BBA - Proteins and Proteomics 1865 (2017) 1719–1727

Contents lists available at ScienceDirect

BBA - Proteins and Proteomics

蛋白质生物学

journal homepage: www.elsevier.com/locate/bbapap

预备知识之超平面：其实很好理解，就是在三维空间里面，其平面就是现在所说的平面；在二维空间中平面的定义为一条直线同理四维空间中的平面定义为一个三维图形，以此类推（越是简单的定义觉不好理解，源哥可以结合我翻译过来的中文试图去理解一下，或者查一些资料）

(Machine-)Learning to analyze in vivo microscopy: Support vector machines[☆]

机器学习

支持向量机

支持向量机：
SVMs,can be used for anakyze data , regression analysis , recognize patterns and classification

机器学习：
研究计算机怎样模拟和实现人类的学习行为，以获取新的知识和技能，不断改善自身的性能。是人工智能的核心，是计算机具有智能的根本途径，主要使用归纳，综合而不是演绎.
Machine learning is a scientific discipline that explores the construction and study of algorithms that can learn from data.
key point:build a model ; overlap with computational statistics ; prediction-making

Michael F.Z. Wang[a,b], Rodrigo Fernandez-Gonzalez[a,b,c,d,*]

[a] Institute of Biomaterials and Biomedical Engineering, University of Toronto, Toronto, ON M5S 3G9, Canada
[b] Ted Rogers Centre for Heart Research, University of Toronto, Toronto, ON M5G 1M1, Canada
[c] Department of Cell and Systems Biology, University of Toronto, Toronto, ON M5S 3G5, Canada
[d] Developmental and Stem Cell Biology Program, The Hospital for Sick Children, Toronto, ON M5G 1X8, Canada

ARTICLE INFO

Keywords:
Machine learning
Image analysis
In vivo microscopy
Model organisms

摘要：
采用高分辨率的显微技术，长期探测细胞及其亚显微结构中的生命体的动力学特征，揭露了基本的组织形成修复方面的问题。
现存的一些问题，信息储存、高分辨率图像的分析、延时图像在手动处理上过于复杂。
问题的解决：
利用SVMs，用高效的分析方法分析图像。
主要内容：
1.活体成像分析：引入数学模型、测量、图像分类；
2.不同的SVM，不同的细胞分裂和追踪的算法；
3.线虫世系追踪：酵母检测中的蛋白质定位；果蝇遗传中基因表达的动力学。
结论：
在复杂的图像处理系统以及新型的微观模型。

附：关于支持向量机以及机器学习一些定义和介绍可以参见各种百科（baidu&wiki）

ABSTRACT

The development of new microscopy techniques for super-resolved, long-term monitoring of cellular and subcellular dynamics in living organisms is revealing new fundamental aspects of tissue development and repair. However, new microscopy approaches present several challenges. In addition to unprecedented requirements for data storage, the analysis of high resolution, time-lapse images is too complex to be done manually. ~~Machine learning techniques are ideally suited for the (semi-)automated analysis of multidimensional image data.~~ In particular, support vector machines (SVMs), have emerged as an efficient method to analyze microscopy images obtained from animals. Here, we discuss the use of SVMs to analyze in vivo microscopy data. We introduce the mathematical framework behind SVMs, and we describe the metrics used by SVMs and other machine learning approaches to classify image data. We discuss the influence of different SVM parameters in the context of an algorithm for cell segmentation and tracking. Finally, we describe how the application of SVMs has been critical to study protein localization in yeast screens, for lineage tracing in *C. elegans*, or to determine the developmental stage of *Drosophila* embryos to investigate gene expression dynamics. We propose that SVMs will become central tools in the analysis of the complex image data that novel microscopy modalities have made possible. ~~This article is part of a Special Issue entitled: Biophysics in Canada, edited by Lewis Kay, John Baenziger, Albert Berghuis and Peter Tieleman.~~

目前对于高精度成像的显微镜技术和来自生物样本的无光毒性成像使得获取大容量的经过长期观察得到的生命体的发展和过程得以实现。高分辨率的图像和光切成像有着史无前例的复杂性。同时，大数据的生成对于显微镜技术是一项巨大的挑战。尤其是基于加注释于图像的传统分析方法，捕捉储存于图像的信息时未免有些捉襟见肘。人工处理是不可行的，因为其复杂性和耗时性并且因为被限制的数据处理方法很容易产生偏差。然而，机器学习可以理想地适用于大容量的生物数据的考察。机器学习，结合于自动化的图像分析，可以被应用于图像的分割从而探测细胞以及亚细胞尺度下的表现型或者对整个生命体的发展阶段进行分类。

1. Introduction

The advent of novel microscopy techni for better spatial resolution [1], and non-phototoxic imaging of biological samples [2] has enabled high-content, and long-term monitoring of developmental and physiological processes in living organisms [3–7]. Super-resolution and light-sheet microscopy images possess unprecedented complexity. The massive volumes of data generated by these techniques have brought about new challenges for microscopy. In particular, conventional analysis approaches, which depend on the intervention of a user for image annotation, fail at capturing all the information stored in the images [8]. User involvement is often impractical, as it is cumbersome and time-consuming; and can introduce biases and limit the types and quality of the analyses conducted. In contrast, machine learning is ideally suited to mine the contents of biological images. Machine learning, in combination with automated image analysis, can be applied to group pixels for image segmentation, to detect phenotypes at the cellular and subcellular scales, or to categorize developmental stages at the level of entire organisms.

Machine learning describes a set of statistical methods to classify input data (individual pixels, groups of pixels, images, etc.) into different categories. Traditionally, machine learning techniques have been organized into supervised and unsupervised approaches. In supervised machine learning, an expert user specifies the categories for a subset of data, usually referred to as the training set. By analyzing the common features of data in a specific category (in the case of cells of a certain type, for instance, range of sizes, shapes, fluorescence intensities and distributions of different molecular markers, etc.), the algorithm can predict the category that new data belong to. Supervised learning algorithms differ in how they map input data to output categories based on the information provided by the training set. In unsupervised methods, no a priori information is given to the algorithm. At best, a user may specify the number of categories that are expected in the data. The algorithm tries to find groups within the provided data based on

[☆] This article is part of a Special Issue entitled: Biophysics in Canada, edited by Lewis Kay, John Baenziger, Albert Berghuis and Peter Tieleman.
[*] Corresponding author at: Institute of Biomaterials and Biomedical Engineering, University of Toronto, Toronto, ON M5S 3G9, Canada.
E-mail address: rodrigo.fernandez.gonzalez@utoronto.ca (R. Fernandez-Gonzalez).

http://dx.doi.org/10.1016/j.bbapap.2017.09.013
Received 4 July 2017; Received in revised form 25 September 2017; Accepted 27 September 2017
Available online 30 September 2017

预备知识之支持向量机（在前面也已经提到过）：
支持向量机是基于统计学习理论的新的机器学习方法，其主要用途是解决分类问题和回归问题：将线性不可分的样本映射到高维的特征空间，然后再进行线性分类，最后实现对原空间的非线性分类

Box 1

Libraries available for the adoption of SVM.

SVM拥有强大的程序库支持，使用多种程序语言，可以以最简便的分析方式处理数据：
svmlight/libsvm/scikit-learn/weka

A number of freely available libraries exist to incorporate SVMs to an image processing and/or analysis pipeline with minimal effort. Some of them are:

- SVM^{light} [39]. Implemented in the C programming language, SVM^{light} uses a scalable memory approach, and thus can handle thousands of support vectors. Interfaces exist for a variety of other computer languages, including Matlab, Python, and Java, thus providing the efficiency of C within rapid prototyping environments. Available at http://svmlight.joachims.org.
- LIBSVM [40]. There are versions of LIBSVM implemented in C/C + + and in Java, thus facilitating integration with other programming environments. LIBSVM also provides Python, R, and Matlab interfaces, as well as CUDA extensions for parallel computing using Graphics Processing Units (GPUs), which results in significantly reduced processing times. Available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.
- Scikit-learn [41]. Scikit-learn is a machine learning toolbox for the Python programming language. Scikit-learn uses the C implementation of LIBSVM interfaced to Python. In addition to SVMs, Scikit-learn implements many other machine learning methods, including supervised and non-supervised algorithms. Available at http://scikit-learn.org/stable.

WEKA [42]. WEKA is a collection of machine learning algorithms implemented in Java. Similar to Scikit-learn, WEKA uses LIBSVM to implement SVMs. Available at http://www.cs.waikato.ac.nz/ml/weka.

SVMs不仅可以线性计算：
在SVMs中如果一个超平面是没有被定义的，那么支持向量机可以将其转化为已经存在的或者是曾经被定义过的模型。支持向量机可以在通过高维度下处理等方式来解决现有模型不能解决的问题

feature analysis, as described above, creating new categories as sufficiently distinct data emerge. Thus, supervised algorithms are useful in the categorization of data into predetermined groups (pattern classification), while unsupervised methods provide the opportunity to uncover structure and organization within the data (pattern discovery).

Support Vector Machines (SVMs) are a supervised learning technique that has become popular in the analysis of biological data [9–11]. Microarray analysis experiments demonstrate that SVMs are fast, even when analyzing training sets with a million of samples; and the classification accuracy of SVMs outperforms that of other machine learning methods. Furthermore, SVM implementation is relatively easy due to the availability of several programming libraries that facilitate this task (Box 1). Here, we review the application of SVMs to the analysis of in vivo imaging data. We introduce the mathematical framework behind SVMs and the features used to analyze microscopy images from model organisms. In this context, we discuss SVM applications to investigate biological questions at the subcellular, cellular, and animal scales.

2. Learning from a teacher: support vector machines

支持向量机分为两个部分（它是以二进制方式处理数据）
test set：负责结果精确性
training set：
每个数据点被看做是多维向量，这里向量维数的确定来源于所需要的维数（即维数不是一成不变的）；
在这里数据被分为两种：‘ 红色 ’和‘ 蓝色 ’

2.1. An introduction to support vector machines

SVMs are a supervised machine learning algorithm [11]. Supervised algorithms use two data sets: a training set, to learn about the categories present in the data; and a test set, to establish the accuracy of the algorithm and/or to conduct data analysis. Each data element in the training set is considered an m-dimensional vector, where m is the number of features used to characterize the element (Box 2, Fig. 1A). In addition, an expert user classifies each element in the training set as belonging to one of two categories (Fig. 1A, red and blue). The purpose of the SVM is to build an m-dimensional hyperplane that provides the maximum separation (or margin) between the closest vectors in each category (Box 3, Fig. 1A, black line). The vectors that define the position of the hyperplane are referred to as support vectors, as removing them would change the position of the hyperplane (Fig. 1A, black circumferences). After defining the SVM, subsequent data will be classified as belonging to one category or the other based on their position in space with respect to the hyperplane.

源哥，这里关于黑线的描述不太懂，你试着看一下！（就是这里下划线的部分）

SVMs are binary classifiers, as they distinguish between two categories; and non-probabilistic, because they unequivocally assign new data to one of two categories. It is possible to use SVMs to achieve multi-class classification using different strategies. For instance, in a one-vs.-all approach, an SVM is trained per class to determine if data belong or not to that class. Another approach for multi-class classification with SVMs is one-vs.-one, in which a different SVM is defined for each possible pair of classes, and new data are scored by each of the classifiers to find the optimal category, based, for instance, on the distance to the separating hyperplanes. Thus one drawback of multi-class schemes using SVMs can be their computational cost.

Notice that linearity is not a strict requirement of SVMs. In cases where a linear hyperplane cannot be defined (Fig. 1B), it is often possible to transform existing features or define additional ones to solve the classification problem in a different feature space or in a higher-dimensional one in which the two categories can be separated by a hyperplane (Fig. 1B′). The functions used to change the feature space of the problem are referred to as kernels. The optimal hyperplane can be mapped back into the original feature space (Fig. 1B). A note of caution: adding features can lead to “overfitting”, or the definition of a hyperplane that is too specific to the training data, and thus likely to misclassify test data (Fig. 1D). To avoid overfitting, a soft margin is often used, in which misclassification of some of the training data is allowed in order to obtain a hyperplane that can be better generalized to classify test data (Fig. 1D′). The degree to which misclassifications are allowed in the training set depends on the value of regularization parameter, C, that must be balanced to ensure proper training of the classifier without overfitting the training set (Box 3).

2.2. An example: applying SVMs to detect dividing cells in Drosophila embryos

利用支持向量机探测果蝇胚胎中的细胞分裂

方法：利用支持向量机中延时摄像的功能来将细胞分裂的时间缩短
利用分水岭算法：将像素的强度转化为高度值，模仿流动过程

SVMs can be integrated in image processing and analysis pipelines. As an example, we will describe the use of SVMs in the context of an algorithm for cell segmentation and tracking that can handle dividing cells. Our data set consists of time-lapse confocal microscopy images of *Drosophila* embryos expressing Gap43:mCherry, a fluorescent cell membrane reporter [12] (Fig. 2A). The segmentation is based on a region-growing method, the watershed algorithm [13]. The watershed algorithm considers pixel intensities as elevation values, and simulates a flooding process starting at a collection of seed points, one per object. When water overflows an object (i.e., when it would spill into the adjacent object), a watershed line is built that delineates the boundary between the objects (Fig. 2A′). A requirement for the watershed algorithm is the identification of one (and only one) seed point per object, which can be done manually or using automated methods [14,15]. The presence of more than one seed per cell leads to oversegmentation, or objects that are incorrectly split into multiple ones in the segmentation results. Too few seeds will cause undersegmentation, or the merging of objects in the results.

荧光探针

分水岭算法

An efficient way to integrate cell segmentation and tracking in time-lapse microscopy sequences consists of propagating the seeds used to

这里的分水岭算法主要用于细胞分裂，他通过对像素点的灰度值的处理来得到一种‘ 地形起伏 ’的图像。分水岭图像中的线条由节点，边缘或者是节点和边缘共同组成。
A grey-level image may be seen as a topographic relief,where the grey level of a pixel is interpreted as its altitude in the relief. In graphs, watershed lines may be defined on the nodes, on the adges, or hybrid lines on both nodes and edges.Watersheds may also be defined in the continuous domain.

下面的四个附注是我从下面的这几个仪器中找到我可以看懂的一些部分，有些学术性太强以至于虽然翻译过来但是却没有一点用处，如果源哥愿意的话你可以看一下这几个仪器（但是我觉得用处不大）

further more：
在生物图像中通过测量可以得到直观的形态学特征，包括面积，周长，体积，表面积等，或者是强度的测量，比如像素的平均值，中位数或者是每一个可能出现的像素点的频率

Box 2

What to learn: useful features for the analysis of biological images.

The features used to classify objects formed by groups of pixels in biological images can be intuitive morphological measurements, including area, perimeter, volume, or surface area; or intensity measurements, such as mean pixel value, median, or a histogram with the relative frequency of occurrence of each possible pixel value (in this case, each histogram bin could be considered as an independent feature). In addition, more complex features are commonly used:

- *Feret diameters* [43]. Feret diameters are shape measurements that quantify the distance between two tangents to an object, parallel to each other and with a specific orientation with respect to the object. Feret diameters are calculated at multiple orientations for any given object. Feret diameters can be thought of as the size of an object when measured by a Vernier caliper. With increasing irregularity in shape, the maximum and minimum Feret diameters will diverge more from each other. Thus, the ratio of maximum-to-minimum Feret diameters can be used to quantify object elongation, such as in the measurement of cell alignment.
- *Gabor filters* [44,45]. Texture measurements quantify the spatial distribution of pixel values. Gabor filters are texture measurements used to detect the presence of image features with specific orientations and frequencies. Gabor filters are the product of a Gaussian filter that makes the measurement local by limiting the region of the image considered, and a sinusoidal wave that restricts the frequency and orientation sought after. A Gabor filter will return a high signal on a region of the image that contains structures that match the periodicity and orientation of its sinusoid. Thus, for example, Gabor filters could be used to detect and characterize directional protrusions extended by a cell, or to classify cells as protrusive or not by comparing their Gabor filter values to those of cells that do assemble oriented protrusions.
- *Haralick textures* [46]. Haralick features are also texture measurements. To define the Haralick textures, gray level co-occurrence matrices must be defined first. The co-occurrence matrix is defined for a certain orientation or (x, y) offset, and it encodes in element (i, j) the number of times that a pixel with value j is found with an offset (x, y) with respect to a pixel with value i. In other words, the co-occurrence matrix stores the frequency of a pixel value at a specific orientation and distance from a different pixel value. Based on the co-occurrence matrices, it is possible to define 14 features including contrast (a measurement of gray level variation between a pixel and its neighbours), homogeneity, entropy (randomness in the gray values, greatest when all the elements of the co-occurrence matrix have the same value, i.e., all gray values occur with the same frequency), correlation, image moments (mean, standard deviation, asymmetry, and kurtosis or sharpness of the peak of the gray level distribution), etc.
- *Hu moments* [47]. Hu moments are shape measurements based on a binary representation of the object. They are moment invariant, indicating that they are insensitive to scaling, translation, or rotation. Thus, for example, two objects with the same shape but different sizes and/or inverted along one axis will have the same Hu moment values.
- *Local structure* [24]. Local structure features quantify multiscale morphological and intensity parameters for cells. The calculation of local structure features involves the application of an adaptive threshold to the original image. In adaptive thresholding, each pixel in the image is compared to a different threshold to determine if the pixel belongs to an object or to the image background. For each pixel, the threshold is the mean value of the pixels within a window of size S around the pixel in question. Local structure features are calculated by applying adaptive threshold with increasing values of S to the original image, and calculating morphological and intensity features for the objects detected by thresholding. Smaller values of S quantify fine features, while larger values quantify coarser features. Feature values are computed relative to similar values calculated after cell segmentation (using the region-growing, watershed algorithm [13]), to normalize changes in the values of the features from finer to coarser structures.
- *Scale Invariant Feature Transform (SIFT) features* [48,49]. SIFT features are largely invariant to scale, rotation, viewpoint, or illumination. Gaussian (smoothing) filters of increasing standard deviation (i.e. scale) are applied to the image, and the outputs of consecutive filters are subtracted, resulting in a set of difference-of-Gaussian images. Local minima and maxima (keypoints) are detected in the three-dimensional set of difference of Gaussian images. Keypoints displaying low contrast, or on edges (which are sensitive to noise) are discarded. For the remaining keypoints, histograms of gradient orientation and magnitude for the pixels around the keypoint are constructed, and the dominant gradient orientation and magnitude are assigned to the keypoint. New keypoints are created at the same location if the histograms display multiple peaks. This approach accomplishes scale and rotation invariance. To obtain viewpoint and illumination invariance, gradient magnitude histograms as a function of orientation are calculated at different locations with respect to the keypoint (i.e. using the keypoint as the top-left or right, or bottom-left or right corner of the sampling window) and assigned as features to the keypoint. This approach allows recognition of intensity patterns at specific positions with respect to the keypoint.
- *Zernike moments* [50]. Originally defined to represent optical aberrations, Zernike polynomials do not contain redundant information [51]. Zernike moments are sensitive to scaling and translation, but their magnitude is rotation invariant. Thus, objects must be segmented, centered and resized before measuring Zernike moments if position or size must not affect the classification of an object. Because Zernike polynomials are non-redundant, the combination of the Zernike moments of an image reconstructs the original image. Reconstruction of a continuous image in principle requires an infinite number of Zernike moments. However, comparison of the original image to the reconstruction obtained from a limited number of Zernike moments is indicative of the amount of information included in those moments.

Wavelet transform features [52]. A Fourier transform decomposes an image into its different frequency components, which can be used to characterize cellular morphology [16]. However, the Fourier transform of an image does not consider the duration (or scale in space domain) of different frequencies. Wavelet transforms provide both frequency and scale information, i.e. for every frequency component, it is possible to establish the scale with which it can be found in the image. Different wavelets (not necessarily sines and cosines, as is the case in Fourier transforms) can be used to "probe" the image, and by varying the scale of the wavelet, it is possible to determine at which size the wavelet occurs. Wavelet parameters can be used to characterize objects of different sizes (e.g. protrusions of different lengths), and the scaling factor can also be used to normalize wavelet features and make them scale-invariant.

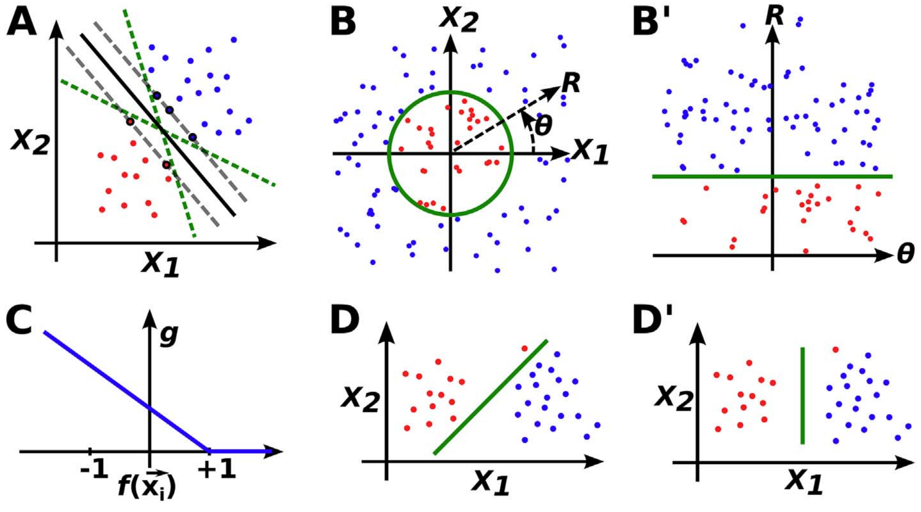

Fig. 1. Support vector machines for data classification. (A) Cartoon example showing training data from two different categories (red and blue) represented as a function of two features, x_1 and x_2. Green dotted lines indicate potential hyperplanes separating the two categories. The solid black line defines the optimal classification hyperplane with the greatest margin (dashed gray lines). Support vectors are indicated by black outlines. (B) Separation of data from two different categories (red and blue) is not always possible with a linear classifier (B). However, features can be transformed (B′, $R = \sqrt{x_1^2 + x_2^2}$, $\theta = tan^{-1}\frac{x_2}{x_1}$), or additional features can be added to represent the data in a space in which the two categories can be separated by a linear boundary (B′, green line). The boundary can be mapped to the original feature space (B, green circumference). (C) Hinge-loss function (g, y-axis) for a specific classifier result ($f(\mathbf{x_i})$, x-axis). (D) Overfitting may lead to the definition of a hyperplane that correctly classifies all training samples, but that its suboptimal in terms of its margin (D, green). Allowing misclassification of some training data can produce a hyperplane with a greater margin (D′, green).

segment the cells in an image to subsequent images [16]. Seed positions can be adjusted by quantifying local tissue movements using particle image velocimetry. However, dividing cells represent a challenge for seed propagation, as seeds need to be split when a cell divides to ensure the presence of one seed per cell. Logistic regression, a supervised machine learning algorithm that returns a probability that an object belongs to a certain class, has been used to detect cells that are about to divide based on how their morphology changes over time and split the corresponding seed into two before applying the watershed algorithm [16] (Fig. 3B). Here, we illustrate the use of SVMs for similar purposes.

We applied SVMs to identify dividing cells and prevent under-segmentation errors. After defining seeds in the first time point of time

Box 3
Mathematical formulation of SVMs.

Mathematically, SVMs are classifiers, $f(\mathbf{x})$, such that:

$$f(\mathbf{x}) = \begin{cases} \geq 0, y = +1 \\ <0, y = -1 \end{cases} \tag{1}$$

where $\mathbf{x}$ is a vector of m input features, and y is the classification output corresponding to $\mathbf{x}$ (in this formulation, -1 or 1 to distinguish between two possible categories). A training set consists of known ($\mathbf{x}$, y) pairs. Assuming a linear classifier, it can be expressed with the scalar equation of a hyperplane:

$$f(\mathbf{x}) = b + \theta_1 x_1 + \theta_2 x_2 ... + \theta_m x_m = b + \boldsymbol{\theta}\mathbf{x} \tag{2}$$

where x_j are the components of $\mathbf{x}$, $\boldsymbol{\theta}$ is a vector perpendicular to the hyperplane, and $\mathbf{w} = (b, \boldsymbol{\theta})$ defines the classifier. Given the training set, defining the SVM requires finding the optimal value of $\mathbf{w}$. The optimal value of $\mathbf{w}$ maximizes the margin (Fig. 1A, black line). Several algorithms exist to calculate $\mathbf{w}$. A commonly used method consists of minimizing the mean cost, J, of the resulting classification:

$$J(\mathbf{w}) = \frac{1}{n} \sum\nolimits_{i=1}^{n} g(\mathbf{w}, \mathbf{x_i}) \tag{3}$$

where n is the number of training samples, $\mathbf{x_i}$ is the feature vector for training sample i, and g is a cost function. Cost functions can also be defined in multiple ways. A frequently used cost function is a "hinge-loss" (Fig. 1C):

$$g(\mathbf{w}, \mathbf{x_i}) = max(0, 1 - y_i(b + \theta\mathbf{x_i})) = \max(0, 1 - y_i f(\mathbf{x_i})) \tag{4}$$

where y_i is the classification (-1 or $+1$) corresponding to training sample i. The hinge-loss function has a value of 0 if sample i is correctly classified (y_i and $f(\mathbf{x_i})$ have the same sign). However, if sample i is not correctly classified, the value of the hinge-loss function will increase linearly with $f(\mathbf{x_i})$.

The optimal value of $\mathbf{w}$ may not generate the best classifier if it leads to a very narrow separation between the two classes (Fig. 1D). Multiple classifiers based on different values of $\mathbf{w}$ can be constructed, and the final classification can be determined by polling the most frequent classifier output. More frequently, the algorithms to optimize $\mathbf{w}$ are extended to maximize the margin, even if some training data are misclassified (Fig. 1D′). The requirement for perfect classification of the training set can be relaxed using a regularization parameter, often referred to as C, which determines how important it is to correctly classify each sample in the training set. C is introduced in the mean cost function as:

$$J(\mathbf{w}) = \left[\frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i f(\mathbf{x_i}))\right] + C\ \|\boldsymbol{\theta}^2\| \tag{5}$$

Large values of C will increase the mean cost associated with classification errors, and thus they will converge towards hyperplanes with small margins and will tend to over-fit the training data. In contrast, small values of C will result in hyperplanes with large margins that misclassify some of the training data.

Table 1
Average feature values for dividing and non-dividing cells. Values are mean ± standard error of the mean. The values are for 70 dividing cells in seven embryos, and 100 non-dividing cells in five embryos.

Feature	Dividing cells	Non-dividing cells
Maximum change in cell area	2.058 ± 0.148	1.422 ± 0.043
Minimum shape factor	1.304 ± 0.011	1.373 ± 0.010
$\min_1$/max radii	0.360 ± 0.007	0.412 ± 0.011
$\min_2$/max radii	0.424 ± 0.006	0.491 ± 0.012
1st Fourier frequency	0.187 ± 0.015	0.160 ± 0.016
2nd Fourier frequency	2.155 ± 0.046	0.978 ± 0.053
3rd Fourier frequency	0.412 ± 0.032	0.417 ± 0.027
4th Fourier frequency	0.485 ± 0.023	0.242 ± 0.014
5th Fourier frequency	0.225 ± 0.014	0.135 ± 0.008
6th Fourier frequency	0.214 ± 0.014	0.098 ± 0.005
7th Fourier frequency	0.121 ± 0.008	0.073 ± 0.004
8th Fourier frequency	0.106 ± 0.007	0.069 ± 0.004
9th Fourier frequency	0.082 ± 0.005	0.050 ± 0.003
10th Fourier frequency	0.064 ± 0.005	0.041 ± 0.002

lapse microscopy images, the watershed algorithm was applied to segment the cells. Based on the segmentation results, we used a linear SVM to decide if any cell was about to divide, and in that case, split the corresponding seed into two before propagating the seeds to the next time point. The SVM was trained using a group of cells that had been segmented after manually editing the seeds to account for cell divisions. Fourteen morphological features were measured on the polygons resulting from the watershed segmentation [16] (Table 1). The features included the maximum change in cell area (with area change measured as the ratio of the current area to the minimum recorded cell area), the minimum recorded shape factor (perimeter^2-to-area ratio, an measurement of circularity), and a number of features that change their value as cells round up during metaphase (Fig. 3B, − 225 s) and become dumbbell-shaped during cytokinesis (Fig. 3B, − 15 s): the ratios of each of the two shortest cell radii to the longest radius, and the first ten frequency components of the Fourier transform of the distance from the cell centroid to each one of the pixels on the cell surface.

We integrated the SVM in our watershed segmentation-seed propagation algorithm. The SVM was applied to the polygons resulting from the segmentation of previous time points. When a cell was classified as dividing, the corresponding seed was split into two, and the two new seeds were placed along the longest cell axis, at half of the distance between the cell centroid and the cell boundary. We used a test set formed by 30 dividing and 60 non-dividing cells in 7 embryos. To investigate how the size of the training set affected classification accuracy, we used training sets of increasing sizes, all with a 1:1 mixture of dividing and non-dividing cells. Using a training set including only 8 cells, we correctly classified over 90% of the cells in the test set, with 3% of non-dividing cells classified as dividing (false positives). Using a training set with 72 cells, we correctly classified over 97% of the test cells, with a similar fraction of false positives (Fig. 2B). We also investigated the importance of the regularization parameter, *C*. As expected, we found that both large and small values of *C* led to reduced classification efficiencies (Fig. 2C). In the case of large *C* values, none of the samples in the training set were misclassified, thus causing overfitting (Fig. 2D). In contrast, small *C* values caused up to 16% of the elements in the training set to be misclassified. Values of the regularization parameter in the range of 0.2–17.6 (corresponding to 10–1% misclassified training set elements, respectively) yielded classification accuracies over 97% (Fig. 2D). Thus, SVM can be used to accurately detect, segment, and track dividing cells, thus enabling quantification of the dynamics of cell division (timing, duration, orientation, etc.) in developing tissues. Importantly, care must be taken in the selection of SVM parameters, particularly training set size and softness of the boundary.

3. SVM applications

3.1. Molecular scale: screening for protein localization

SVMs have been extensively used to characterize diffraction-limited protein localization patterns in large-scale screens using yeast. Initial efforts were based on a publicly available collection of yeast strains expressing proteins tagged with the green fluorescent protein (GFP) [17]. The collection, created at the University of California, San Francisco, covered 75% of the yeast genome, and included 2713 images displaying the patterns of localization of the GFP-tagged proteins. Images were manually curated to define 20 localization patterns, and proteins were manually assigned to one or more of these patterns. SVMs were defined to automate the classification of further patterns and detect new ones [18]. Features were measured at the level of cell populations, without cell segmentation (86 features, including morphology, Gabor and Haralick textures), and at the single-cell level, after cell segmentation (185 features, including cell shape, Zernicke moments, Haralick textures, and wavelet features) [19]. The authors

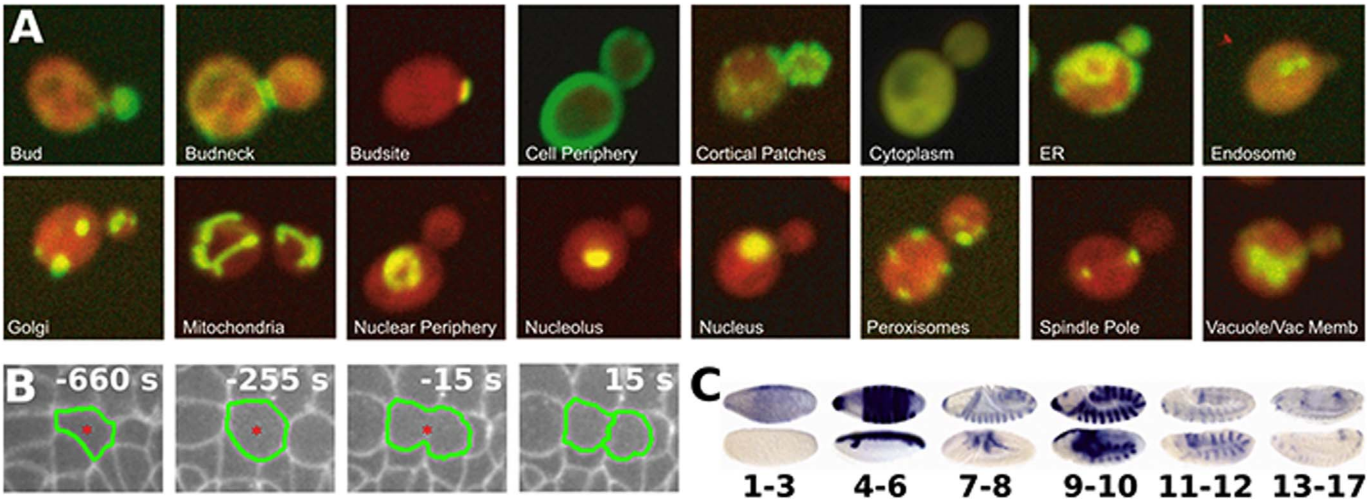

Fig. 2. Using SVMs to detect and track cell division. (A) Spinning disk confocal microscopy images of cells in a *Drosophila* embryo expressing Gap43:mCherry (a membrane marker) before (A) and after (A') watershed-based segmentation. Green polygons indicate segmented cells, red circles show the seeds points for the watershed algorithm. (B–C) Percentage of correctly classified dividing (red), non-dividing (blue) and total (black) test cells with respect to the number of cells in the training set (B, $C = 1$) and the value of the regularization parameter, C (C, training set formed by 80 cells). Note the difference in the Y-axis range between B and C. (D) Percentage of misclassified training set cells for different values of the regularization parameter, C (training set formed by 80 cells).

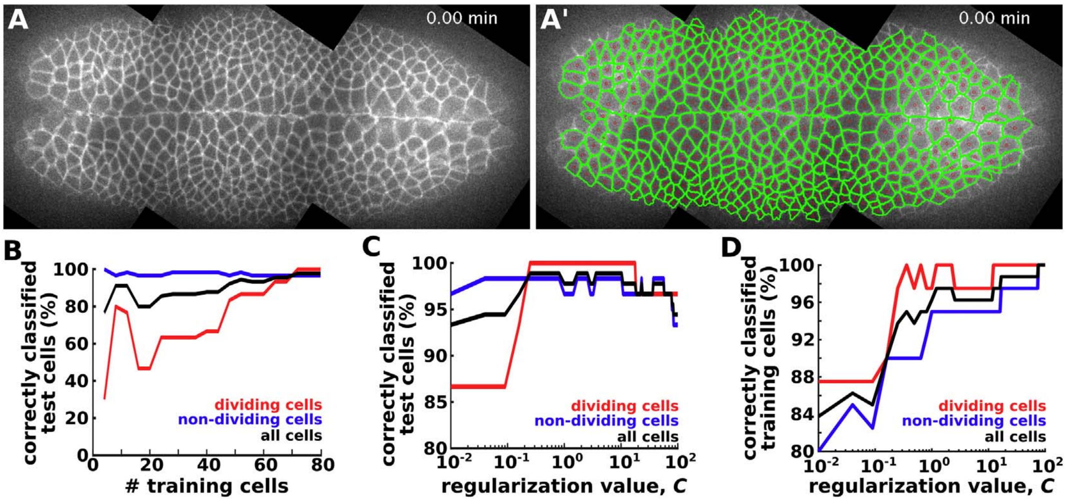

Fig. 3. SVMs can be applied to distinguish between categories in diverse biological processes. (A) Yeast cells expressing GFP-tagged proteins (green) with different subcellular localization patterns. SVMs can be used to distinguish between these patterns based on shape of the fluorescent region, intensity, and texture. Reproduced with permission from [21]. (B) Cells in a *Drosophila* embryo expressing a fluorescently-tagged cell membrane marker. A cell that enters mitosis is outlined in green. The cell rounds up during metaphase (–225 s), and takes on a dumbbell shape during cytokinesis (–15 s). These cell shape changes can be used to train an SVM to distinguish between dividing and non-dividing cells. Time is with respect to cell division. Reproduced with permission from [16]. (C) *In situ* hybridizations showing gene expression patterns at different stages of embryonic development in *Drosophila*. Developmental stages are indicated. SVMs can distinguish between different developmental stages using multiscale texture features based on Gabor filters. Images are from the BDGP [32, 33]. Reproduced with permission from [34].

defined 25 binary classifiers to distinguish between pairs of categories [20]. Each classifier was trained using 6-fold cross-validation [18]. In *k*-fold cross-validation, the data are split into *k* groups, and *k*-1 groups are used to train the classifier, while the last group is used for testing the accuracy of classification. The procedure is repeated *k* times, each time using a different group for testing, such that all groups are used for testing once. The classification accuracy is calculated as the mean result for each one of the *k* iterations. Because six different subsets of data were used to train each of the 25 classifiers, a total of 150 different classifiers were developed. Images were annotated based on the majority vote of the 150 classifiers, providing a confidence level in the assignment using the fraction of classifiers that returned the category selected by the majority. While time-consuming, this approach was able to correctly classify the four most common localization patterns (cytoplasm, nucleus, mitochondrion, and endoplasmic reticulum) with an accuracy of over 90% using field-level features. The classification accuracy decreased to 81% when all 20 possible localization patterns were used, due to the difficulty in training for patterns represented by few images in the training set. Notably, 501 images were identified in which the algorithm classification did not match the visual classification, and some of the corresponding proteins were verified to have mixed localization patterns that were better captured by the SVM algorithm than by the visual score. Thus, SVM-based methods are not only useful to classify images, but also to detect bias and errors in human-based image annotation.

利用SVMs可以研究蛋白质组的动力学

The powerful combination of yeast-based screens and SVMs has been used to investigate proteome dynamics, or how different treatments affect protein localization [21]. Yeast strains expressing a cytosolic red fluorescent protein (RFP) to visualize cell boundaries, and one of 4100 proteins tagged with GFP were imaged by high-throughput spinning disk confocal microscopy. Several hundred features based on cell shape, GFP and RFP intensity, and GFP textures at multiple scales were used to characterize the cells. Sixteen potential protein localization patterns were defined a priori, from visual inspection of images in an initial screen of wild-type cells (Fig. 3a). The training set consisted of over 70,000 cells. Binary classifiers were developed to identify individual localization patterns, but also for quality control of the data (e.g. presence/absence of cells, dead cells, or cell cycle stage). The automated classification system produced results that were in agreement with previous visual annotations [17] 94% of the time [21]. Misclassified proteins displayed low abundance or belonged to a subcellular compartment with a similar morphology to another compartment [21], suggesting that additional features independent of organelle shape may further increase the accuracy of classification. SVM-based classification was applied to proof-of-principle studies to quantify changes in protein localization upon pharmacological treatments that disrupted yeast proliferation (hydroxyurea and rapamycin), and in mutants for a lysine deacetylase, Rpd3, which regulates gene expression by posttranslational modifications of histones. A total of 15 screening conditions were tested, and over 20 million cells were analyzed. The analysis of hydroxyurea-treated samples identified previously unreported changes in the localization of 40 proteins with respect to the wild type. Most unreported changes in localization after hydroxyurea treatment were for proteins that displayed mixed localization patterns in controls, thus making it harder to visually detect the changes upon treatment. A similar approach was subsequently used to investigate the mechanisms of DNA repair in yeast [22]. Here, a collection of ∼5000 mutant yeast strains, high-throughput microscopy, and 470 features measured by image analysis and processed by SVMs, were used to identify 345 mutants that displayed increased number of fluorescently-tagged, DNA damage foci upon genetic or pharmacological manipulations. These results highlight the power of automated classification schemes, which enable scaling from visual classification of thousands of cells to automated scoring of millions of cells; and detection of changes in protein localization often difficult to identify for the human eye.

Quality control is particularly important when applying machine learning methods to images. The results of a pattern classification algorithm will heavily depend on how well the data in the test set can be assigned to the categories present in the training set. One way to provide quality control, is to introduce data of poor quality in the training set, and classify it as such. For instance, in the analysis of proteome dynamics in yeast, cells were initially segmented using the RFP channel to subsequently extract information about the GFP signal distribution in individual cells [21]. Segmentation errors can lead to a misrepresentation of the protein localization pattern. Similarly, dead cells can result in errors when characterizing the distribution of a protein, which will likely change when cells die. To overcome these issues, mis-segmented cells and dead cells were included in the training set and classified as such [21]. Mis-segmented and dead cells can be identified based on their morphology and fluorescence pattern, and thus, SVMs were created to detect and prevent further analysis of these low-quality data, thus illustrating the use of SVMs for data quality control.

SVMs can be used to reduce data dimensionality. Limiting the number of features used to train an SVM is important to avoid overfitting, particularly when the number of calculated features is too high. An SVM method for feature selection, originally developed to detect changes in protein localization upon drug treatment [23], has been applied to the study of how yeast protein localization patterns diverge for proteins with common ancestors [24]. In this work, 623 features were calculated initially, including morphological, moments, intensity, texture, and local structure metrics. To avoid overfitting, SVMs were used to identify the optimal hyperplane (using the fewest features) that could separate cells labeled for a specific protein from a reference set consisting of a mixture of cytoplasmic, nuclear, mitochondrial, and endoplasmic reticulum proteins from a public data set. This work illustrates an alternative application of SVMs to reduce data dimensionality, avoid overfitting, and accelerate data processing.

3.2. Cellular scale: lineage tracing

C. elegans undergoes a stereotypical development that allows consistent tracking of the ancestors and progeny (the lineage) of any given cell [25,26]. Several efforts have tried to automate lineage tracing from time-lapse, confocal microscopy images of embryonic development, with the goal of subsequently identifying phenotypes in genetic, RNA interference, or chemical screens [27,28]. The first step for lineage tracing is the automated segmentation and tracking of cells or nuclei. Segmentation in a living animal in 3D often results in errors, specially as the number and density of cells increase. Segmentation errors propagate into cell tracking, and can result in incorrect lineage annotations. One of the most frequent error types during the automated lineage tracing of *C. elegans* embryos is the classification of movements as cell divisions (movement-to-division), due to the presence of dividing cells in close proximity to migratory ones [29]. To correct for movement-to-division errors during lineage tracing of embryos expressing a fluorescent histone marker, an SVM was used to classify dividing and non-dividing cells [29]. The SVM used 82 features, including developmental time (more errors occur as cell density increases), cell age (which determines the likelihood of a cell to divide), relative morphology and spatial arrangement of parent and children cells, 3D nuclear shape (which changes during mitosis), histone content, etc. Ten-fold cross-validation was used to determine the efficacy of SVM-based detection of dividing cells, demonstrating an accuracy in the classification of cells as dividing of 88%, which represented a modest but significant 4% improvement over the uncorrected outcome of a lineage tracing tool. Of note, we recently reported accuracies of 97% in the detection of dividing cells in developing *Drosophila* embryos using only 14 features exclusively related to cell morphology (Fig. 3B) [16]. In this review, we demonstrate similar results using SVMs (Fig. 2). Our data raise the possibility that the use of an excessive number of features may lead to overfitting and compromise the results of other studies. However, we used a different, non-linear supervised learning method, logistic regression; and features related to cell shape, which are unavailable when using fluorescent nuclear labels and may have increased our ability to successfully identify dividing cells.

Simple classification schemes can be used to generate novel biological hypotheses. For instance, the mechanisms that generate, maintain or reduce heterogeneity within tissues are not well understood. An example of heterogeneity at the gene regulation level is found in intestinal cells in *C. elegans*, which display 100-fold differences in the expression of transgenes [30]. In a recent study, a GFP-tagged transgene was expressed in intestinal cells in *C. elegans*, and the fluorescence level of individual cells was recorded [30]. An SVM was built to classify cells into GFP positive or negative, using as features the fluorescence levels of four sets of sibling cells with a common ancestor ("lineal" prediction), or the fluorescence of neighbouring cells (spatial prediction). Ten-fold cross-validation demonstrated that lineal predictions were significantly more accurate than the baseline prediction of classifying all cells as bright, while spatial predictions were no better than the baseline. These results suggest that gene expression levels within a tissue are regulated when asymmetric cell divisions take place (lineal), and not through spatial cues.

3.3. Organism scale: characterizing developmental stage and gene expression patterns

In addition to their application to the analysis of subcellular and cellular dynamics, SVM can also be used to investigate whole-organism images. A popular application has been the automated identification of developmental stages in embryos, to annotate databases or create timelines of gene expression dynamics based on staining of fixed animals.

In *C. elegans*, SVMs have been used to identify different stages of development (embryo, larva, or adult), with the aim of detecting developmental defects in high-throughput screens [31]. After applying image segmentation to detect the animals from images, 13 morphological features were extracted for each animal, including size, perimeter, length of the symmetry axis, changes in width, and number of transitions between concave and convex in the shape of the animal. Objects were classified using a one-to-one approach, in which three SVMs were defined to distinguish between each of the possible stage pairs (embryo-larva, embryo-adult, and larva-adult). Ten-fold cross-validation was applied to evaluate the accuracy of the classifiers, using 100 images including approximately 100,000 animals that had been manually annotated. Each of the three classifiers produced correct labels ~90% of the time, with the classifier that distinguished between embryo and larva displaying the lowest classification accuracy due to the small size of the animals in early developmental stages, which made it difficult to extract informative features. In spite of this, the results of this work demonstrate that a small number of carefully selected features can yield highly predictive classifiers, particularly when the data are being partitioned into a small number of classes.

In contrast, highly resolved pattern recognition often requires more complex classification schemes. In *Drosophila*, 3724 images of in situ hybridizations -a technique used to visualize gene expression- from the Berkeley *Drosophila* Genome Project (BDGP) [32,33] were characterized using Gabor filters with different scales and orientations (Fig. 3C) [34]. SVMs were applied to score the stage of embryonic development in each image using a voting approach. 150 different SVMs were generated based on 30 random partitions of the data into a training set and a test set, with 5 different partition ratios ranging from 50% to 90% of the data assigned to the training set. Furthermore, 7 different algorithms were used to find the optimal value of **w** (Box 3), thus producing a total of 1050 different classifiers. Each one of the SVMs was a binary classifier trained to recognize whether an embryo was or not at a certain stage of development (out of 15 possible stages). The predicted developmental stage was the one that received the most votes from all of the classifiers, after weighing the contribution of each classifier based on their accuracy. Depending on whether the stage immediately before or the stage immediately after the assigned one received more votes, embryos were further sub-classified as being early or late into their stage of development, respectively. The method was applied to classify the developmental stages of 36,802 images from the FlyExpress database [35], which is based on two high-throughput studies [32,36]. Notably, 87% of the images were assigned to the proper development stage [34], thus demonstrating the ability of SVMs to perform the initial annotation of large image databases that can subsequently be curated by an expert user.

Supporting the use of SVMs for automated image database annotation, an SVM-based framework was developed to assign one or more developmental and anatomical terms from controlled vocabularies to individual BDGP images [37]. SIFT features were used to characterize image patches, and a binary SVM classifier was created for each term of the vocabulary, to determine if an image could be associated with that term as a function of the SIFT features. The framework was applied to 1438 images from the BDGP. Approximately one third of the 1438 test images received top vocabulary predictions different from the manual annotations in the BDGP. Upon re-inspection, some of the images were found to be mis-annotated. These results further highlight the utility of SVM-based approaches to automate the annotation of image databases, and to detect errors in manual annotations.

4. Conclusion

在图像的处理上，使得前所未有的空间和时间上的细节的处理加以实现。基于这样的计算速度和精确度以及方案实现的便捷性，SVM是在大规模数量的复杂的显微图像的情况下提取一定数量的理想工具。

Super-resolution and light-sheet microscopies have enabled unprecedented spatial and temporal detail in the imaging of animal development and repair. Based on their speed, accuracy, and relative ease of implementation, SVMs are ideal tools to extract quantitative information at multiple scales from complex microscopy images. Only initial efforts have been taken to apply SVMs to the analysis of super-resolution and light-sheet microscopy data [7,8,38]. However, we expect that the complexity of images acquired from living animals, combined with the ever-decreasing cost of computational power, will lead to the popularization of SVMs (and other machine learning methods) for the analysis of in vivo microscopy.

我们希望从活体动物得到的复杂的成像能使得SVMs能更广泛地被接受于活体成像的分析

Transparency document

The Transparency document associated with this article can be found, in online version.

Acknowledgments

Work in our laboratory is supported by grants from the Canada Foundation for Innovation (#30279), the Natural Sciences and Engineering Research Council of Canada (#418438-13), the Translational Biology and Engineering Program of the Ted Rogers Centre for Heart Research, the Canada First Research Excellence Fund for the Medicine by Design Initiative of the University of Toronto, and an Early Researcher Award to RFG from the Ontario Ministry of Research and Innovation (#ER14-10-170). RFG is the Canada Research Chair (Tier II) in Quantitative Cell Biology and Morphogenesis.

References

[1] A.M. Sydor, K.J. Czymmek, E.M. Puchner, V. Mennella, Super-resolution microscopy: from single molecules to supramolecular assemblies, Trends Cell Biol. 25 (2015) 730–748.

[2] R.M. Power, J. Huisken, A guide to light-sheet fluorescence microscopy for multiscale imaging, Nat. Methods 14 (2017) 360–373.

[3] P.J. Keller, A.D. Schmidt, J. Wittbrodt, E.H. Stelzer, Reconstruction of zebrafish early embryonic development by scanned light sheet microscopy, Science 322 (2008) 1065–1069.

[4] R. Tomer, K. Khairy, F. Amat, P.J. Keller, Quantitative high-speed imaging of entire developing embryos with simultaneous multiview light-sheet microscopy, Nat. Methods 9 (2012) 755–763.

[5] H. Zhan, R. Stanciauskas, C. Stigloher, K.K. Dizon, M. Jospin, J.L. Bessereau, F. Pinaud, In vivo single-molecule imaging identifies altered dynamics of calcium channels in dystrophin-mutant *C. elegans*, Nat. Commun. 5 (2014) 4974.

[6] S. Schnorrenberg, T. Grotjohann, G. Vorbruggen, A. Herzig, S.W. Hell, S. Jakobs, In vivo super-resolution RESOLFT microscopy of *Drosophila melanogaster*, elife 5 (2016).

[7] P. Strnad, S. Gunther, J. Reichmann, U. Krzic, B. Balazs, G. de Medeiros, N. Norlin, T. Hiiragi, L. Hufnagel, J. Ellenberg, Inverted light-sheet microscope for imaging mouse pre-implantation development, Nat. Methods 13 (2016) 139–142.

[8] F. Amat, W. Lemon, D.P. Mossing, K. McDole, Y. Wan, K. Branson, E.W. Myers, P.J. Keller, Fast, accurate reconstruction of cell lineages from large-scale fluorescence microscopy data, Nat. Methods 11 (2014) 951–958.

[9] B.E. Boser, I.M. Guyon, V.N. Vapnik, A training algorithm for optimal margin classifiers, Fifth Annual Workshop on Computational Learning Theory, ACM, Pittsburgh, PA, 1992, pp. 144–152.

[10] C. Cortes, V.N. Vapnik, Support-vector networks, Mach. Learn. 20 (1995) 273–297.

[11] W.S. Noble, What is a support vector machine? Nat. Biotechnol. 24 (2006) 1565–1567.

[12] A. Martin, M. Gelbart, R. Fernandez-Gonzalez, M. Kaschube, E. Wieschaus, Integration of contractile forces during tissue invagination, J. Cell Biol. 188 (2010) 735–749.

[13] S. Beucher, The watershed transformation applied to image segmentation, Scanning Microsc. Suppl. 6 (1992) 299–314.

[14] R. Fernandez-Gonzalez, J.A. Zallen, Oscillatory behaviors and hierarchical assembly of contractile structures in intercalating cells, Phys. Biol. 8 (2011) 045005.

[15] C.Y. Leung, R. Fernandez-Gonzalez, Quantitative image analysis of cell behavior and molecular dynamics during tissue morphogenesis, Methods Mol. Biol. 1189 (2015) 99–113.

[16] M.F. Wang, M.V. Hunter, G. Wang, C. McFaul, C.M. Yip, R. Fernandez-Gonzalez, Automated cell tracking identifies mechanically oriented cell divisions during *Drosophila* axis elongation, Development 144 (2017) 1350–1361.

[17] W.K. Huh, J.V. Falvo, L.C. Gerke, A.S. Carroll, R.W. Howson, J.S. Weissman, E.K. O'Shea, Global analysis of protein localization in budding yeast, Nature 425 (2003) 686–691.

[18] S.C. Chen, T. Zhao, G.J. Gordon, R.F. Murphy, Automated image analysis of protein localization in budding yeast, Bioinformatics 23 (2007) i66–71.

[19] K. Huang, R.F. Murphy, From quantitative microscopy to automated image understanding, J. Biomed. Opt. 9 (2004) 893–912.

[20] C.W. Hsu, C.J. Lin, A comparison of methods for multiclass support vector machines, IEEE Trans. Neural Netw. 13 (2002) 415–425.

[21] Y.T. Chong, J.L. Koh, H. Friesen, S.K. Duffy, M.J. Cox, A. Moses, J. Moffat, C. Boone, B.J. Andrews, Yeast proteome dynamics from single cell imaging and automated analysis, Cell 161 (2015) 1413–1424.

[22] E.B. Styles, K.J. Founk, L.A. Zamparo, T.L. Sing, D. Altintas, C. Ribeyre, V. Ribaud, J. Rougemont, D. Mayhew, M. Costanzo, M. Usaj, A.J. Verster, E.N. Koch, D. Novarina, M. Graf, B. Luke, M. Muzi-Falconi, C.L. Myers, R.D. Mitra, D. Shore, G.W. Brown, Z. Zhang, C. Boone, B.J. Andrews, Exploring quantitative yeast phenomics with single-cell analysis of DNA damage foci, Cell Syst. 3 (2016) 264–277 (e210).

[23] L.H. Loo, L.F. Wu, S.J. Altschuler, Image-based multivariate profiling of drug responses from single cells, Nat. Methods 4 (2007) 445–453.

[24] L.H. Loo, D. Laksameethanasan, Y.L. Tung, Quantitative protein localization signatures reveal an association between spatial and functional divergences of proteins, PLoS Comput. Biol. 10 (2014) e1003504.

[25] J.E. Sulston, H.R. Horvitz, Post-embryonic cell lineages of the nematode, *Caenorhabditis elegans*, Dev. Biol. 56 (1977) 110–156.

[26] J.E. Sulston, E. Schierenberg, J.G. White, J.N. Thomson, The embryonic cell lineage of the nematode *Caenorhabditis elegans*, Dev. Biol. 100 (1983) 64–119.

[27] Z. Bao, J.I. Murray, T. Boyle, S.L. Ooi, M.J. Sandel, R.H. Waterston, Automated cell lineage tracing in *Caenorhabditis elegans*, Proc. Natl. Acad. Sci. U. S. A. 103 (2006) 2707–2712.

[28] F. Long, H. Peng, X. Liu, S.K. Kim, E. Myers, A 3D digital atlas of *C. elegans* and its application to single-cell analyses, Nat. Methods 6 (2009) 667–672.

[29] Z. Aydin, J.I. Murray, R.H. Waterston, W.S. Noble, Using machine learning to speed up manual image annotation: application to a 3D imaging protocol for measuring single cell gene expression in the developing *C. elegans* embryo, BMC Bioinf. 11 (2010) 84.

[30] H.H. Le, M. Looney, B. Strauss, M. Bloodgood, A.M. Jose, Tissue homogeneity requires inhibition of unequal gene silencing during development, J. Cell Biol. 214 (2016) 319–331.

[31] A.G. White, B. Lees, H.L. Kao, P.G. Cipriani, E. Munarriz, A.B. Paaby, K. Erickson, S. Guzman, K. Rattanakorn, E. Sontag, D. Geiger, K.C. Gunsalus, F. Piano, DevStaR: high-throughput quantification of *C. elegans* developmental stages, IEEE Trans. Med. Imaging 32 (2013) 1791–1803.

[32] P. Tomancak, A. Beaton, R. Weiszmann, E. Kwan, S. Shu, S.E. Lewis, S. Richards, M. Ashburner, V. Hartenstein, S.E. Celniker, G.M. Rubin, Systematic determination of patterns of gene expression during *Drosophila* embryogenesis, Genome Biol. 3 (2002) (RESEARCH0088).

[33] P. Tomancak, B.P. Berman, A. Beaton, R. Weiszmann, E. Kwan, V. Hartenstein, S.E. Celniker, G.M. Rubin, Global analysis of patterns of gene expression during *Drosophila* embryogenesis, Genome Biol. 8 (2007) R145.

[34] L. Yuan, C. Pan, S. Ji, M. McCutchan, Z.H. Zhou, S.J. Newfeld, S. Kumar, J. Ye, Automated annotation of developmental stages of *Drosophila* embryos in images containing spatial patterns of expression, Bioinformatics 30 (2014) 266–273.

[35] S. Kumar, C. Konikoff, B. Van Emden, C. Busick, K.T. Davis, S. Ji, L.W. Wu, H. Ramos, T. Brody, S. Panchanathan, J. Ye, T.L. Karr, K. Gerold, M. McCutchan, S.J. Newfeld, FlyExpress: visual mining of spatiotemporal patterns for genes and publications in *Drosophila* embryogenesis, Bioinformatics 27 (2011) 3319–3320.

[36] E. Lecuyer, H. Yoshida, N. Parthasarathy, C. Alm, T. Babak, T. Cerovina, T.R. Hughes, P. Tomancak, H.M. Krause, Global analysis of mRNA localization reveals a prominent role in organizing cellular architecture and function, Cell 131 (2007) 174–187.

[37] Y.X. Li, S. Ji, S. Kumar, J. Ye, Z.H. Zhou, *Drosophila* gene expression pattern annotation through multi-instance multi-label learning, IEEE/ACM Trans. Comput. Biol. Bioinform. 9 (2012) 98–112.

[38] S. Maji, M.P. Bruchez, Inferring biological structures from super-resolution single molecule images using generative models, PLoS One 7 (2012) e36973.

[39] T. Joachims, Making large-scale support vector machine learning practical, in: B. Scholkopf, C. Burges, A. Smola (Eds.), Advances in kernel methods - support vector learning, MIT Press, Place Published, 1999, pp. 169–184.

[40] C.C. Chang, C.J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (2011).

[41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos,

D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine Learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[42] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I.H. Witten, The WEKA data mining software: an update, ACM SIGKDD Explor. Newsl. 11 (2009) 10–18.

[43] W.H. Walton, Ferets statistical diameter as a measure of particle size, Nature 162 (1948) 329–330.

[44] T.P. Weldon, W.E. Higgins, D.F. Dunn, Gabor filter design for multiple texture segmentation, Opt. Eng. 35 (1996) 2852–2863.

[45] A.K. Jain, N.K. Ratha, S. Lakshmanan, Object detection using Gabor filters, Pattern Recogn. 30 (1997) 295–309.

[46] R.M. Haralick, K. Shanmugam, I.H. Dinstein, Textural features for image classification, IEEE Trans. Syst. Man Cybern. (1973) 610–621.

[47] M. Hu, Visual pattern recognition by moment invariants, IRE Trans. Inf. Theory 8 (1962) (179- &).

[48] D.G. Lowe, Object recognition from local scale-invariant features, computer vision, 1999, The Proceedings of the Seventh IEEE International Conference on, Ieee, 1999, pp. 1150–1157.

[49] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2004) 91–110.

[50] M.R. Teague, Image analysis via the general theory of moments, J. Opt. Soc. Am. 70 (1980) 920–930.

[51] F. Zernike, Inflection theory of the cutting method and its improved form, the phase contrast method, Physica 1 (1934) 689–704.

[52] I. Daubechies, Orthonormal bases of compactly supported wavelets, Commun. Pure Appl. Math. 41 (1988) 909–996.

目前对于高精度成像的显微镜技术和来自生物样本的无光毒性成像使得获取大容量的经过长期观察得到的生命体的发展和过程得以实现。高分辨率的图像和光切成像有着史无前例的复杂性。同时，大数据的生成对于显微镜技术是一项巨大的挑战。尤其是基于加注释于图像的传统分析方法，捕捉储存于图像的信息时未免有些捉襟见肘。人工处理是不可行的，因为其复杂性和耗时性并且因为被限制的数据处理方法很容易产生偏差。然而，机器学习可以理想地适用于大容量的生物数据的考察。机器学习，结合于自动化的图像分析，可以被应用于图像的分割从而探测细胞以及亚细胞尺度下的表现型或者对整个生命体的发展阶段进行分类。

Feret用于形状测量，它对两个物体之间的距离进行了量化，并且会探测物体向。Feret是根据任何给定的物体的不同方向计算的。随着形状的不规则性，Feret测得的最大直径和最小的直径将会发生变化。因此，最大到最小的直径的比值可以用来量化物体的伸长情况，例如测量细胞的排列。

Gabor滤过器可以被用来检测和描述细胞的定向突出部分，或者将细胞分类为突出的细胞。

傅里叶变换将图像分解成不同的频率成分，可以用来描述细胞形态。