cells in the culture, the speed of cells and the average clump size of cells. The work also demonstrates that there is a difference in single cell behavior among different cultures. In summary, it is proposed that the approach described in this project provides a potential way of analyzing the average behavior of cells in different cultures.

Content

1. Introduction

With the development of biological technology, the study of cell behavior is becoming increasingly important to biological research. The researchers studying cell cultures have great enthusiasm for understanding intercellular behaviors. Due to the restriction in existing biological techniques, current studies only focus on the number of cells in a culture and has little capability of detecting any intercellular behaviors that are not easily obtained, such as the speed of cells within time-lapse spectroscopy videos. In this case, many cell tracking software systems have been developed to extract more information from the videos in addition to the number of cells. However, most of the cell tracking software systems are either unsuitable for this specific project or have an unstable cell tracking accuracy. To help resolve this problem, this project provides a method of analyzing the output of such cell tracking software systems and extracting useful information from the resulting data sets. The MATLAB® numerical computing environment was applied to preprocess the data and extract feature so that the output data will be more reliable and provide more useful information. Machine learning algorithms were also applied to understand the relationships between features extracted and better characterize the different cell cultures.

The thesis is composed of five chapters. The first chapter is an introduction to the whole project. The second chapter is a literature review which provides the research background. The literature review consists of three parts, cell culture and characterization which provides the biological evidence, cell tracking techniques and finally, machine learning algorithms which may be useful for the project. Chapter 3 describes the methodology adopted in work, Chapter 4 presents the results and data analysis undertaken and the final chapter presents the conclusions and suggestions for future development.

2. Literature review

2.1 Cell culture and characterization

2.1.1 The Cell

Cells are usually referred to as the 'building blocks of life', a cell is the smallest unit of life and forms the elementary structural and biological unit of every identified living organism.

Cells are extremely micro in shape and have a variety of shapes.[1] Cells are mainly comprised of the nucleus, cytoplasm and membrane. And the cells contains many biomolecules like proteins and nucleic acids.[2] In general, all living things except viruses are known to be made up of cells, however, virus life must also be embodied in cells.[2] Most microorganisms such as bacteria and protozoa consist of a single cell, which is called single-celled organism, while plants and animals are made up of multiple cells, which is called multicellular organisms.[3][4]

Robert Hooke discovered cells in 1665. [6][7] Matthias Jakob Schleiden and Theodor Schwann put forward the cell theory in 1839 which suggests that: all organisms are formed by cells; cells are the fundamental unit of structure and function in all living organisms, and all cells come from pre-existing cells.[8]

Cells can be divided into prokaryotic cells and eukaryotic cells. The reproduction of cells is achieved by the division of cells. The continuous division of cells starts from the completion of one division until the completion of the next cell cycle.[9][10] There are four ways of cell division. The eukaryotic cells have three ways which are mitosis, amitosis, meiosis. Mitosis and amitosis two are somatic cell division. And the prokaryotic cell divides in binary fission.[9][10][11]

2.1.2 Urothelium

The urothelium is a kind of transitional epithelium. The transitional epithelium lines in the organs of the lower urinary system, including the bladder. It is formed by three different cell layers. Urothelium is a mitotic resting tissue with very low turnover and high regeneration potential [15]. The surface of the urothelium cells have multiple unbalanced plaques of asymmetric unit membrane (AUM), which forms a transcellular urinary barrier [16]. Urothelium-specific uroplakin (UPK) genes express a certain mark for the urothelium cell differentiation and form the AUM [17].

The urothelium cell has great intercellular permeability due to the tight connection between proteins which is made up of cytoplasmic plaque proteins linking to actin cytoskeleton, and integral transmembrane proteins of the pore. In human urothelium, different stages of differentiation are well expressed according to previous research. For example, claudin 4 and claudin 5 expressed at terminal differentiation, claudin 6 and claudin 7 are associated with intermediate cells. Claudins are a family of proteins which are the most important components of the tight junctions. For all epithelial cells, Cytokeratins(CK) are keratin proteins found in the intracytoplasmic cytoskeleton. There are 20 subtypes of CK [18]. The results of immunohistochemical analysis of normal human urothelium showed that CK13 was expressed in basal and intermediate layers, while CK20 was only expressed in surface cells. CK13 expression may be converted to CK14 expression in squamous metaplasia [19].

2.1.3 Peroxisome proliferator-activated receptors (PPAR)

In the field of molecular biology, the peroxisome proliferator-activated receptors (PPARs) are a group of nuclear receptor proteins that function as transcription factors regulating the expression of genes.[29] PPARs play essential roles in the regulation of cellular differentiation, development, and metabolism (carbohydrate, lipid, protein) [30], and tumorigenesis[31] of higher organisms. The urothelium is a potential target tissue of PPAR agonists.

There are three PPAR subtypes, PPARa, PPARb/d and PPARg. These three different subtypes are a part of the NRC1 nuclear hormone receptor family. The PPARs and the retinoid X receptor (RXR) heterodimer connect to the peroxisome proliferator response elements (PPREs) in the promoters of target genes [20]. PPAR, a key regulator of mammalian metabolism (including fatty acid oxidation) is activated by naturally occurring or metabolized fatty acids, such as prostaglandins and leutrienes [21]. PPAR is part of an effective treatment for a variety of diseases, including type 2 diabetes, atherosclerosis and obesity [22]. Therefore, they are important. Now we also have a lot of synthetic agonists. Fibrin, such as clofibrate and fenofibrate, are anti-atherogenic drugs that activate PPARa. Thiazolidinediones, such as teglitazone (TZ) and rosiglitazone (RZ), which are used to treat Type 2 diabetes, activate PPARg. Other specific PPARd agonists include GW0742 and L165041. Biagonists that activate PPARa and PPARg have been shown to be effective agents in the treatment of metabolic disease [19]. However, some studies have shown that double agonists promote the carcinogenesis of the bladder urinary epithelium [23]. Dual PPARa/g agonists may indirectly promote urinary epithelium cancerization through the crystal induced urothelial injury response [23], and urothelium may also be directly affected by signal effects [24]. The full effects on humans of the use of dual PPARa/g agonists remains unclear.

The concentration of PPAR agonists has an effect on the proliferation of NHU cells. Low concentrations of PPAR agonists have no significant inhibition on the growth of mice, while high concentrations of PPARd and PPARg agonists can cause the death of cells.

2.1.4 Normal human urothelium cell culture

To investigate the effects of different PPAR agonists on the proliferation and differentiation of human urinary epithelial cells, we used a human urinary epithelial (NHU) cell culture system [25]. Normal human urothelial tissue was obtained from patients without urothelial carcinoma. In this experiment, we removed normal urothelial tissue from patients who had never had urothelial carcinoma (all samples used in the study were obtained with the informed consent of the patient and approved by the relevant research ethics committee) [32]. First, the urothelium was isolated from the stroma. Collagenase and bovine pituitary extract, epidermal growth factor (EGF) and cholera toxin were added to serum free medium (KSFM) of keratinocytes in a low calcium environment [25]. In this experiment, there are three independent cell lines used as replicates. During the experiment, NHU cells showed the ability of rapid regeneration and proliferation. Cells are maintained in a limited number of cell lines by continuous reproduction. Both exogenous and autocrine mechanisms are involved in cell proliferation. Cell proliferation can be blocked by EGF receptor inhibitors, and EGF receptor behavior can be monitored by western blot analysis [27]. According to immunocytochemical analysis of CK (e.g. CK7+, CK8+, CK17+, CK18+, CK19+) and junctional protein (E-cadherin, claudin+, claudin 4+, claudin 7+) expression, the results showed that the phenotype of NHU cell culture was very similar to that of basal/intermediate urothelial cells in situ [32]. CK14 expression instead of CK13 expression showed squamous metaplasia. Cultured NHU cells cannot be differentiated during culture process unless UPK1b is added. The expression of markers related to in situ urothelial differentiation in NHU cells was negative including UPK1a, UPK2, UPK3a, CK13, CK20, claudin 3 and claudin 5 [26]. The image of NHU cell culture is shown as fig 3.1:

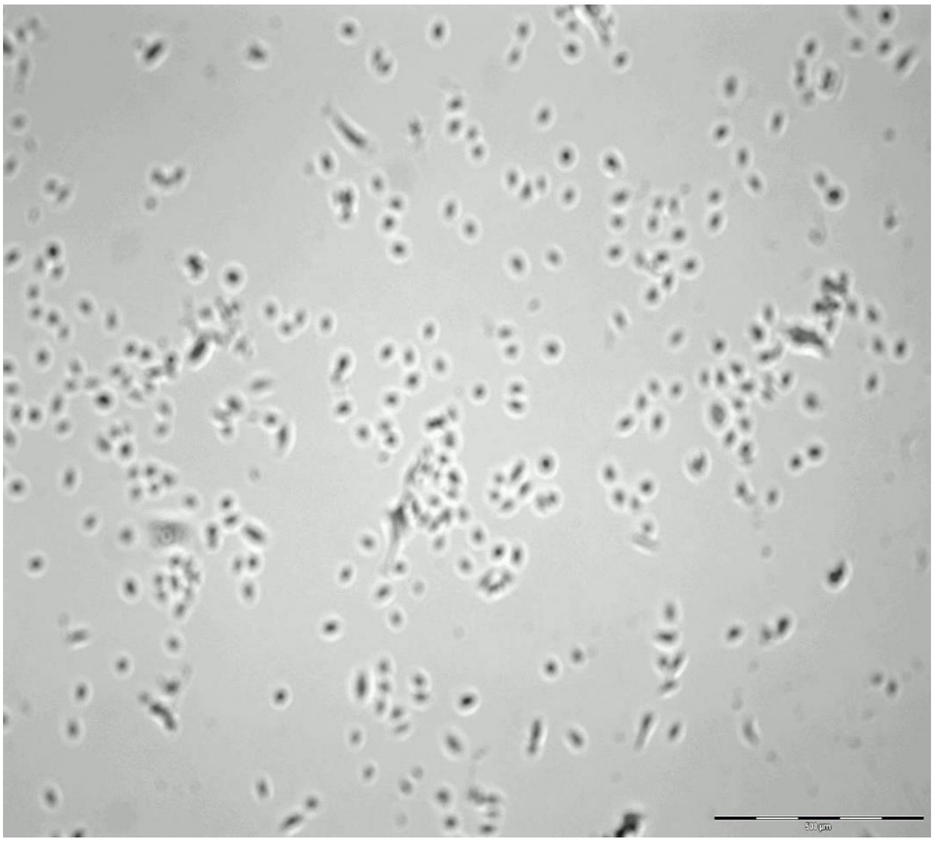


Fig. 3.1 Single frame of normal urothelium cell culture in time-lapse video

2.2 Cell tracking

In the biological field, segmenting and tracking moving cells in time-lapse video sequences is a challenging task. It is demanded by many applications in both scientific and industrial settings. Properly characterizing how cells change shape and move as they interact with their surrounding environment is key to understand the mechanobiology of cell migration and its multiple implications in both normal tissue development and many diseases.

There are two major factors that influence a cell tracker's performance. They are (i) the ability of recognizing cells and (ii) keeping the track of them. The typical input to a cell tracker system is a time-lapse video composed of sequential frames of cell culture photos taken at a regular time interval. Generally, the first step of cell tracking is to split the video into single frames and detect every cell within it. To process each frame, a filter is applied to reduce the effect of noise and improve the light and contrast due to the unbalance properties of the video background. Then image processing methods and tools are applied to determine in which areas cells exist, usually the central position of the cell. Video processing, in this way, is image processing repeated for each frame. There is a relationship in the position of cells between a series of continuing frames. The cell tracking algorithms attempt to locate the cells between frames. It is also the major challenge of cell tracking. The current cell tracking technique is only finding the relationships of cells between frames just by its position, which is an unstable method and cause a huge number of cell loss when cell number is big and video takes a long time. There are many different types of cell tracker on market now. Most of the cell tracker is facing the videos which have few cells, small region and high microscope accuracy aiming to gain more information out of single cells such as characterizing the shape of cells and the component inside the cell. There is a small number of cell tracker which is targeting the videos containing large number of cells and a wider area.

2.2.1 Ctracker

Ctracker is a cell tracker software system developed by Matthew Bedder of the Department of Computer Science at the University of York. This software was developed with the purpose of generating an automated cell tracking program based on the Open CV computer vision programming library [25]. The input video of the Ctracker is a series of time lapse frames each containing between several hundred to several thousand cells. However, one problematic property of most cell tracking systems is the loss or inability to continue tracking cells through the course of the video for various reasons, such as occlusion by other cells and cells exiting the field of focus.

Original video is the input to Ctracker. Ctracker processes the video frame by frame. The software first preprocesses each frame and then attempts to find the possible location of cells, and track these to the next frame. Preprocessing first applies a gaussian blur filter to reduce noise in each frame followed by a thresholding processing with pre-set value to generate a binary image. A distance calculation is then used for additional processing of binary image. The core of the cell is assigned with a large value. The edge of cell is assigned with a small value and the background is assigned with a value of zero. In this way, you can get a cell that estimates the location of the center. The local maximum is then selected from the maximum to minimum scores, and the small areas near the maximum of each selection are filtered to reduce repeated selection for a cell. The selected maximum (x,y) coordinates are then used to estimate the cell position within the frame. To do this, a gaussian filter is first used to multiply the distance from the position of the cell in the previous frame to the position of the cell near the previous frame. The maximum number of pixels in that region can then be applied to assess the current cell location. This method is proved to be effective by Zhen Zhang in his previous work [26], this cell tracker could accurately track down over 80% of the cells under current frame with proper setting in this project. The application of gaussian filters indicates that the preferred match is adjacent to the original cell location. Although the cell tracking program is highly efficient, it cannot continuously identify and track the cell location for the entire video period. This causes a lot of cell loss. The causes of cell loss are varied. In addition to natural factors such as cell damage and death, the software searches for matches near the cell points of the previous frame, which will also affect the contrast and brightness changes of video. This means that if cell recognition is applied only to the initial video frame, many cells will not be tracked in the latter half of video. To solve these problems, a technique was used to eliminate each cell location of replication from the tracking process and to periodically conduct a new round of cell identification to discover new candidate locations. The effectiveness of this technique was discovered by locating enough cells in video to describe the number of cells. The fig 3.2 presents the process of the Ctracker to detect and track cells.

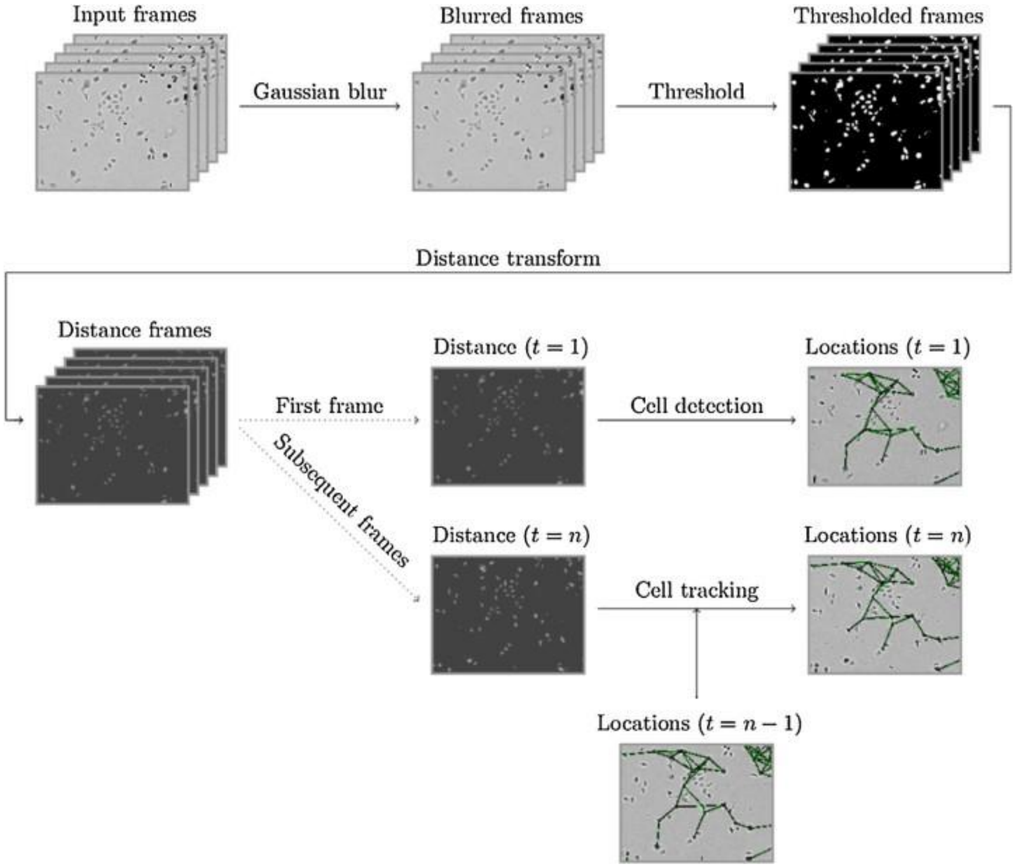
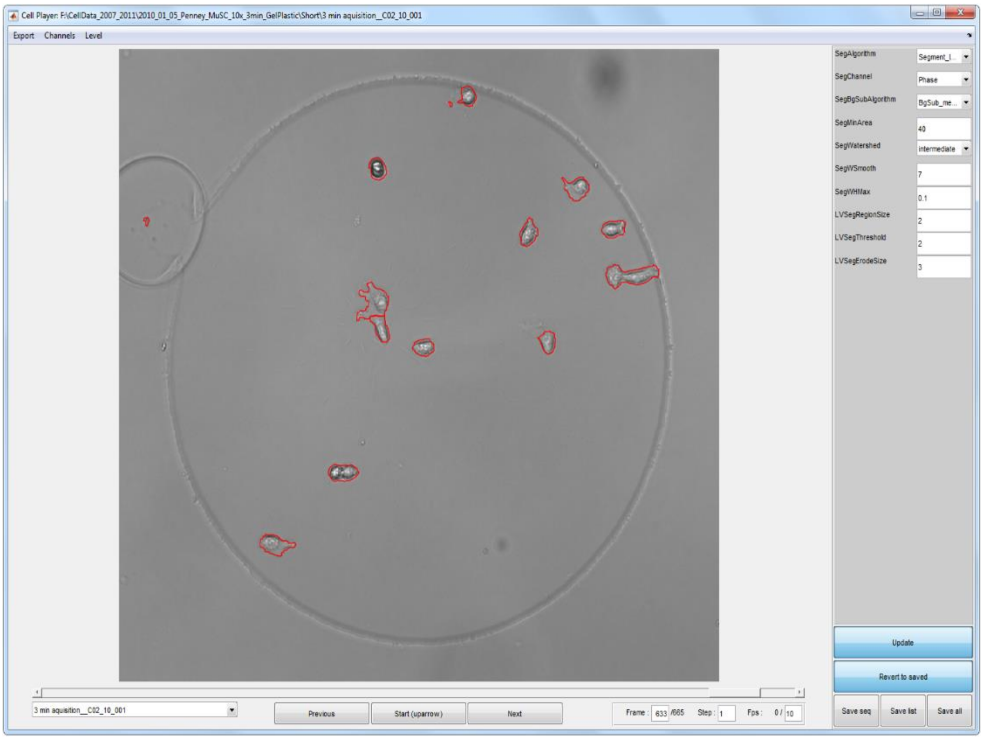


Fig. 3.2 The process used for detecting cell locations within a video, and tracking of detected cells between video frames.

2.2.2 Baxter-Algorithms tracker

The Baxter Algorithm (BA) is a software package for the tracking and analysis of cells in microscope images. The software can handle images produced using either transmission microscopy (e.g. bright field, phase contrast, and differential interference contrast (DIC)) or fluorescence microscopy (e.g. wide field, confocal, and light sheet).

The analysis of transmission microscopy images is limited to 2D, but 3D stacks of fluorescent images can be processed. In addition to cell tracking, the BA can perform automated analysis of fluorescent histological sections of muscle tissue, and automated analysis of myoblast fusion. The software is written in MATLAB, but it also contains some algorithms written in C++, which are compiled into mex-files. The software as a whole is presented by K. E. G. Magnusson in 2016 [27] and an example of how the software can be used to analyze muscle stem cell (MuSC) behavior is found by P. M. Gilbert, K. L. Havenstrite, 2010 [28]. The data association algorithm used to generate cell tracks is described. The software has shown outstanding performance compared to other software in the ISBI Cell Tracking Challenges according to K. E. G. Magnusson's previous research [27]. As figure 3.3 shows, the Baxter-Algorithm tracker shows great accuracy in cell tracking and cell segmentation with low cell density.



Cell Player: F:\CellData_2007_2011\2010_01_05_Penney_MuSC_10x_3min_GelPlastic\Short\3 min aquisition__C02_10_001

SegAlgorithm: Segment_l...
SegChannel: Phase
SegBgSubAlgorithm: BgSub_me...
SegMinArea: 40
SegWatershed: intermediate
SegWSmooth: 7
SegWHMax: 0.1
LVSegRegionSize: 2
LVSegThreshold: 2
LVSegErodeSize: 3

3 min aquisition__C02_10_001

Frame : 633 /665 Step : 1 Fps : 0 / 10

Fig. 3.3 Example of the Baxter-Algorithms software used in cell tracking

2.3 Machine learning

2.3.1 Introduction

Machine learning (ML) is the scientific study of algorithms and statistical models that computer systems use in order to perform a specific task effectively without using explicit instructions, relying on patterns and inference instead.[33] Machine learning is a technology in which computers have a similar learning ability as humans. It is a kind of data mining technology to find useful knowledge from a large amount of data. Machine learning of computers can be divided into two types, supervised learning and unsupervised learning, according to the types of data processed.[38]

> **Supervised learning:** Supervised learning is a method in machine learning, which can be learned from labelled training materials or establish a learning model and infer new instances based on this model. Training data consists of input objects (usually vectors) and expected outputs. The output of a function can be a continuous value (regression analysis) or a prediction of a classification label (classification). Supervised learning is based on the experience and skills acquired in the learning process and can correctly answer the questions that have not been learned, so that the computer can obtain the generalization ability, which is the ultimate goal of supervised learning. Supervised learning is widely used in handwritten word recognition, image processing, voice recognition, spam classification and genetic diagnosis.[38]
>
> **Unsupervised learning:** Solving various problems in pattern recognition according to training samples whose classes are unknown (not labelled) is called unsupervised learning. Unsupervised learning is not limited to solving problems with clear answers, so its learning objectives need not be very clear.

Unsupervised learning is of great value in video analysis, voice signal analysis, social network analysis and other aspects. At the same time, data visualization and pre-processing tools as supervised learning methods are also widely used.[38]

2.3.2 Genetic programming

In artificial intelligence, an evolutionary algorithm (EA) is a subset of evolutionary computation, a generic population-based metaheuristic optimization algorithm.[35] An EA uses mechanisms inspired by biological evolution, such as reproduction, mutation, recombination, and selection.[36] Candidate solutions to the optimization problem play the role of individuals in a population, and the fitness function determines the quality of the solutions (see also loss function). Evolution of the population then takes place after the repeated application of the above operators.

Inspired by the biological simulation technology, professor Holland and his students from the University of Michigan in the United States created an adaptive probabilistic optimization technology, a “genetic algorithm”　based on the genetic and evolutionary mechanism of biological adaptation and optimization of complex systems.[10] The term "genetic algorithm" was first proposed in 1967 by Bagley, a student of Holland, in his doctoral thesis.[35]

Genetic programming (GP) is an evolutionary algorithm (EA) that is a subset of machine learning.[35] EAs are used to find solutions to problems that humans do not know how to solve directly. EAs tend to generate better solutions without human bias. Genetic Programming (GP) is a program evolution technique that does not fit (usually random) programs to begin with, adapting operations similar to natural genetic processes to specific tasks by applying them to a population of programs.[36] It is essentially a heuristic search technique that has some similarities to "hill climbing," which searches the space for the best or at least the right programs across all programs.[11] Inspired by Darwinian evolution theory, GP software system implements an algorithm that uses random variation, crossover, fitness function and multi-generation evolution to solve user-defined tasks. GP can be used to discover functional relationships between features in the data and to group the data. the concept of genetic programming was put forward long ago, but limited computational power.[12] With the development of computer hardware, we have been able to construct a simple program according to the ideas of the genetic programming, but still can't fully exert the ability of genetic programming. Computational power is really the only obstacle to genetic programming problem.

The basic workflow of genetic programming is as follows(as fig3.4 shows):

1) Initialization: At the beginning of the algorithm, a number of programs are randomly generated, to form a population. These programs can be randomly generated or initially designed by hand, and can be considered to be a set of good solutions to help evolution on its way.

2) Evaluation: These programs compete in a user-defined task. Use fitness function to test the execution effect of the program, and sort by performance results. This is the same as the evaluation function as other optimization algorithm. We all choose one or several of the better randomly generated algorithms. The difference is that the evaluation function of genetic programming may include running time, space complexity and other issues.

3) Evolution: Similar to genetic algorithms, the best-performing programs are copied and modified. There are two different ways of evolutions.

   Mutation: random modification.

Crossover: the removal of some part of an optimal program and the substitution of some part of another optimal program, resulting in the copying and modification of many new programs based on the original one but different from them.

4) Reproduction: At each stage of replication and modification, the algorithm evaluates the quality of the program with an appropriate function. Because the population size remains constant, many of the worst performing programs are removed from the population to make room for new programs, which are called the "next generation," and the whole process is repeated over and over again.

5) End conditions:

   (1) the optimal solution is found

   (2) find a solution that performs well enough

   (3) the solution has not been improved after several generations

   (4) the algebra of reproduction has reached the prescribed limit

Maybe for some problems, such as board games, there is no optimal solution at all, so only approximate solutions need to be found. Moreover, in the learning process, the performance of the program also depends on the performance of opponents, and in case of strong, it will be strong. Since the algorithm has self-learning ability, a nearly perfect algorithm tree is constructed step by step by strategy.[12] Since it is randomly generated, the generated solutions do not look like algorithms designed by human beings, for example, they are too complicated, but the final results are correct.

In genetic programming, the best programs are always preserved. And the algorithm is copied and modified on this basis. So there's reason to believe that each generation will do better than the last. Survival of the fittest, survival of the fittest, is also applicable to computer algorithm competition.[10] In particular, in genetic programming, along with the algorithm itself and all of its parameters, are evolutionary rules of survival of the fittest automatically designed.

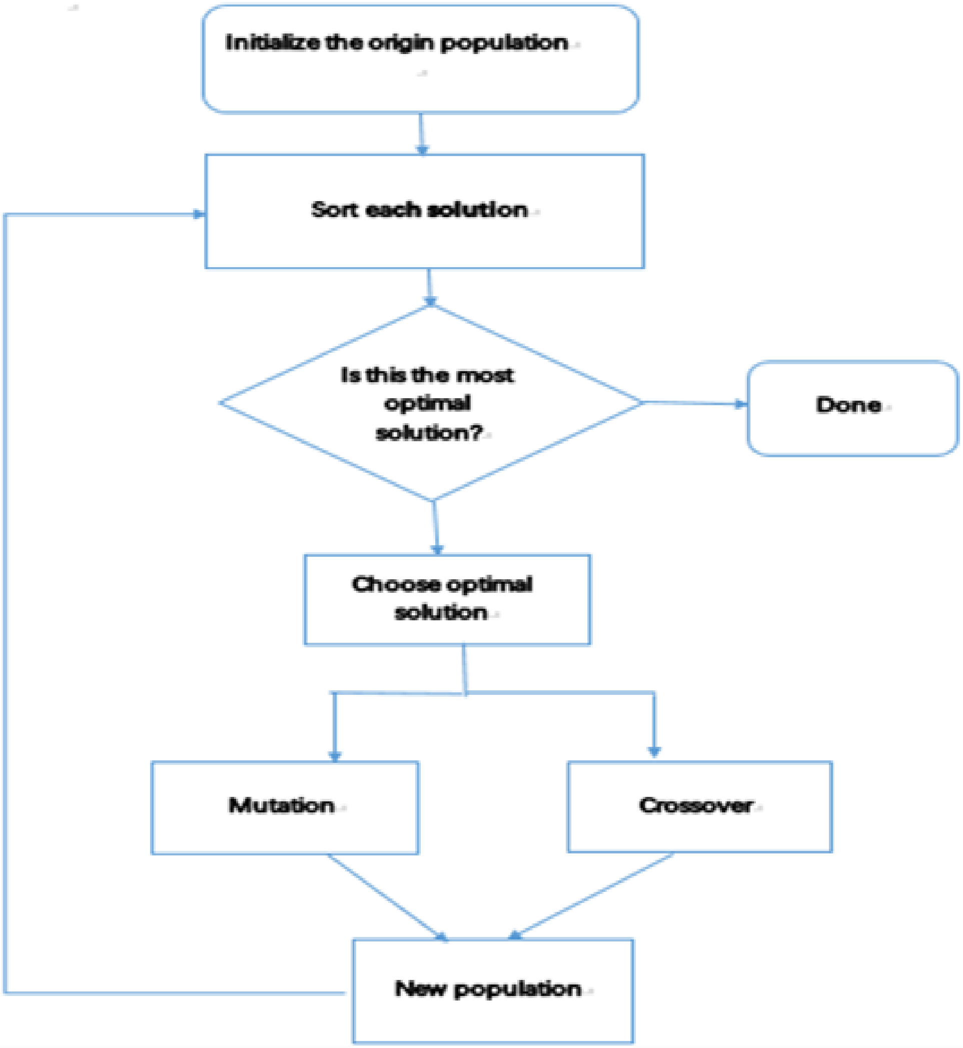


Fig. 3.4 Diagram of genetic programming process

2.3.3 CGP

Cartesian genetic programming is a form of genetic programming that uses a graphical representations to encode computer programs. It grew out of an evolving approach to digital circuits developed by Miller et al. In 1997 [12]. However, the term "cartesian genetic programming" first appeared in 1999 [13] and was proposed as a general form of genetic programming 2000 [14]. It is called "cartesian" because it uses a two-dimensional grid representing yhe arrangement of program nodes.

Cartesian Genetic Programming (CGP) is a form of Genetic Programming (GP) which is itself an Evolutionary Algorithm (EA). The basic unit of the cartesian genetic programming are chromosomes which represents functioning programs. The result of the CGP algorithm is a computer program that computes output based on input. These programs can be symbolic equations, Boolean logic circuits, neural networks, or almost any combination of connected computational elements. In the case of symbolic equations, each equation is composed of independent functions. CGP can be used to create programs that "link" these individual functions together to create symbolic equations.

A chromosome is an acyclic (and sometimes circular) graph of nodes. Each chromosome consists of three genes: the junction gene, the functional gene, and the output gene. The link gene stores the link method of the graph, and each node in the graph can be connected to any program input or other node. Functional genes describe the functions performed by each node in the graph (in symbolic equations, they can be addition, subtraction, multiplication, etc.). The set of possible functions that each node can perform is user-defined. Which node in the output gene description graph is used as the program output. Program output can be obtained from any program input or output of any node in the diagram.

There is a slight difference in basic workflow between genetic programming and CGP. It is mentioned as follows:

(1) Initialization: The initial population usually consists of randomly generated chromosomes. The user defines the number of inputs that the

program makes, the number of internal nodes that arity each node and the number of outputs that the program makes and creates chromosomes at random given conditions.

(2) Sorting: The active node is usually determined before calculating the fitness of CGP chromosome. This is because chromosomes are often run multiple times by fitness functions, and it is a waste of computational time to calculate the output of nodes that never contribute to phenotypic operations. The compute activity node is an O (n) operation that performs only once per chromosome. The exact fitness of chromosomes depends on how well they perform on a given task.

(3) Evolution: CGP typically only uses mutation to create the children form the selected parents. Common mutation methods used are probabilistic mutation and point mutation.

(4) Reproduction: CGP typically selects the fittest member(s) of the population to become the parent(s). The evolutionary strategy commonly used by CGP is to select one parent from each generation and use it to produce four children via mutation alone. The next generation then comprises of the selected parent and the four generated children. The population size is therefore five; four children plus one parent. This strategy is formally written as (1 + 4)-ES.

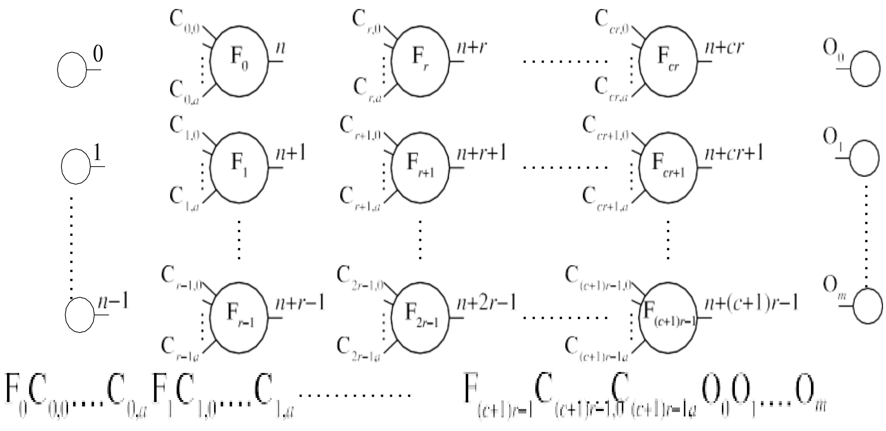


Fig. 3.5 General form of CGP. It is a grid of nodes whose functions are chosen from a set of primitive functions. The grid has n_c columns and n_r rows. The number of program inputs is n_i and the number of program outputs is n_o. Each node is assumed to take as many inputs as the maximum function arity a. Every data input and node output is labeled consecutively (starting at 0), which gives it a unique data address which specifies where the input data or node output value can be accessed (shown in the figure on the outputs of inputs and nodes).

2.3.4 Artificial Neural Networks

The Artificial Neural Network (ANN) is a widely used parallel and interconnected network composed of simple adaptive units whose organization can simulate the interaction of biological nervous system to real world objects.[38] In the classical ANN model, simple unit, namely m-p neuron model. We know that perceptron and Logistic regression are both linear classification models, and the difference between them lies in the choice of classification function. Typical m-p neuron models have the same input and output as Logistic regression, but Sigmoid exists here as an activation function. In other words, Sigmoid represents the output of one neuron, not the output of the whole ANN. A diagram graphically represents the MP neuron(fig.3.6):

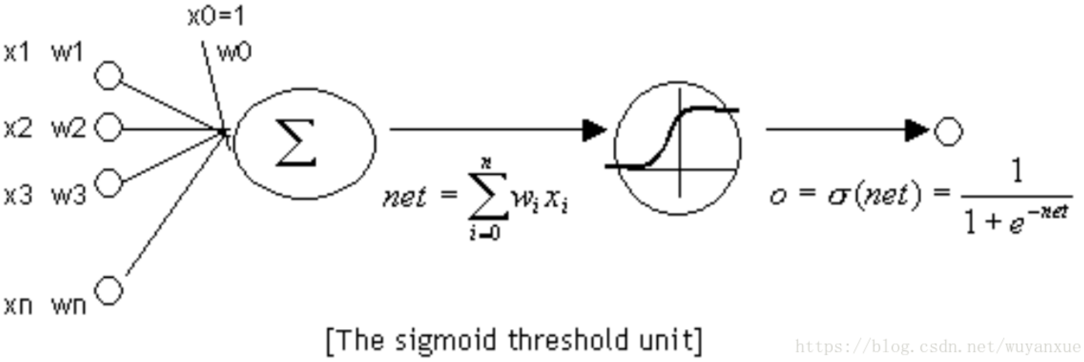

x0=1
w0
x1 w1
x2 w2
x3 w3
xn wn
Σ
$net = \sum_{i=0}^{n} w_i x_i$
$o = \sigma(net) = \frac{1}{1+e^{-net}}$
[The sigmoid threshold unit]
https://blog.csdn.net/wuyanxue

Fig. 3.6 Simple structure of the basic neuron unit

We know that biological neural networks are made up of a large number of biological neurons. Similarly, ANN is composed of multiple neuron models (fig 3.7) connected according to certain rules. [38]A fully connected neural network（at least three layers） has the following characteristics：

1. Neurons are arranged in layers. The first layer is called Input layer, the middle layer is called Hidden layer, and the last layer is called Output layer.
2. There are no connections between neurons in the same layer.
3. Each neuron at the N layer connects all the neurons at the N-1 layer (the meaning of Full connected), and the output of the neuron at the N-1 layer is the input of the neuron at the N layer.
4. Each neuron connection has a weight. Notice that this X=(x1,x2,x3) represents an input vector, and Y=(y1,y2) represents an output vector.
5. Hidden layers can be multiple.

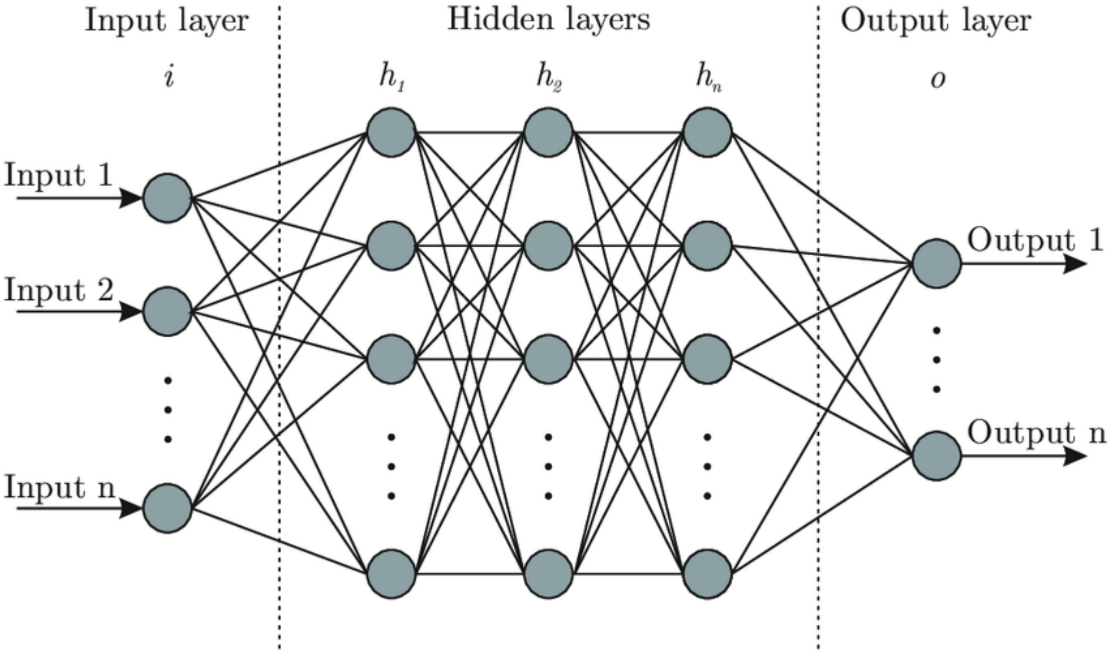


Fig. 3.7 Structure of a fully connected network

2.3.5 Support vector machines

Support vector machines (SVM) are a binary classification model. Its basic model is the linear classifier with the largest interval defined in the feature space, and the interval is the most important difference from perceptron.[33] The SVM also includes kernel techniques, which makes it essentially a nonlinear classifier. The learning strategy of SVM is interval maximization, which can be formalized as a problem to solve convex quadratic programming and is equivalent to the regularized hinge loss function minimization problem. The learning algorithm of SVM is the optimization algorithm of convex quadratic programming.[34]

The basic idea of SVM learning is to solve the separation hyperplane which can correctly divide training data set and have the largest geometric interval. As shown in the figure 3.8 below, [w·x+b=0] is the separation hyperplane.[34] For linearly separable data sets, there are an infinite number of such hyperplanes (perceptrons), but the separation hyperplane with the largest geometric interval is the only one.[33]

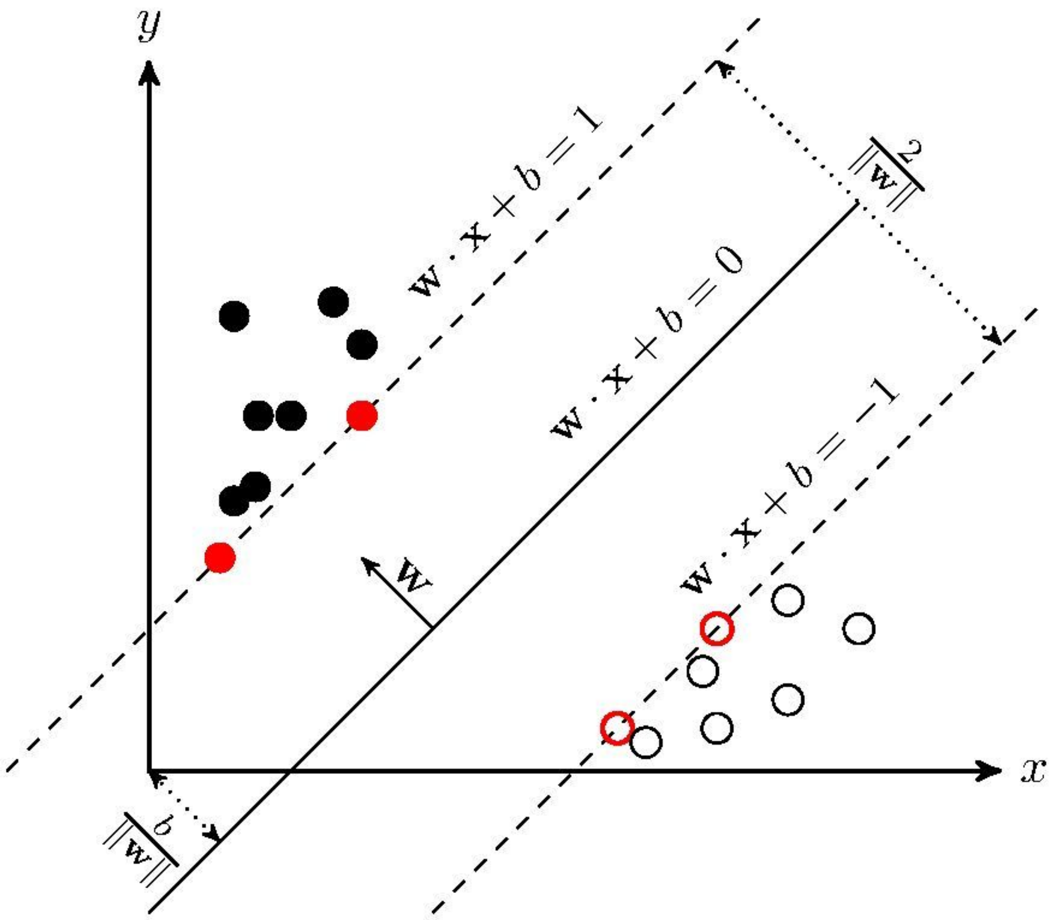

y
x
$\mathbf{w} \cdot \mathbf{x} + b = 1$
$\mathbf{w} \cdot \mathbf{x} + b = 0$
$\mathbf{w} \cdot \mathbf{x} + b = -1$
$\frac{2}{\|\mathbf{w}\|}$
$\frac{b}{\|\mathbf{w}\|}$
$\mathbf{w}$

Fig. 3.8 The principle of SVM algorithms

In this project, the algorithm we selected must be a white-box algorithm in order to better understand the behavior characteristics of cells. The ANN is a black box algorithm and needs a large dataset. It is not suitable for the current project in this case. SVM is one of the best classifiers. Because its own optimization goal is to minimize the structural risk, it has a good generalization ability. At the same time, through the concept of margin, SVM can get a structured description of data distribution, which reduces the requirement of data scale and data distribution. SVM is quite suitable to build a classifier in this project. CGP has always been a great solution to the white box problem. It provides a visual process and result for the problem and it has good applicability when it comes to a complicated question with small dataset. Due to the time limit, only SVM is applied in this project. CGP application for this project could be good direction for future research.

So far as can be determined, there are very few publications in the field of applying machine learning and other computational approaches to the analysis of cell cultures. The most relevant work in this field to date is reported by Zhen Zhang published in BioSystems (2016) [26]. It presents a novel method for classification and characterizing urothelium cell culture exposed to different concentrations of adenosine triphosphate (ATP) and a selective purinergic P2X antagonist (PPADS), acquired over a period of 24 hours [26]. It opens a new way of analyzing time-lapse video of cell culture. Based on his previous study, I develop a new efficient system which could provides more useful information from the dataset.

3. Methodology

The methodology of this study is an improved version of Zhang's previous study, in which Zhang presents a new way to classify and characterize cell culture by using CGP. He collects video of cell cultures and use cell tracking software to achieve the first-handed cell position data and thus presents the way of the feature extraction (including the individual cell movement and intercellular behaviors) before applying CGP to the extracted features [26]. Based on Zhang's analytical procedure, this study makes a series of technical improvements in the following aspects. Firstly, the two project is targeting different biological problems. Secondly, the feature extraction part is completely different from the previous work. There is a set of features selected to analyze the average behavior of cells under single frame. A time-lapse graph is then generated using the average value of different features each frame. It helps better analyzing the cell behavior and provides more information with the visual figure of these features. The GUI development module in MATLAB is applied to fulfill this task. The data visualization could help better understand the dataset for the biological research in cell behavior field. Thirdly, the feature extracted for single cell behavior is based on a different set of requirements in order to receive an advance in cell recognition accuracy. Lastly, only SVM is applied in this project to build the classifier using the features extracted for single cell behavior under a series of frame.

3.1 Cell culture

The cells were seeded in 12 plates separately as 4 sets of replicates among three cultures which are control group, TZ group and the T0070907(full named T00709070907) group. To investigate the effect of PPARg on cells behavior, the first control group is seeded in the culture of 0.1% DMSO. The second control group is cultured in 0.1% DMSO adding 1μM TZ as an PPARg activator. The third control group is cultured in 0.1% DMSO adding 1μM T0070907 as an PPARg inhibitor. Cell culture at the level of environmental automatics was observed by video microscope. Time-lapse video that last five days consists of different images with the size of 1367*1068 pixels. The images are captured every 10-minute.12 videos of cell culture are generated in the end, video 1,2,3,4 represent the control culture, video 5,6,7,8 represent the TZ culture, video 9,10,11,12 represent the T0070907 culture. The researcher Zhen Liu in biological department executes the cell cultures and provides the videos for further study. According to his results, I finished the further study and wrote this report all by myself. All the people who provides guidance and help in this project are all mentioned in the acknowledgement part.

3.2 Cell tracking

The videos of the cell culture is formed at a frame by frame basis. We then use cell tracking software to detect the cells in each frame and track the detected cells. The tracker we are using for this project is the ctracker developed by Matthew Bedder of the Department of Computer Science at the University of York which is able to tracker multiple cells at the same time [26]. The output of the ctracker is a series of x,y coordinates representing the positions of cells in every frame. The result of ctracker will undergo a further process to improve tracking accuracy (resizing the detection window). The result automatically generated by computer is compared to the one generated in biological methods. After multiple comparisons, it is found out that even though we can’t track all the cells in each frame, we still have the ability to keep track of about 80% cells in each frame. After discussion with the researchers in biological department, this accuracy is sufficient enough in this project. It is the general practice of the biological research.

The ctracker is developed based on the OpenCV computer visual library.
However, they do have some pitfalls. First, the gaussian blur is used to eliminate the

noise. Then a threshold is applied to the videoframe to generate a binary image. the threshold value is decided by the light and contrast of the background. However, in real time video, the light reflection could cause a difference on the microscope lens which lead to the unbalance in light and contrast of the image background value. After going through a threshold, the binary image will experience a distance calculation processor to assign different value to different area in the frame. The core of the cell is usually assigned with a large value, the edge of the cell is assigned with a small value and the background is assigned with zero. In this way, you can get the center that estimates the location of a cell. The local maximum is then selected from the maximum to minimum scores, and the small areas near the maximum of each selection are filtered to reduce repeated selection for a cell. The selected maximum (x,y) coordinates are then used to estimate the cell position within the frame. However, when the binary image turns into a digital matrix, some points could be recognized as cells by mistake due to the uneven light and contrast in the undergoing frame. In this case, the value of the corner would usually be bigger than the middle even though there isn't any cell on the corner. As the fig.4.1 shows, there are many wrong dots generated on the corner of the picture due to the light and contrast issue. In this case, the size of windows is reduced and the areas where most error occurs are ignored. After checking the conditions of brightness and contrast of the original pictures, three corners are cut off from the original picture. Each frame in the video is resized before applying the ctracker to increase the cell tracking accuracy. And the new result is shown in fig4.2.

While tracking cells works well, it can 't be consistently tracking cell locations during the whole video. In order to detect as many cells as possible, a large number of cell locations will be set as initializations. This also means that in the first few frames of tracking, the deviation can be very large. But over time, most positions converge rapidly to the same position. The cell tracking process of cell tracker will only track the cells recognized one frame before. It doesn't have the ability to search for new cells. We need to manually set a restart frequency to search for new cells. In this way, if we set the cell tracker at a low restart frequency. It may cause a miss out of important cell event, like cell proliferation, cell death or cell getting in/out of the frame. On the other hand, if we set cell tracker at a very high restart frequency. The percentage of number of points recognized as cells by mistake will reach a sharp growth which makes the data unreliable. In this case, a good restart case is very important for the research process (restart rate can be controlled by the setting parameters regen).

To solve these problems, a method of periodic detection is adopted to remove duplicate unit locations and find new candidate locations during the tracking process. The frequency of the periodic detection could be controlled by changing the parameter regen. For the first frame of a periodic detection, there will be many duplicate unit locations. And it takes further 3~5 frames to remove the duplicate points. In this case, the length of the periodic detection can't be too short. On the other hand, the new cells will not be detected during the periodic detection and there is also a number of cell loss each frame. The periodic detection can't be too long. The proper number of regen should be long enough to remove the duplicate unit locations and not too long to lose too many cells. In this project, the cell division cycle is 6 frames. In this case, the number of regen is set to 6 which means there will be a restart every 6 frame. The outputs of the cell tracker are compared to the ones generated using biological methods. The result shows that over 80% of the cells under current frame are tracked accurately. After discussion with the researchers in biological department, this result is considered very effective and adequately to describe the cell population.

There are some basic setting parameters for the cell tracker as follow:

-points: One of the most important parameters for the cell tracker. It sets the maximum amount of cells a cell tracker can keep track of. The time to process video increases as the number of points increases. In this case we are using 800 points as the setting.

-lineLength: it is a parameter that changes the output video. It will draw a line between the cells which are close to each other, but it has no influence on the output data file.

-sw: It sets the size of the search window for tracking cells between frames.

-inv: It decides whether the background color is dark or bright. It is set to be bright originally

-regen: It is the most important setting parameters for the cell tracker. It sets the restart frequency of the cell tracker. The time to process video increases a lot as the number of regen decreases. It regenerates the tracking points according to the frequency which is set by changing “regen”. The number can’t be too large (losing too much cell) or too small (make tracking points unreliable). For different project, there is a suitable number of “regen”. In this case, it is set to be 6.

-channel: Sets the color of tracking dots in the output video.

-h: Shows the help text.

The input of the cell tracker is a video recorded in avi format. The output can be a data file containing the x,y coordinates compiled in csv format and an output video file using colored dots to mark the cell in the videos and lines are drawn in between. Some trackers only work with a 30 frames per second video. Some videos need to be pre-processed before it enters the cell tracker if the frequency s not compatible.

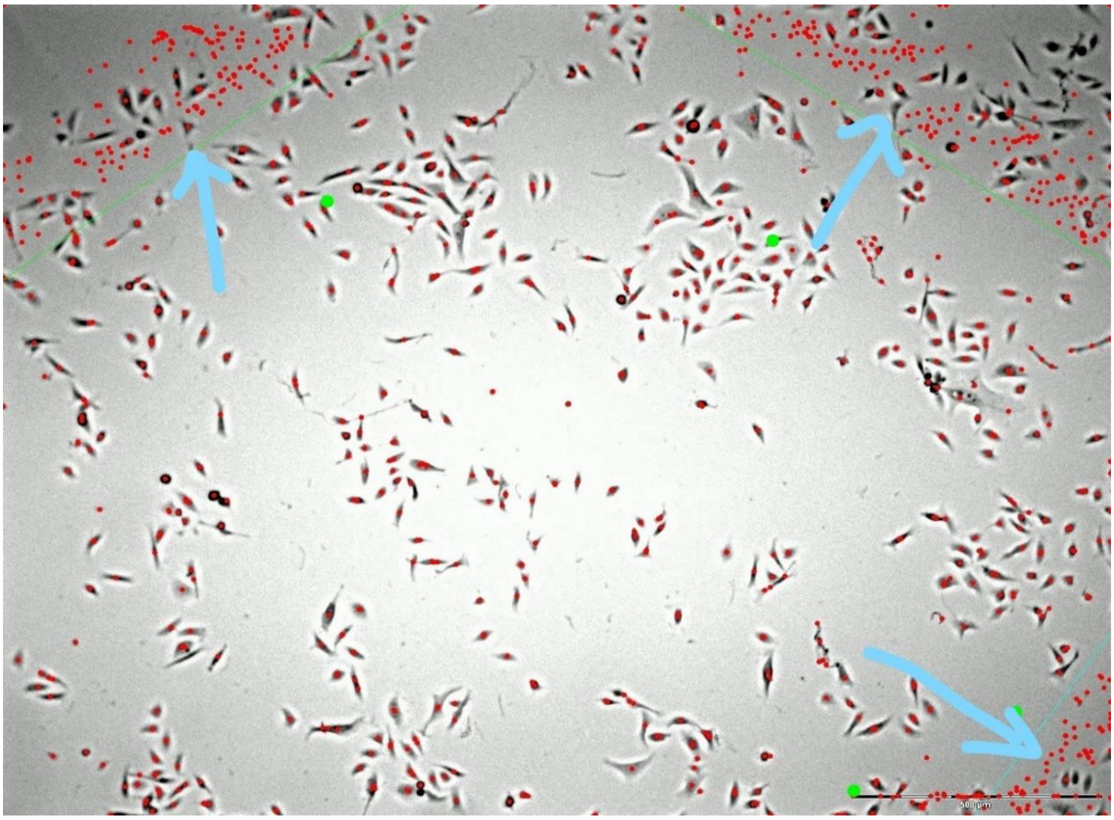


Fig. 4.1 All the raw data marked as the red dot in this frame and the area above two green line and blow the blue line is cut off in order to receive a better accuracy

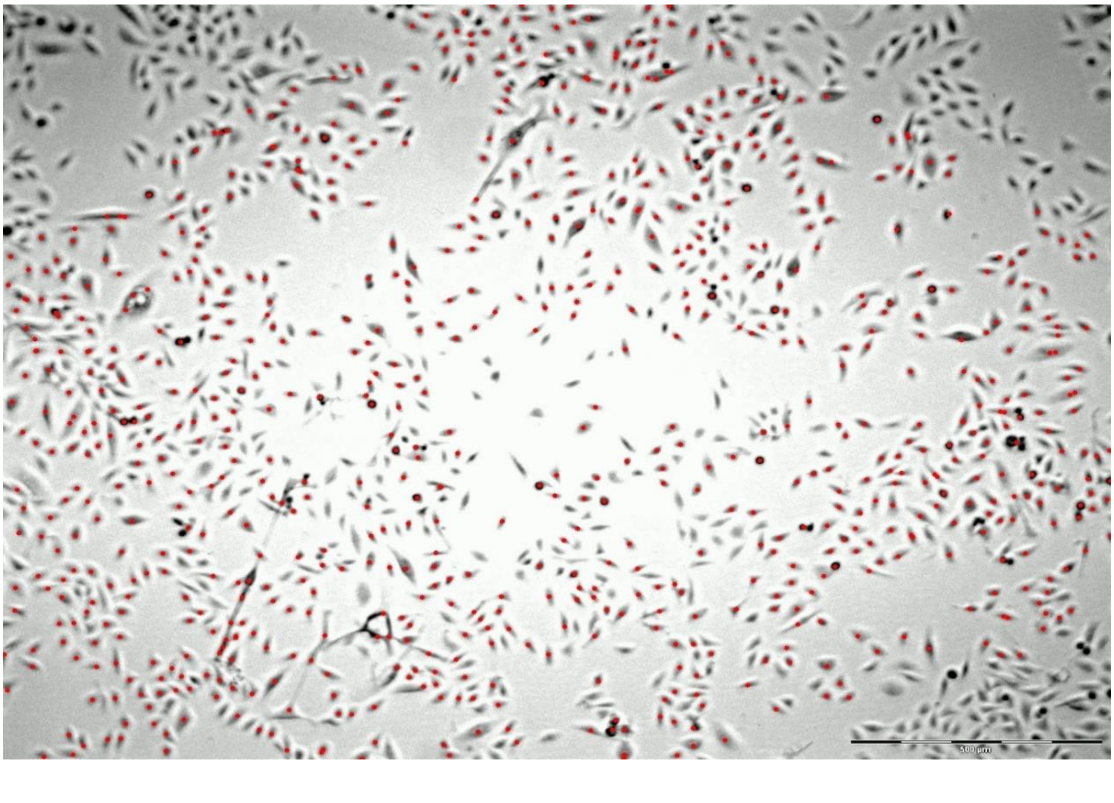


Fig. 4.2 All the tracking cells marked after cutting the selected part of the frame

3.3 Feature extraction

After the process of cell tracking, we receive a dataset of x,y coordinates of cell positions. The dataset only records the positions of cells under current frame. Usually, this data set includes a huge amount of data. In this case, the raw data will undergo a series of feature engineering before it can be used in machine learning algorithms. Feature engineering refers to the process of transforming original data into feature vectors. Feature engineering is the most important initial step in machine learning, which directly affects the effect of machine learning and usually takes a lot of time. Typical feature engineering includes data cleaning, feature extraction, feature selection and other processes. The feature engineering of this project will be carried out using Matlab.

The features are selected from the videos according to the attributes of the basic cell behaviors and the cell behaviors relate to clumping (according to biological need). The features are extracted in two different ways: based on frames and based on cells. The features extracted based on frames describe the average behaviors of cells in each frame. The output can generate different charts in terms of time which may provide a tremendous help to better understand the cell behavior in biological research. The other way of extracting features is based on cells, every sample represents a single cell behavior over a period of time. This dataset can be further processed and then used as the input to build a better classifier.

Details of the definition of different features are provided below in table 4.1.

features	description
Cell number	Total number of cells in current frame
Clump number	Total number of clumps in current frame
Speed	Average speed of cell in tracking time
Angular speed	Average angular speed of cells in tracking time
Clump duration	The length of time that cell stays in a clump
Clump size	The size of the clump which this cell exist in

Pre-binding speed	The average speed of cells before it enter in a clump(6 frames average)
Post-binding speed	The average speed of cells after it enter in a clump(6 frames average)
In-clump speed	The average speed when cell stays in a clump
Pre- binding angular speed	The average angular speed of cells before it enter in a clump(6 frames average)
Post-binding angular speed	The average angular speed of cells after it enter in a clump(6 frames average)
In-clump angular speed	The average angular speed when cell stays in a clump

Table. 4.1 brief description of the extracted feature

Speed

Speed is the unit of physics used to indicate how fast or slow an object moves.The speed is numerically the ratio of the displacement of the object in motion to the time it takes for that displacement to occur.

Calculation formula of speed v = Δ x/Δ t

In this case, the speed is calculated according to the cell position deviation between frames. This feature explains how fast and how slow the cell moves in some level.

Speed between two points(x1,y1), (x2,y2):

$$\text{speed} = \frac{\sqrt{(x_2 - x_1)^2 - (y_1 - y_2)^2}}{dt}$$

Angular speed

Angular speed is the rate of change of angular position of a rotation. In physics, angular speed refers to how fast an object rotates or revolves relative to another point. In this project, it is used to describe the direction change in single cells between frames. To calculate the angular speed in this case, the value of angle($\tan^{-1}\frac{x_2-x_1}{x_2-x_1}$) is applied. The angular speed is the change in angle value between frames.

Angle of this point: current point (x1,y1), next point(x2,y2)

$$\text{ang} = \tan^{-1}\frac{y_2-y_1}{x_2-x_1}$$

Angular speed between two points: ang1, ang2

$$\text{angular speed} = \frac{ang1 - ang2}{dt}$$

Cell number

Cell number is the basic feature of the cell culture. It describes the existing cell in the current frame. Due to the restrictions on the video and the cell tracker, the number of cells only has an at least 80 percent of accuracy when compared to the total number in the original frame. Luckily, this kind of bias between computational number and the actual cell number is acceptable for biological research.

Clumping features

Cells in suspension may attach to one another and form clumps for a variety of reasons. The most common cause of cell clumping is the presence of free DNA and cell debris in the culture medium, which occurs following cell lysis. The aggregation of 5 or more cells is defined as a cell clump and the maximum distance between two center of the cells to form a clump is the average distance of the urothelium cells which is 8 pixels in the time-lapse video in this project. The BFS (Breadth First Search) is used to find out the aggregation of cells. Clumping is a promising area in biological field under studying. In this case, there are many features about clumping in this project. The features related to cell clumping is shown in Fig. 4.3.

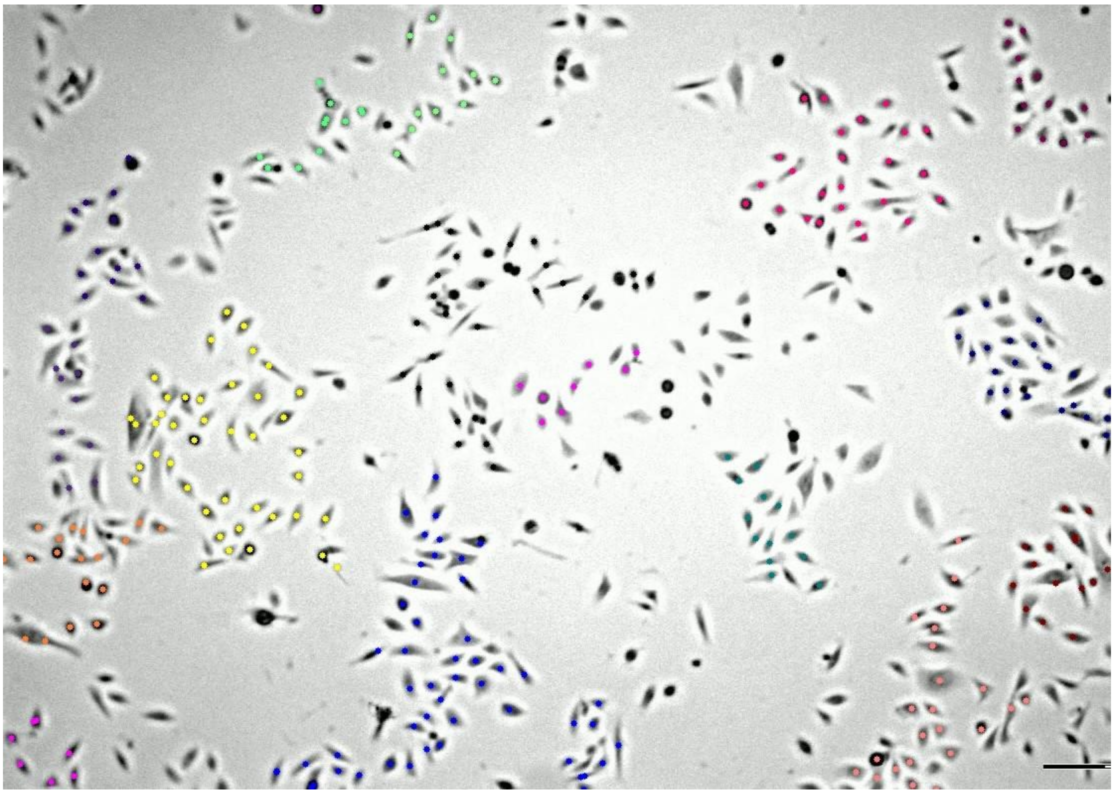


Fig. 4.3 single frame from the origin video, different color is applied with different clumps in current frame

Clump number

This feature describes the number of clumps in the current frame. This feature is influenced by the number of cells. Due to the loss of cells, there is a possibility that some clumps can be ignored when some cells in they are not recognized.

Clump size

Clump size describes how many cells are there in a clump. The clump size only has 70-80% accuracy due to the cell loss in cell tracker and videos capturing. However, the inaccuracy in clump size doesn’t have too much influence on the result.

Features in clump

It contains the speed in clump and the angular speed in clump. It describes the cell behavior inside a clump. Compared to the features such as clump number and clump size, it is a feature based on the single cell rather than a clump.

Pre-binding/post-binding feature

The pre-binding/post-binding feature mainly describe the angular speed and the speed before/after the cell enter the clump. According to the cell tracker setting, the regenerate period is 6 frames. The pre-binding cells require a cell to stay out of a clump for more than 3 frames before it enters a clump. The post-binding cells require a cell to stay out of a clump for more than 3 frames after it leaves a clump. The requirement is very strict in order to enhance the accuracy so there are very few cells that meet this requirement. It is not a suitable feature for the feature-frame graphic.

4.4 Application of machine learning algorithm

The main problem of this project is a multiple classification problem which intends to build a classifier that could sort out three different culture (control, TZ and T0070907). This multiple classification problem can be reduced to a combination of two simple binary classification tasks. In this case, SVM (supporting vector machine) is used to build the classifier in this project. The optimal hyperplane of feature space partition is the target of SVM, and the idea of maximizing margin is the core of SVM method. Nonlinear mapping is the theoretical basis of SVM method. SVM uses inner product kernel function instead of nonlinear mapping to high-dimensional space. SVM is a novel small sample learning method with solid theoretical basis. It basically does not involve probability measure and law of large numbers, so it is different from existing statistical methods. In essence, it avoids the traditional process from induction to deduction, achieves the efficient "transduction reasoning" from training samples to prediction samples, and greatly simplifies the usual classification and regression tasks. In SVM, a few support vectors determine the final result which can not only help us grasp the key samples and "eliminate" a large number of redundant samples, but also ensure that this method is simple and has good "robustness". The input of this cell tracker will be a dataset full of samples representing behavior of single cells. Each single sample has 12 features which includes speed, angular speed, number of cells in current frame, and the features concerns with clumping. There will be two different experiments taking places with the SVM. The first one is the classifier between control and TZ, the data set output will be 0 and 1(0 stand for control and 1 stand for TZ).The second one is the classifier between control and T0070907, the data set output will also be 0 and 1(0 stand for control and 1 stand for T0070907). The training set is formed of 2000 randomly selected points from single video. The test set is formed of 200 randomly selected points from single video. There will be 4000 samples used as training set and 400 used as the test set. The samples from the training set and the samples from the test set don’t overlap. Every experiment will be carried out 3 times and only the average accuracy will be used as the final result.

Because of time constraints, only SVM is applied in this project. While CGP is also a great option for further experiment. It is a white box machine learning algorithm and is very useful in figuring out the relationships between features.

4. Result and Analysis

In this experiment, 12 cell cultures are divided into 3 types, (1) Type one is the prototype culture. It's a control culture of normal urothelium cell. (2) Type two is the prototype culture with tz adding as the PPARg activator. (3) Type three is the prototype culture with T0070907 adding as the PPARg inhibitor. Each type of culture has 4 replicates which divide the whole 12 culture into 4 control groups. The result consists of two parts. The first part in which most of the results are presented will mainly focus on the behavior between different types of culture using different features. The second part will show the result of SVM implementation using the extracted features.

The purpose of plotting a series of time-lapse figures of different features is helpful to biological research by providing more useful and handy information using analytical methods and data visualization. Due to the deviation in cell planting manually and the unstable cell tracking software, the origin figures usually contain noise and serious fluctuation as the fig. 5.1 shows.

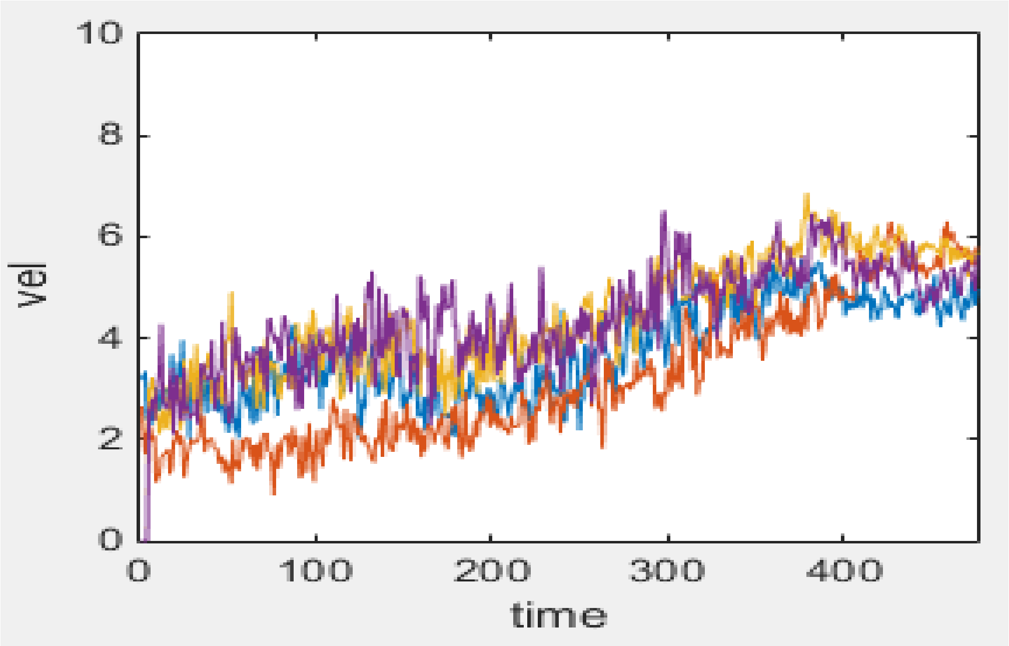


Fig. 5.1 Time-lapse graph comparing speed difference inside the 4 replicates(video 5,6,7,8) of tz culture

Some of these videos might reveals some obvious variance between different cultures. Due to the deviation in cell planting manually and the unstable cell tracking software, there is a lot noise in the original figures and the figures can’t be treated as solid proof in biological research as fig 5.1 shows. After discussion with the researchers from the biological department, it is agreed that the original figures still need some further process before it could be taken as reliable biological results. In this case, the raw figures will be pre-processed in order to eliminate these noise and fluctuation. The first step is to smooth these curves by averaging data several frames rather than providing data each frame. The fig. 5.2 shows the output of this process. The blue line is the origin figure of the average speed each frame. The red line is the smoothed figure after processing. As the fig. 5.2 shows, the line after processing looks better and is more valuable to biological study.

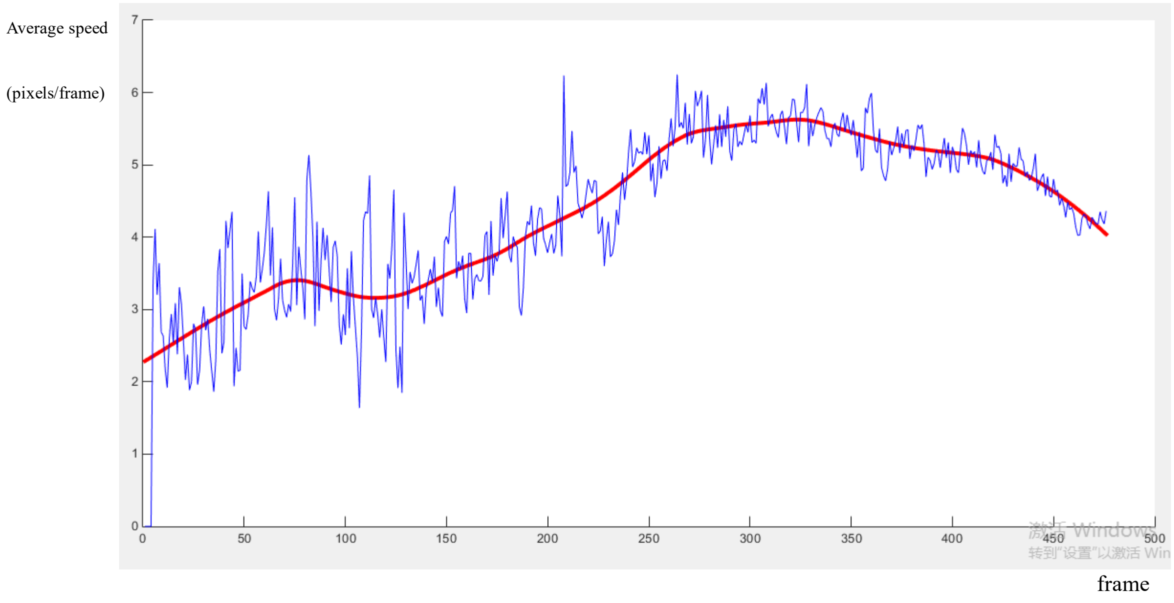


Fig. 5.2 The time-lapse figure of the average speed of a single video. The blue line is the origin data. The red line is the function after smoothing out the origin data.

Control group is a basic setting of the biological experiment. In biological experiments, very small changes can affect the final results. Every experiment would have the different result even if they are set in the same environment. In this case, the control group is very important to biological study. The biologic study currently focus on the mean value and the intervals among the replicates as the fig. 5.3 shows.

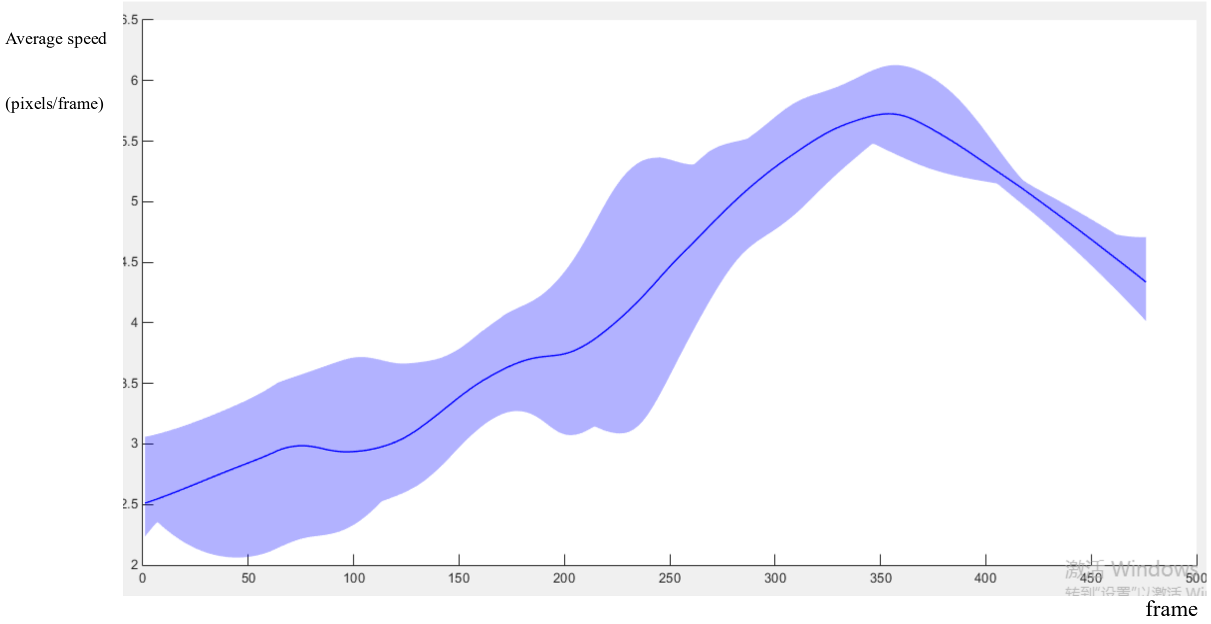


Fig. 5.3 the mean value of the speed with intervals in 4 replicates under control cultures

The blue line in figure 5.3 shows the average speed among 4 replicates and the area which is painted light blue indicates the range of the speed among 4 replicates. Furthermore, the figure of the mean value with the intervals of other 2 types of cell cultures are also plotted to do some comparison as the fig. 5.4 shows. The blue area stands for the control culture. The red area stands for the TZ culture. The yellow area stands for the T0070907 culture. The blue line curves the mean value of the control culture. The red line curves the mean value of the TZ culture. The yellow line curves the mean value of the T0070907 culture. After the procedures of preprocessing listed above, the information in the diagram becomes clearer and more helpful to the biological study of cell behavior.

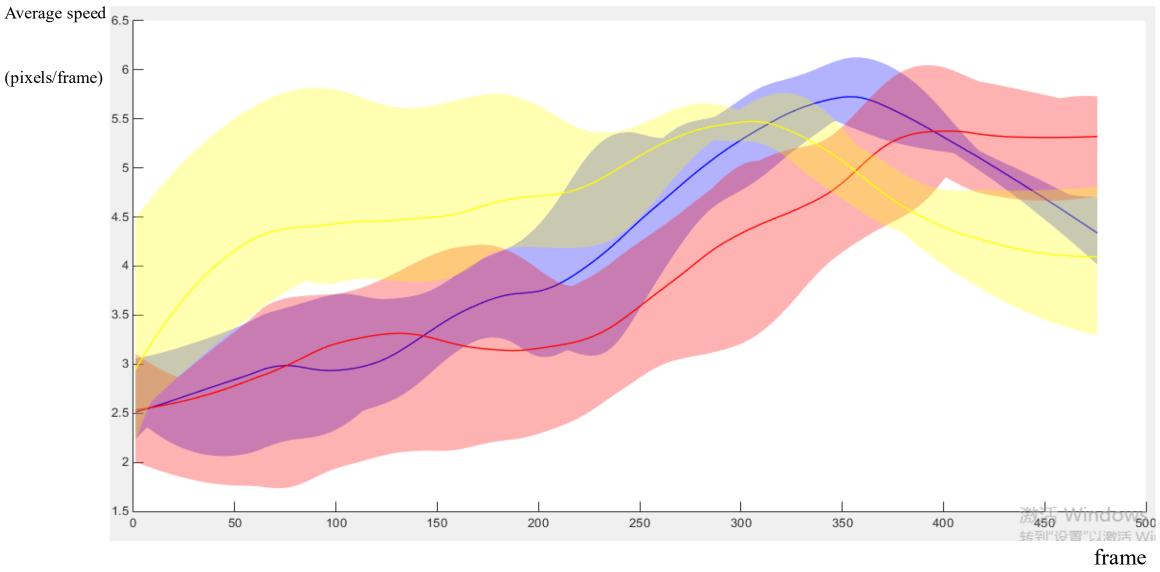


Fig. 5.4 the comparison of the mean value with intervals between different cultures (control, TZ, T0070907). The blue area stands for the control culture. the red area stands for the TZ culture. the yellow area stands for the T0070907 culture.

Here are the results of analyzing different features listed below:

4.1 Number of cell

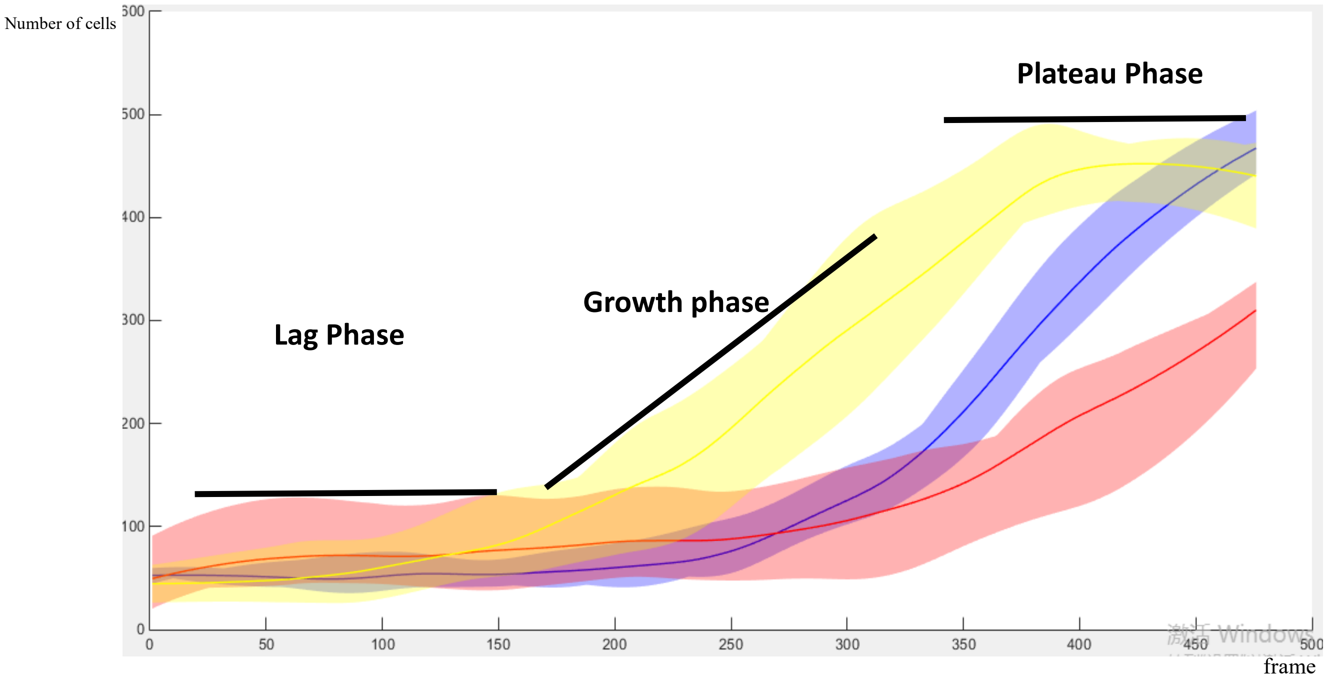


Fig. 5.5 A time-lapse graph of the number of cell graph of all 12 videos, comparing the difference in the number of cells among three types of culture. The blue line stands for mean value of the control culture, the blue area stands for the intervals. The red line stands for the TZ culture, the red area stands for the intervals. The yellow line stands for the T0070907 culture. the yellow area stands for the intervals

As is shown in the figure above, Fig. 5.5 compares the difference of the number of cells among different types of cell cultures of all 12 videos. In this figure, the abscissa is the number of frame and the ordinate is the total number of cells in this frame. As the figure shows, the blue line draws the mean value of the number of cells in control culture, the red line draws the mean value of the number of cells in TZ culture and the yellow line draws the mean value of the number of cells in T0070907 culture. The number of cells tends to increase in all cell cultures but at a different speed due to the different drugs addition to the cell culture. In general, the TZ culture has the lowest number of cells while the control culture and the T0070907 has more. The result is quite similar to the recent biological research which indicates that the population growth of cell in the TZ group is slower than the control and T00709070907 groups. However, the biological research has its own limitation. The biological results only find out the difference in certain time point rather than plotting a time-lapse graph and the only parameter it observes is the number of cell under current biologic analytical condition. In figure 5.5, the blue area stands for the range of the number of cells in control culture, the red area stands for the range of the number of cells in TZ culture and the yellow area stands for the range of the number of cells in T0070907 culture.

As the painted areas shows, the variance inside a culture is not too big. There is not too much difference among different control groups. For all 12 cultures, the population is always growing throughout the whole video. It also points out that the population growth of cell in T0070907 is the fastest at the beginning but the growing speed start to fall at the end of the video. It is suggested that T0070907 helps the cell population to grow at the beginning, and with the number of cells grows, the culture area become crowded and due to this reason, the growing speed of cell population starts to fall. However, the control culture hasn’t met the critical point so the growing speed of its population doesn’t slow down.

In conclusion, the growing in the number of cell experiences three phases. The lag phase, growth phase and the plateau phase. As the fig. 5.5 shows, the three phases are marked on the figure. The cell number is similar between groups during the lag phase at the start of the culture period, when the cells are adapting to the environment before they start to grow. The cells in the +T0070907 recover fastest and enter growth phase first, followed by the control cells. The cells grow until they become contact-inhibited because of cell density. Interestingly, the T0070907 cells plateau at a lower terminal density than control cells. The final cell number in the TZ group is significantly lower than the cell

numbers of other two groups and therefore it doesn’t reach the plateau phase.

4.2 Average speed

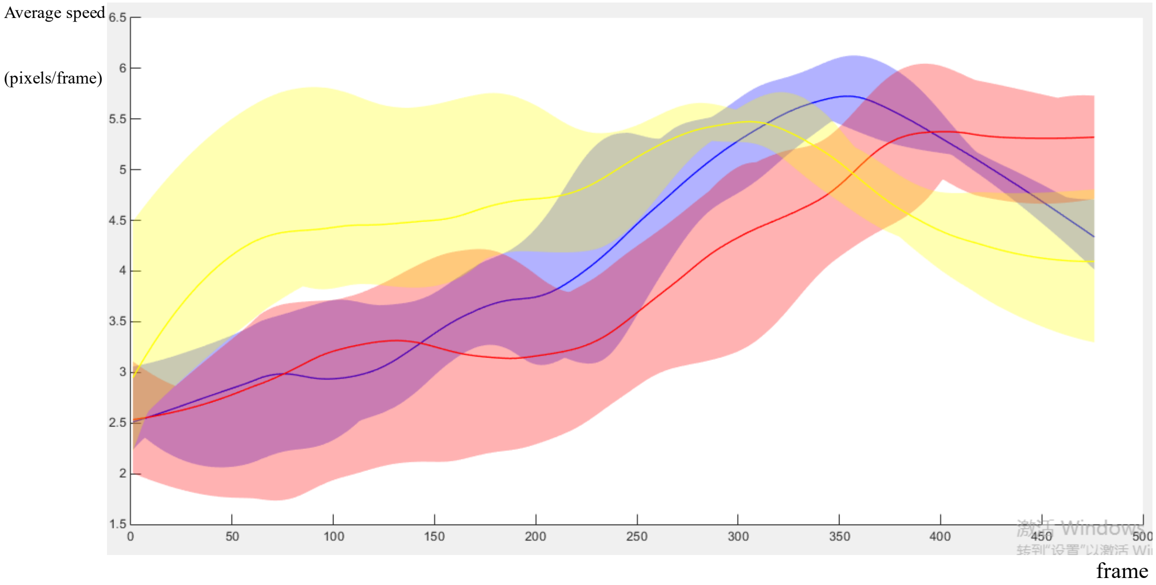


Fig. 5.6 A time-lapse graph of the average speed of cell graph of all 12 videos, comparing the difference in the number of cells among three types of culture. The blue line stands for mean value of the control culture, the blue area stands for the intervals. The red line stands for the mean value of TZ culture, the red area stands for the intervals. The yellow line stands for the mean value of T0070907 culture. the yellow area stands for the intervals

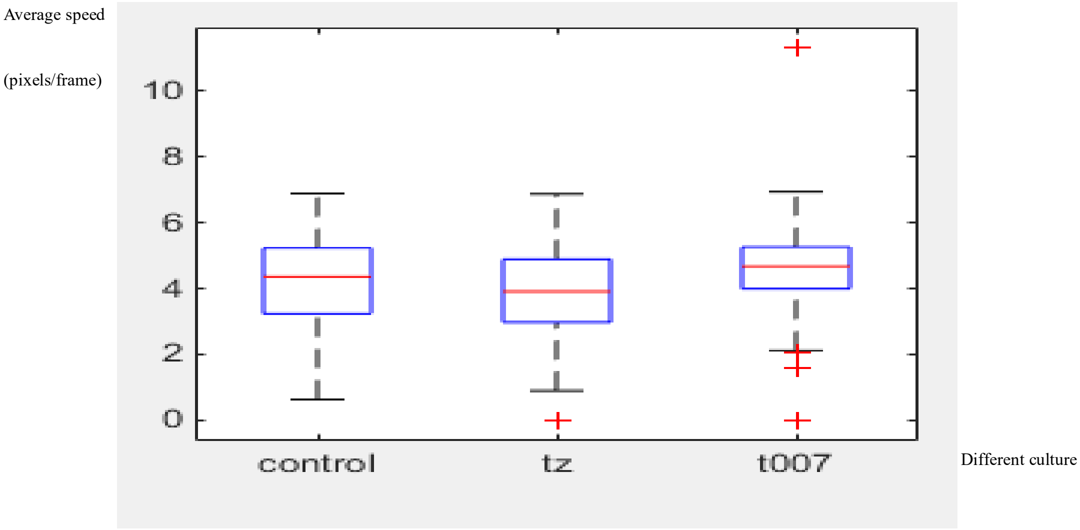


Fig. 5.7 Boxplot of average speed each frame among all 12 videos comparing three different types of culture

Except for the population of cells, other features don’t have a very obvious difference.

The cell speed is extracted using the position change of cell between frames. Average speed is the average number of speed among all the trackable cells under current frame. Basically, the value of the average speed changes in a small range but it still has some difference between different types of cultures. Fig. 5.5 is the time-lapse graph of the average speed comparing different cultures which is exposed to different drugs. In this figure, the abscissa is the number of frame and the ordinate is the total number of cells in this frame. As the figure shows, the blue line draws the mean value of the average speed in control culture, the red line draws the mean value of the average speed in TZ culture and the yellow line draws the mean value of the average speed in T0070907 culture. It is clear that the average speed of the cells in T0070907 culture has a higher speed value at the beginning. The T0070907 culture grows at the beginning and then falls after it reaches a certain value. The same thing happens for the control cell culture except for its peak comes a little bit later. It is obvious that the TZ culture haven’t reach its peak and the average speed is still growing. After careful study, it is found that the change of average speed follows the same pattern. The average speed always grows at first and then reach a peak and falls finally. This pattern works for the control culture and the T0070907 culture. Due to the duration of the video, we can’t figure out the further development of the TZ culture. It can be reasonable suggested that the average speed will keep rising before reaching a critical point and then fall. The addition of TZ to the cell culture may impose restrictions on the cell activities. To verify this assumption, the box plot fig. 5.7 is generated. As we can see on the fig. 5.7, the average speed of T0070907 culture and control culture is higher than that of the TZ culture.

In fig. 5.6, the blue area stands for the range of the average speed of cells in control culture, the red area stands for the range of the average speed of cells in TZ culture and the yellow area stands for the range of the average speed of cells in T0070907 culture. According to the figure, the variance in the same type of culture is quite large. However, this variance does not influence the pattern in which the curve of the average speed goes up in the beginning and falls down until it reach certain value. Also the yellow area is always on the top in the early stage of the videos and the red area is always on the bottom.

4.3 Average size of the clump

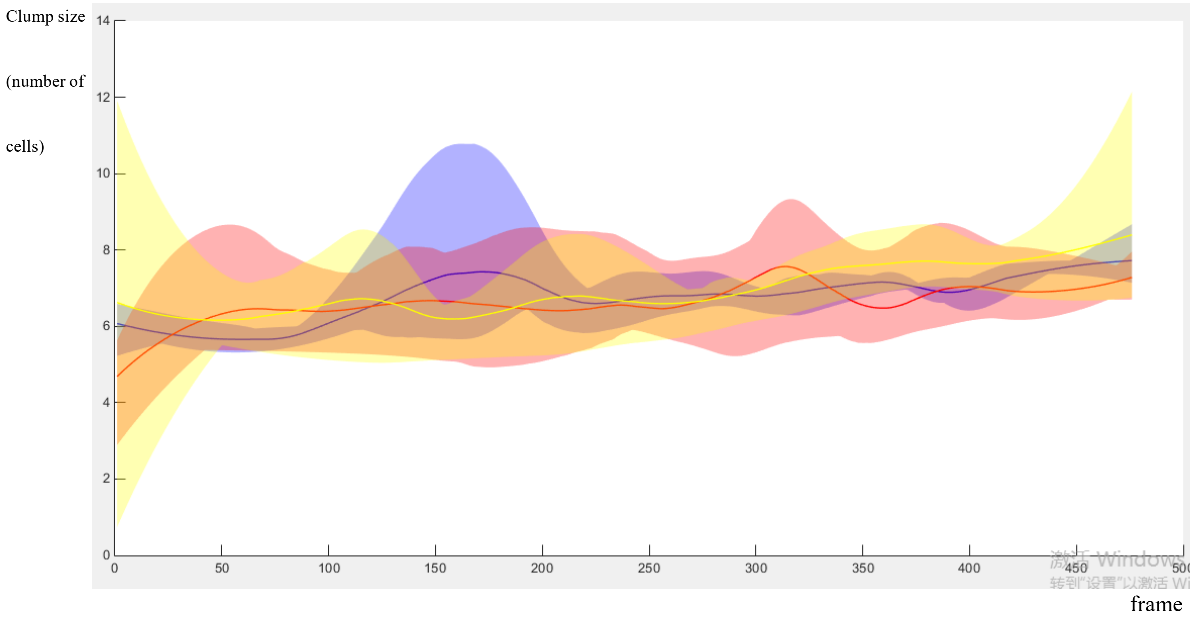


Fig. 5.8 A time-lapse graph of the average size of cell clump of all 12 videos, comparing the difference in the number of cells among three types of culture. The blue line stands for mean clump size of the control culture, the blue area stands for the intervals. The red line stands for the average clump size of TZ culture, the red area stands for the intervals. The yellow line stands for the average clump size of T0070907 culture. the yellow area stands for the intervals

The feature clump size represents the average size of the clump. It fetches the size of every clump in the current frame and calculates the average value of clump size each frame. The figure 5.8 is the time-lapse graph of the average clump size comparing different cultures which is exposed to different drugs among all 12 videos. As the figure shows, the blue line draws the average value in control culture, the red line draws the average value in TZ culture and the yellow line draws the average value in T0070907 culture. There is not much difference among the average clump size under different culture. In figure 5.8, the blue area stands for the range of the average clump size of cells in control culture, the red area stands for the range of the average clump size of cells in TZ culture and the yellow area stands for the range of the average clump size of cells in T0070907 culture. Due to the unavoidable deviation during the feature extraction, the value of average clump size is a quite unstable feature. As the figure shows, it has a big variance in range. The unavoidable cell loss when we use the cell tracking program could lead to a strong fluctuation in the number of clumps. The fluctuation eventually causes the instability in averaging the clump size each frame. In general, the average clump size feature is an unstable feature. It can provide very limited help to the biological research.

4.4 Speed in clumps

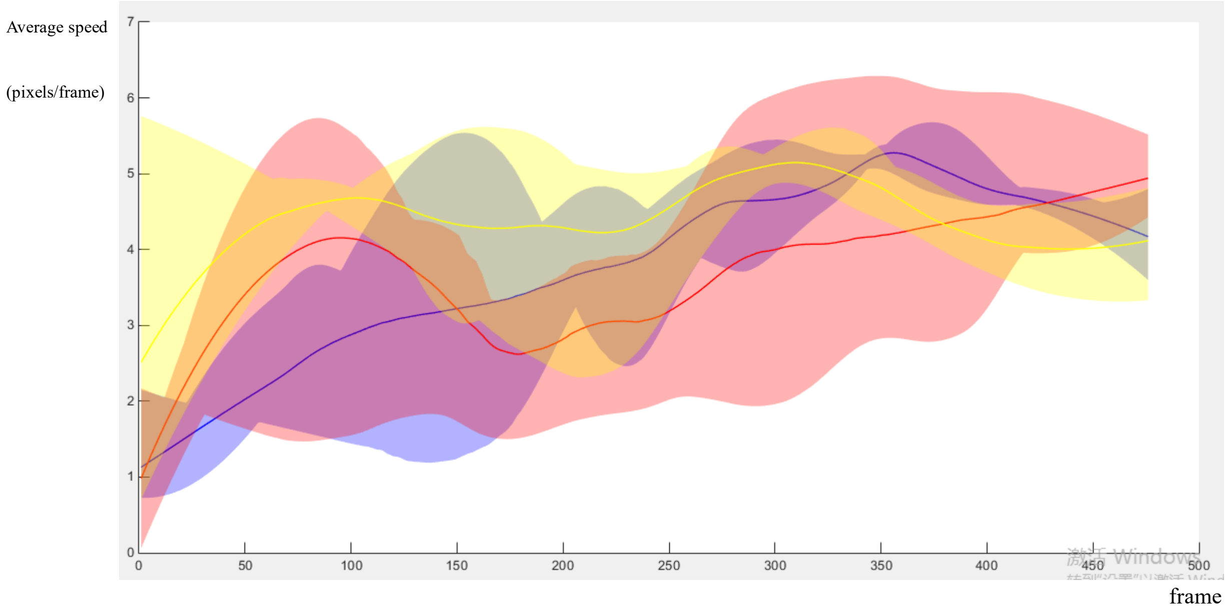


Fig. 5.9 A time-lapse graph of the average speed of cells in clump of all 12 videos, comparing the difference in the number of cells among three types of culture. The blue line stands for mean speed in clump of the control culture, the blue area stands for the intervals. The red line stands for the average speed in clump of TZ culture, the red area stands for the intervals. The yellow line stands for the average speed in clump of T0070907 culture. the yellow area stands for the intervals

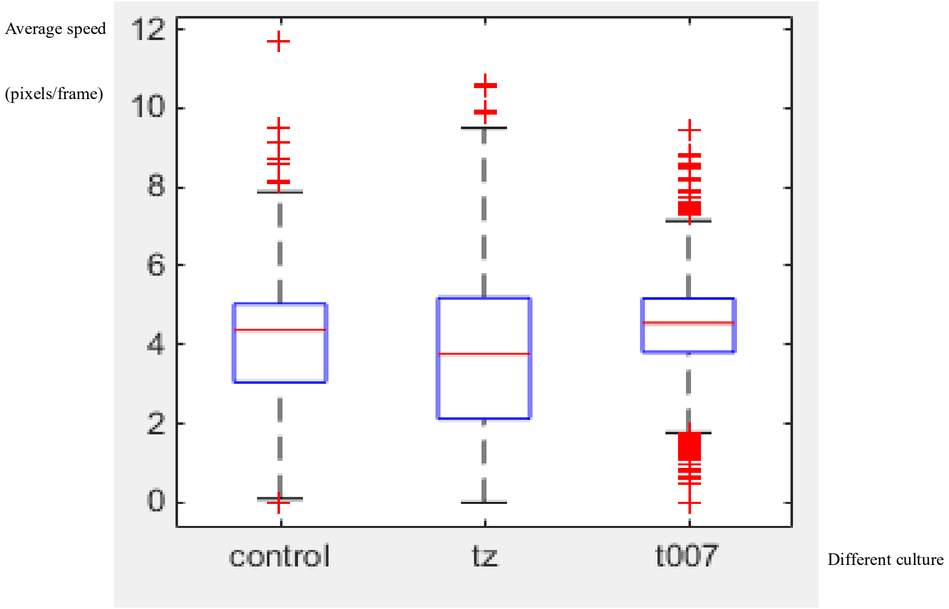


Fig. 5.10 Boxplot of average speed in clump each frame among all 12 videos comparing three different types of culture

The speed in clumps calculates the average speed of the cells which are currently in clumps. The fig. 5.9 is the comparison of the speed in clump under three different cell cultures among all 12 videos. As the figure shows, the blue line draws the average speed in clumps in control culture, the red line draws the average speed in clumps in TZ culture and the yellow line draws the average speed in clumps in T0070907 culture. The blue area stands for the range of the average speed of cells in clumps in control culture, the red area stands for the range of the average speed of cells in clumps in TZ culture and the yellow area stands for the range of the average speed of cells in clumps in T0070907 culture. This result seems to be quite unstable for this feature (the speed in clump). There are lots of variants in average speed of cells in clumps among the replicates in each group. The figure of average speed of cells in clumps seems to have a high fluctuation of the amplitude at the early stage of the cell culture and finally converge to a value in the end. We also look into the boxplot of all the speed of cells in clumps over the whole video period to see if there is any difference among them. As the fig. 5.10 shows, it is a surprise that when we use the average value of the average speed of cells in clumps each frame over all 12 videos to build a boxplot. There is a difference among three types of culture. The T0070907 culture and control culture seem to have a higher value than the TZ culture has. In general, the average speed of cells in clumps all has a wide range among three different cultures, It is a very unstable feature and this result can’t provide a convincing proof for biological study.

4.5 Average angular speed

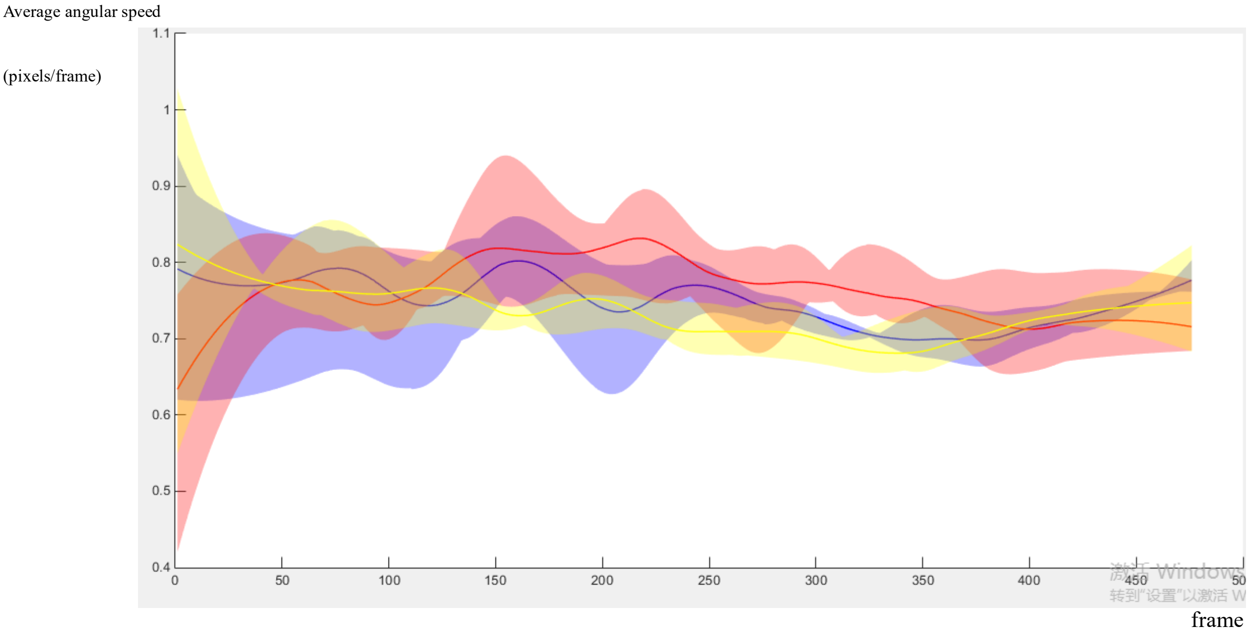


Fig. 5.11 A time-lapse graph of the average angular speed of cell of all 12 videos, comparing the difference in the number of cells among three types of culture. The blue line stands for mean value of the angular speed in control culture, the blue area stands for the intervals. The red line stands for the mean value of the angular speed in TZ culture, the red area stands for the intervals. The yellow line stands for the mean value of the angular speed in T0070907 culture. the yellow area stands for the intervals

The average angular speed is the average of angular speeds among all the trackable cells in each frame. The fig. 5.19 is the comparison of the angular speed in three different cell cultures among all 12 videos. As the figure shows, the blue line draws the mean value of angular speed in control culture, the red line draws the mean value of angular speed in TZ culture and the yellow line draws the mean value of angular speed in T0070907 culture. The blue area stands for the range of the average angular speed of cells in control culture, the red area stands for the range of the average angular speed of cells in TZ culture and the yellow area stands for the range of the average angular speed of cells in T0070907 culture. As the result shows, there isn't some obvious variance between different control groups. In biological research, it is important to have a result like this as it shows that result is consistent. It gives more confidence in these results.

4.6 Other features

There are several features which will not be plotted in this project, like the post-binding and pre-binding features. The rules to extract these features is set to be very strict so that we could receive a more accurate result. However, the strict extraction rules could also lead to a lack of sample points. In this case, the suitable points for the post-binding and pre-binding features are so few that we are not able to plot a consistent time-lapse figure using these data. While it doesn’t mean that it is not a useful feature, these features will still be extracted when we consider every cell as a single sample and use it in building the classifier of different cell cultures. With the development of the cell tracking technic, it will be possible to plot graphs of these feature when the cell tracker can generate a more accurate and stable output in the future.

4.7 Further result

After some biological analysis of the former results, it is concluded that the change in the cell density (cell population) could affect the average cell speed or average cell angular speed to some degree. To figure out the underlying relationships, the following figures Fig. 5.12 and Fig. 5.13 are further plotted to capture this relation.

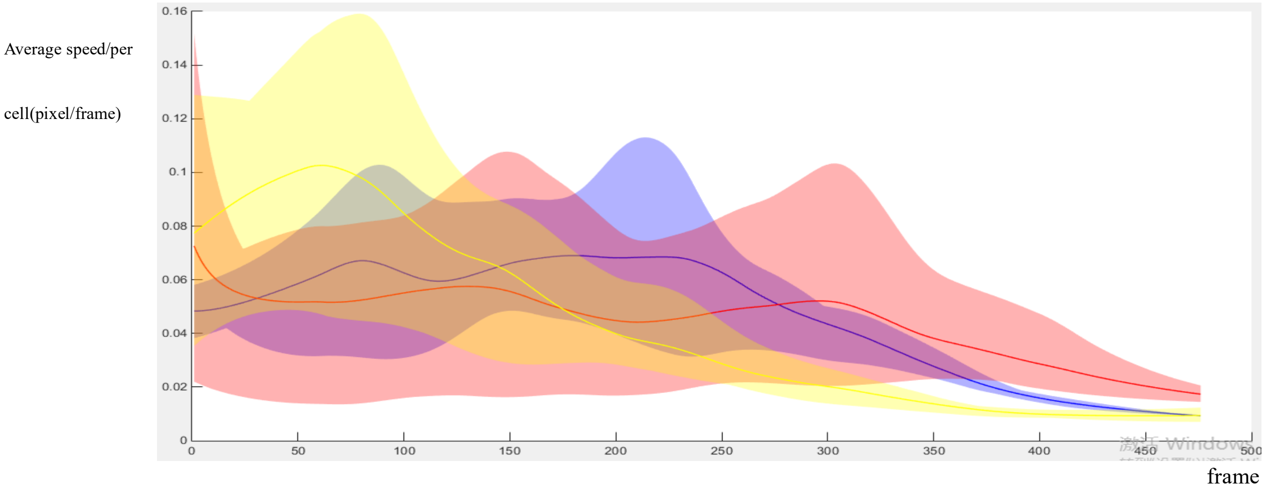


Fig. 5.12 A time-lapse graph of the average speed of cell/the number of cells of all 12 videos, comparing the difference in the number of cells among three types of culture. The blue line stands for mean value in control culture, the blue area stands for the intervals. The red line stands for the mean value in TZ culture, the red area stands for the intervals. The yellow line stands for the mean value in T0070907 culture. the yellow area stands for the intervals

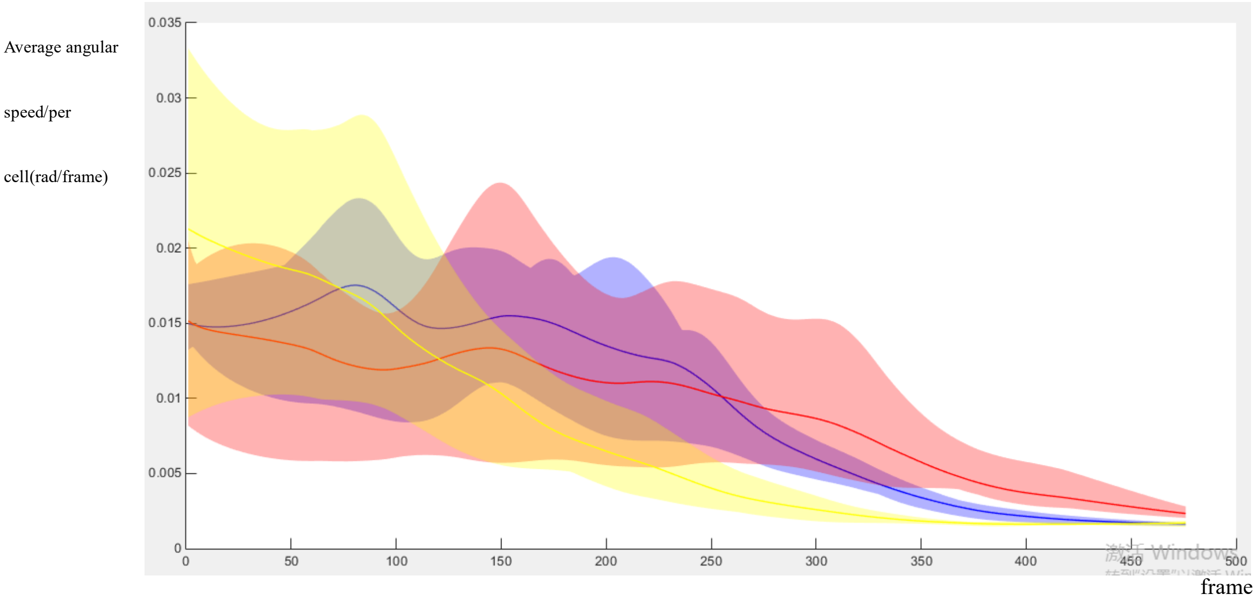


Fig. 5.13 A time-lapse graph of the average angular speed of cell/the number of cells of all 12 videos, comparing the difference in the number of cells among three types of culture. The blue line stands for mean value in control culture, the blue area stands for the intervals. The red line stands for the mean value in TZ culture, the red area stands for the intervals. The yellow line stands for the mean value in T0070907 culture. the yellow area stands for the intervals

For further biological analyze, the figure of average speed (average angular speed)/the cell population over time is generated for better observation. As the figure 5.12 (average speed /the cell population) and 5.13 (average angular speed/the cell population) shows, the blue line curves the mean value in control culture, the red line curves the mean value in TZ culture and the yellow line curves the mean value in T0070907 culture. The blue area stands the range in control culture, the red area stands for the range in TZ culture and the yellow area stands for the range in T0070907 culture. In conclusion, the speed/cell density decrease at about 250 on the axis which is the start of population growth expansion. This suggest the speed may be regulated in some way by the cell density and the speed reach its lowest when cells are confluent in the end.

4.8 SVM application

The SVM is used to build 2 binary classifiers in this project. The first one separates TZ culture with control culture and the second one separates T0070907 culture with control culture. Every cell is recognized as a single sample. For each sample, 12 features are used as the input of the SVM. In this experiment, 12 videos were divided into three control groups. Each control group contains 4 videos under the same culture to study difference in the same group. The experiment is carried out four times in total as followed:

(1) In the first experiment, the training and the test set are extracted from the same group of videos (containing TZ culture and the control culture) to build the classifier separating the control culture and the TZ culture. Although the test set and the training set are randomly selected from the same group of data, the data selected as training set will be neglected when we select the test set to make sure there was no overlap between the data in the test set and the data in the training

set. The result shows a high accuracy for the test set which has an average value about 98% out of 3 runs.

(2) In the second experiment, the training and the test set are extracted from the different group of videos (containing TZ culture and the control culture) to build the classifier separating the control culture and the TZ culture. The training set is generated from one group of the videos while the test set is generated from another group. This experiment is to test out the model's generalization ability. The result shows an accuracy for the test set which has an average value about 76% out of 3 runs.

(3) In the third experiment, the training and the test set are extracted from the same group of videos (containing t007 culture and the control culture) to build the classifier separating the control culture and the t007 culture. The result shows an accuracy for the test set which has an average value about 89% out of 3 runs. (4)

(4) In the last experiment, the training and the test set are extracted from the different group of videos (containing t007 culture and the control culture) to build the classifier separating the control culture and the t007 culture. The result shows an accuracy for the test set which has an average value about 70% out of 3 runs.

The cells in this biological experiment are all seeded manually. In this case, there will be an unavoidable deviation between the culture exposed to the same medicine. While the classification is showing a pretty good result. Due to the time limit of this project, it is a pity that I don't have time to try out all the replicates. It is an only speculation that the classifier using other replicates would work as good as the present one.

5. Conclusion and future work

The thesis presents a novel method for classification and characterization of urothelium cell culture exposed to the different PPARg agonists. It provides a potential way of analyzing cell culture automatically and fulfilling data visualization which has great effect on figuring out the trends of average cell behavior over a period of time. The result indicates a clear difference in the features, like number of cells, average speed and average clump size, among different cultures. There is also a little variance among different cultures in some other features, like the speed in clump. In general, the cells under the T0070907 culture usually have higher value in most of the features compared to the control and TZ culture. The control culture tends to have the similar trends like the T0070907 culture, but with a lower value. The TZ culture is proved to be the most inactive cell culture which has the lowest value in most features. Based on the conclusions above, it can be inferred that the addition with the PPARg activator TZ will inhibit cell behavior. The addition with the PPARg inhibitor TZ will activate the cell behaviors.

In conclusion, the addition of TZ activates a nuclear receptor called PPARg whilst T0070907 inhibits it, so they have opposite effects. The cells in the control group and T007 group grows faster than the TZ group. It suggests that the addition of drug troglitazone (TZ) affect the growth rate of cells. The results also indicate that the cell speed may be directly regulated in some way by the cell density. When cells grow and form clumps, less space was left in the cell culturing plates. Thus, the cells were unable to moving around as quickly as they were at the beginning. The cell speed/cell density graph showed that the value in both groups are similar in the final phase, but obvious difference is observed in the early stage. The addition of drugs may affect some other biological process of the cell other than cell proliferation. This may result in the difference of speed/cell density among the groups.

In this project, these experiments were only performed once and should be repeated to check their reproducibility. Due to the limited time of this project, the biological experiment only carries out once to generate this result. Although intraexperimental replicates are applied in this experiment, it is also important in biology to know if the further experiment will generate the same results a second and third time.

The research progress is not always going well. In the process of this research, I encountered many problems. For example, many efforts are made to increase the accuracy of the output data of the cell. The original cells are planted manually into the cell culture for each video. There is an unavoidable difference between the videos even if there are the same type of cell culture. To reduce this deviation, the cells are seeded several times just to generate a better video. The inaccuracy in the data output of the cell tracking software also adds a lot of difficulties to the follow-up work.

Automatic characterization and classification of different cell cultures is an area with high potential for further research. Due to the time limit of my Msc project, I can only implement some of these ideas and there is plenty of work to be done in this area. The improvement on the cell tracking technic could be the major target of the further study. With more confidence in the tracking data, it could save a lot of data screening work and make the dataset more stable and accurate in the meantime. The extracted features could be used to classify single cell behavior rather than the whole culture, such as using the cells which are different in the same culture to build the classifier (cells in a clump against cells out of clump).

The project makes some methodological improvements of the previous study done by Zhang,2016 [29]. It achieves a series of new findings and presents a way of data visualization which are helpful for further biological research and the application of machine learning algorithms in characterization and classification of different cell cultures.

References

1. Herrmann H, Bär H, Kreplak L, Strelkov SV, Aebi U (July 2007). "Intermediate filaments: from cell architecture to nanomechanics". *Nat. Rev. Mol. Cell Biol*. 8 (7): 562–73.

2. Bruce Alberts(ed.) (2002) . Chapter 21 of *Molecular Biology of the Cell* (fourth edition), published by Garland Science.

3. Bianconi, Eva; Piovesan, Allison; Facchin, Federica; Beraudi, Alina; Casadei, Raffaella; Frabetti, Flavia; Vitale, Lorenza; Pelleri, Maria Chiara; Tassani, Simone (November 2013). "An estimation of the number of cells in the human body". *Annals of Human Biology*. 40 (6): 463–471.

4. Campbell, Neil A.; Brad Williamson; Robin J. Heyden (2006). *Biology: Exploring Life.* Boston, Massachusetts: Pearson Prentice Hall.

5. Karp, Gerald (19 October 2009). *Cell and Molecular Biology: Concepts and Experiments.* John Wiley & Sons.

6.Tero AC (1990). *Achiever's Biology*. Allied Publishers.

7.Maton, A. (1997). *Cells Building Blocks of Life*. New Jersey: Prentice Hall.

8. Schopf JW, Kudryavtsev AB, Czaja AD, Tripathi AB (2007). "Evidence of Archean life: Stromatolites and microfossils". *Precambrian Research*. 158 (3–4): 141–55. Bibcode:2007PreR..158..141S.

9. Raven PH, Johnson GB (2002). *Biology*. McGraw-Hill Education. Retrieved 7 July 2013.

10. Miller, J.F., Thomson, P., Fogarty, T.C. (1998) Designing Electronic Circuits Using Evolutionary Algorithms: Arithmetic Circuits: A Case Study. In: D. Quagliarella, J. Periaux, C. Poloni, G. Winter (eds.) *Genetic Algorithms and Evolution Strategies in Engineering and Computer Science: Recent Advancements and Industrial Applications*, pp. 105–131. Wiley

11. Miller, J.F. (1999) An Empirical Study of the Efficiency of Learning Boolean Functions using a Cartesian Genetic Programming Approach. In: *Proc. Genetic and Evolutionary Computation Conference*, pp. 1135–1142. Morgan Kaufmann.

12. Miller, J.F., Thomson, P. (2000) Cartesian Genetic Programming. In: *Proc. European Conference on Genetic Programming*, LNCS, vol. 1802, pp. 121–132. Springer

13. Hicks RM. (1975) The mammalian urinary bladder: an accommodat- ing organ. *Biol Rev Camb Philos Soc* 1975;50:215–46.

14. Hu P, Meyers S, Liang FX, Deng FM, Kachar B, Zeidel ML, et al. (2002) Role of membrane proteins in permeability barrier function: uroplakin ablation elevates urothelial permeability. *Am J Physiol Renal Physiol* 2002;283:F1200–7.

15. Wu XR, Lin JH, Walz T, Haner M, Yu J, Aebi U, et al. (1994) Mammalian uroplakins. A group of highly conserved urothelial differentiation-related membrane proteins. *J Biol Chem* 1994;269:13716–24.

16. Southgate J, Harnden P, Trejdosiewicz LK. (1994) Cytokeratin expression patterns in normal and malignant urothelium: a review of the biological and diagnostic implications. *Histol Histopathol* 1999;14:657–64.

17. Feige JN, Gelman L, Michalik L, Desvergne B, Wahli W. (2006) From molecular action to physiological outputs: peroxi- some proliferator-activated receptors are nuclear receptors at the crossroads of key cellular functions. *Prog Lipid Res* 2006;45:120–59.

18. Berger J, Moller DE. (2002) The mechanisms of action of PPARs. *Annu Rev Med* 2002;53:409–35.

19. Berger JP, Akiyama TE, Meinke PT. (2005) PPARs: therapeutic targets for metabolic disease. *Trends Pharmacol Sci*. 2005;26:244–51.

20. Cohen SM. (2005) Effects of PPARgamma and combined agonists on the urinary tract of rats and other species. *Toxicol Sci* 2005;87:322–7.

21. Egerod FL, Nielsen HS, Iversen L, Thorup I, Storgaard T, Oleksiewicz MB. (2005) Biomarkers for early effects of carcino- genic dual-acting PPAR agonists in rat urinary bladder urothelium in vivo. *Biomarkers*. 2005;10:295–309.

22. Southgate J, Masters JR, Trejdosiewicz LK.(2002) *Culture of human urothelium.* New York: Wiley; 2002. p. 381–400.

23. Lobban ED, Smith BA, Hall GD, Harnden P, Roberts P, Selby PJ, et al. (1998) Uroplakin gene expression by normal and neoplastic human urothelium. *Am J Pathol* 1998;153: 1957–67.

24. Varley C, Hill G, Pellegrin S, Shaw NJ, Selby PJ, Trejdosie- wicz LK, et al. (2005) Autocrine regulation of human urothelial cell proliferation and migration during regenerative re- sponses in vitro. *Exp Cell Res* 2005;306:216–29.

25. Southgate J, Master JR, Trejdosiewicjz LK. Culture of human urothelium. RI freshney and MG freshney, book. New York; wiley: 2002.p.381-400.

26. Zhang, Z., et al. (2016) Characterization and classification of adherent cells in monolayer culture using automated tracking and evolutionary algorithms. *BioSystems*.

27. K. E. G. Magnusson(2016) *Segmentation and tracking of cells and particles in time-lapse microscopy*, Ph.D. thesis, KTH Royal Institute of Technology, 2016.

28. P. M. Gilbert, K. L. Havenstrite, K. E. G. Magnusson, A. Sacco, N. A. Leonardi, P. Kraft, N. K. Nguyen, S. Thrun, M. P. Lutolf, and H. M. Blau(2010), “Substrate elasticity regulates skeletal muscle stem cell self-renewal in culture,” *Science*, vol. 329, no. 5995, pp. 1078–1081, 2010.

29. Michalik L, Auwerx J, Berger JP, Chatterjee VK, Glass CK, Gonzalez FJ, Grimaldi PA, Kadowaki T, Lazar MA, O'Rahilly S, Palmer CN, PluTZky J, Reddy JK, Spiegelman BM, Staels B, Wahli W (2006). "International Union of Pharmacology. LXI. Peroxisome proliferator-activated receptors".

30. Dunning, Kylie R.; Anastasi, Marie R.; Zhang, Voueleng J.; Russell, Darryl L.; Robker, Rebecca L. (2014-02-05). "Regulation of Fatty Acid Oxidation in Mouse Cumulus-Oocyte Complexes during Maturation and Modulation by PPAR Agonists".

31. Belfiore A, Genua M, Malaguarnera R (2009). "PPAR-gamma Agonists and Their Effects on IGF-I Receptor Signaling: Implications for Cancer".

32. C.L. Varley, J. Southgate. Effects on PPAR agonists on proliferation and differentiation in urothelium. Experimental and Toxicologic Pathology 60 (2008) 435–441

33. Cortes, Corinna; Vapnik, Vladimir N. (1995). "Support-vector networks" . Machine Learning. 20 (3): 273–297.

34. Press, William H.; Teukolsky, Saul A.; Vetterling, William T.; Flannery, Brian P. (2007). " Support Vector Machines".

35. Benko, Attila; Dosa, Gyorgy; Tuza, Zsolt (2010). "Bin Packing/Covering with Delivery, solved with the evolution of algorithms". 2010 IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA). pp. 298–302.

36. Ashlock, D. (2006), Evolutionary Computation for Modeling and Optimization

37. Bishop, Christopher M. (1995). Neural networks for pattern recognition. Clarendon Press.

38. Zhihua Zhou.(2016) “NERUAL NETWORK”. Machine Learning. ISBN 978-7-302-42328-7.