A deep learning and novelty detection framework for rapid phenotyping in high-content screening

Christoph Sommer†, Rudolf Hoefler†, Matthias Samwer, and Daniel W. Gerlich*
Institute of Molecular Biotechnology of the Austrian Academy of Sciences (IMBA), Vienna Biocenter (VBC), 1030 Vienna, Austria

ABSTRACT **Supervised machine learning is a powerful and widely used method for analyzing high-content screening data. Despite its accuracy, efficiency, and versatility, supervised machine learning has drawbacks, most notably its dependence on a priori knowledge of expected phenotypes and time-consuming classifier training. We provide a solution to these limitations with *CellCognition Explorer*, a generic novelty detection and deep learning framework. Application to several large-scale screening data sets on nuclear and mitotic cell morphologies demonstrates that *CellCognition Explorer* enables discovery of rare phenotypes without user training, which has broad implications for improved assay development in high-content screening.**

Monitoring Editor
Charles Boone
University of Toronto

Received: May 30, 2017
Revised: Aug 31, 2017
Accepted: Sep 18, 2017

INTRODUCTION

Advances in microscope automation have facilitated the systematic study of cellular phenotypes resulting from genetic or chemical perturbations. Prevailing image analysis pipelines rely on supervised machine learning to classify cellular phenotypes based on user-defined collections of statistical image features (Boland and Murphy, 2001; Carpenter *et al.*, 2006; Neumann *et al.*, 2006; Bakal *et al.*, 2007; Jones *et al.*, 2009; Ramo *et al.*, 2009; Held *et al.*, 2010; Misselwitz *et al.*, 2010; Sommer and Gerlich, 2013; Boutros *et al.*, 2015; Mattiazzi Usaj *et al.*, 2016). This approach has revealed new gene functions in various RNA interference (RNAi)-based screens (e.g., Goshima *et al.*, 2007; Neumann *et al.*, 2010; Schmitz *et al.*, 2010; Gudjonsson *et al.*, 2012; Sommer and Gerlich, 2013; Liberali *et al.*, 2014; Boutros *et al.*, 2015; Cuylen *et al.*, 2016). While supervised machine learning is in principle broadly applicable, it requires extensive user interaction for the development of new biological assays.

A major limitation of supervised machine learning is the requirement for classifier training for each new assay or variation in the experimental conditions. The classifier training requires representative images for all expected phenotype classes. As possible phenotype morphologies are not always completely known a priori, screening assay development often involves extensive and time-consuming pilot screens that are inspected visually for manual annotation of phenotype classes (Conrad and Gerlich, 2010). This process can be facilitated by interactive learning (Jones *et al.*, 2009), yet rare phenotype classes might not be represented in pilot screens. Moreover, current high-content screening analysis software relies on user-curated collections of feature extraction algorithms, which often require specific software adaptations when establishing new cell biological assays. Developing classifiers for new biological assays has hence remained a major bottleneck in high-content screening.

These limitations might be overcome by unsupervised learning methods, which estimate phenotypic content based on intrinsic data structure. However, high cell-to-cell variability and experimental noise often preclude sensitive and reliable detection of low-penetrance phenotypes. The accuracy of unsupervised phenotype detection can be improved by object features beyond the single-cell level, such as the temporal context (Zhong *et al.*, 2012; Failmezger *et al.*, 2013) or the cell population context (Rajaram *et al.*, 2012; Liberali *et al.*, 2014), but such information is not always applicable for a given biological assay. The detection of unknown phenotypes might be facilitated by novelty detection methods (Yin *et al.*, 2008, 2013; Manning and Shamir, 2014), yet the performance of such methods in genome-wide screening has not yet been tested.

This article was published online ahead of print in MBoC in Press (http://www.molbiolcell.org/cgi/doi/10.1091/mbc.E17-05-0333) on September 27, 2017.

†These authors contributed equally to this work.

Author contributions: D.W.G. and C.S. conceived the project; D.W.G. and M.S. designed experiments: M.S. performed experiments; C.S. analyzed data; R.H. and C.S. implemented software (*CellCognition Explorer* main program and *CellCognition Deep Learning Module*, respectively); D.W.G. and C.S. wrote the paper.

The authors declare no competing financial interests.

*Address correspondence to: Daniel Gerlich (daniel.gerlich@imba.oeaw.ac.at).

Abbreviations used: GPU, graphics processing unit; H2B, histone 2B; KDE, kernel density estimation; MD, Mahalanobis distance; OC-SVM, one-class support vector machine; RNAi, RNA interference; siRNA, small interfering RNA; SVM, support vector machine.

The dependence of machine learning approaches on user-curated image features might be overcome by feature self-learning from the data. This can be achieved by deep learning methodology (LeCun *et al.*, 2015), which has demonstrated impressive performance in various domains, such as face recognition (Taigman *et al.*, 2014) and speech recognition (Sainath *et al.*, 2013). Recent analyses of protein localization in budding yeast by deep learning indicates a high potential in bioimaging (Kraus *et al.*, 2017; Parnamaa and Parts, 2017), but applicability to genome-scale human cell screening data has not yet been explored and an integration of novelty detection with deep learning is not available in community-standard software for high-content screening.

We here present the novelty detection and deep learning software *CellCognition Explorer*. The software yields phenotype scores based on deviation of cell morphologies from negative control images. We demonstrate that *CellCognition Explorer* enables sensitive and accurate cellular phenotype detection in genome-scale screening data without the need for extensive user interaction for data annotation. In addition, deep learning of image-derived features overcomes the dependence on user-curated feature analysis collections and accurate cell segmentation outlines. Hence, *CellCognition Explorer* greatly facilitates rapid screening assay development even when cellular phenotypes are not known a priori.

RESULTS

CellCognition Explorer software

We have developed *CellCognition Explorer*, a machine learning framework for the detection of abnormal cell morphologies without prior user training. Two core technologies enable the unsupervised classification task. Deep learning methods infer numerical descriptors of individual cell objects directly from the raw image pixel data, thus circumventing feature engineering and software adaptations for new biological assays. Novelty detection methods then learn a statistical model of the natural phenotype variation within the negative control cell population. This yields an accurate classification boundary for subsequent detection of any morphological deviations in large-scale screening data—even for phenotypes that are not known a priori.

CellCognition Explorer provides a pipeline for integrated data analysis from raw images to phenotype scores (Figure 1). The software package consists of two programs: The main *CellCognition Explorer* program provides interactive data visualization tools and the possibility of performing versatile analysis workflows using novelty detection methodology, as well as conventional supervised learning methods. It is controlled by a simple graphical user interface (Supplemental Figure S1) that works on all major computer operating systems. *CellCognition Deep Learning Module* is a separate program for graphics processing unit (GPU)-accelerated high-performance computing of deep learning features (Supplemental Figure S2). The implementation as two separate programs provides optimal flexibility for installation of the interactive data exploration and workflow design tool, while enabling the efficient computation of deep learning features by dedicated GPU hardware. Both programs are controlled by graphical user interfaces and distributed as open source software embedded within the *CellCognition* platform (Held *et al.*, 2010; http://software.cellcognition-project.org/explorer/). Interoperability with widely used bioimage software packages, including ImageJ/Fiji (Abramoff *et al.*, 2004; Schindelin *et al.*, 2012), CellProfiler (Carpenter *et al.*, 2006), and Bioconductor (Gentleman *et al.*, 2004), is enabled via the standardized file format *cellH5* (Sommer *et al.*, 2013).

Cell phenotyping by novelty detection

The novelty detection algorithms implemented in *CellCognition Explorer* are designed to learn intrinsic cell-to-cell variability in an untreated negative control cell population autonomously, which sensitizes the classifier to perturbation-induced phenotypes. Abnormal cell phenotypes are then scored either based on the weighted cell object distance in feature space relative to the mean and covariance of a control cell population (Mahalanobis distance,

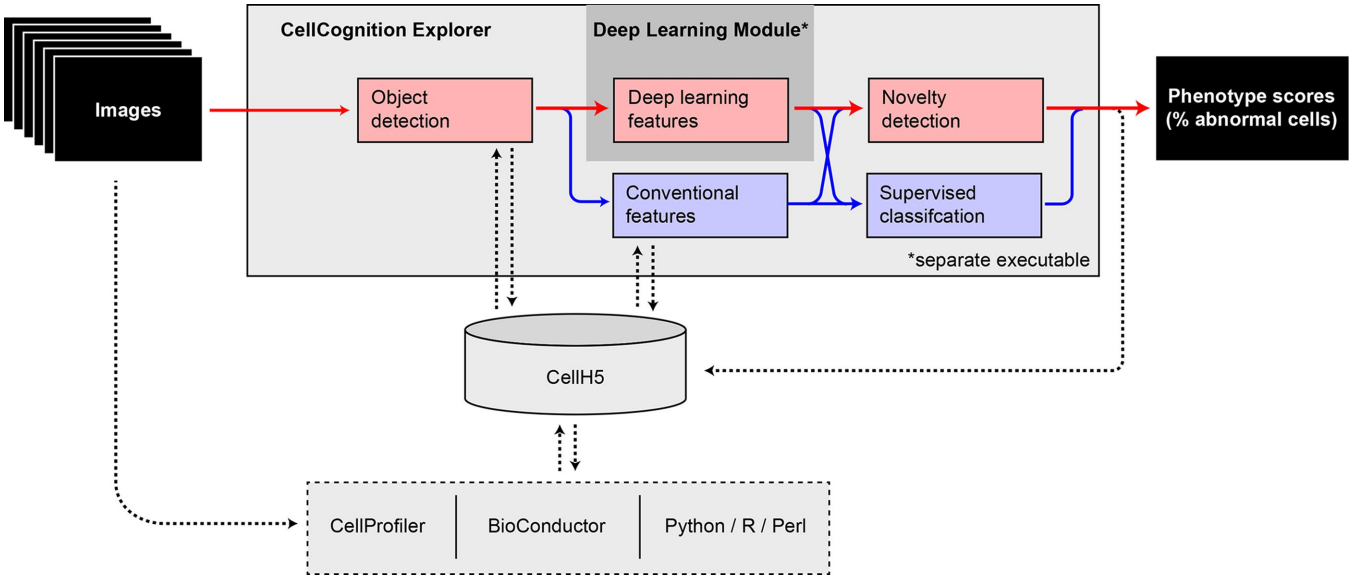

FIGURE 1: Data analysis workflows with *CellCognition Explorer*. Red boxes and arrows indicate the standard workflow with object detection, deep learning, and novelty detection as described in this paper. Blue boxes indicate previously described methods (Held *et al.*, 2010) that can be combined with the new functionality (blue arrows) or with other high-content screening software via CellH5 (Sommer *et al.*, 2013; dashed lines). Deep learning features are computed through a separate program (owing to specific high-performance computing hardware requirements). Workflow integration is achieved through CellH5 data exchange (Sommer *et al.*, 2013).

MD; Pimentel *et al.*, 2014), based on their likelihood after the multivariate probability density of control conditions is estimated (kernel density estimation, KDE; Pimentel *et al.*, 2014), or by fitting a nonlinear hyperplane to control cell objects (one-class support vector machine; Scholkopf *et al.*, 2001) (see *Materials and Methods*). These methods can score and classify cell objects based on conventional precomputed numerical features (e.g., Boland and Murphy, 2001; Murphy *et al.*, 2003; Carpenter *et al.*, 2006; Held *et al.*, 2010) or based on learned representations from the original image pixel data when combined with deep learning (Vincent *et al.*, 2010; Durr and Sick, 2016; Kraus *et al.*, 2016).

To test the performance of the novelty detection methods, we generated a data set representing the full spectrum of morphological phenotypes for a chromatin marker. HeLa cells stably expressing fluorescent histone 2B fused to mCherry (H2B–mCherry) were subjected to RNAi depletion of 1214 genes previously identified as important for mitosis from a genomewide screen (Neumann *et al.*, 2010), using individual transfection of two different small interfering RNAs (siRNAs) per target gene, followed by live-cell imaging on an automated epifluorescence microscope. To establish a reference annotation, we detected cell objects based on local adaptive thresholding and watershed segmentation and calculated 239 conventional numerical features describing texture and shape (Held *et al.*, 2010). We then randomly subsampled 10,000 cell objects and manually annotated their "ground truth" phenotypes by classification into nine different morphology classes: normal interphase nuclei and various different outlier morphologies, representing mitotic stages, dead cells, and abnormal nuclear shapes (Figure 2a,

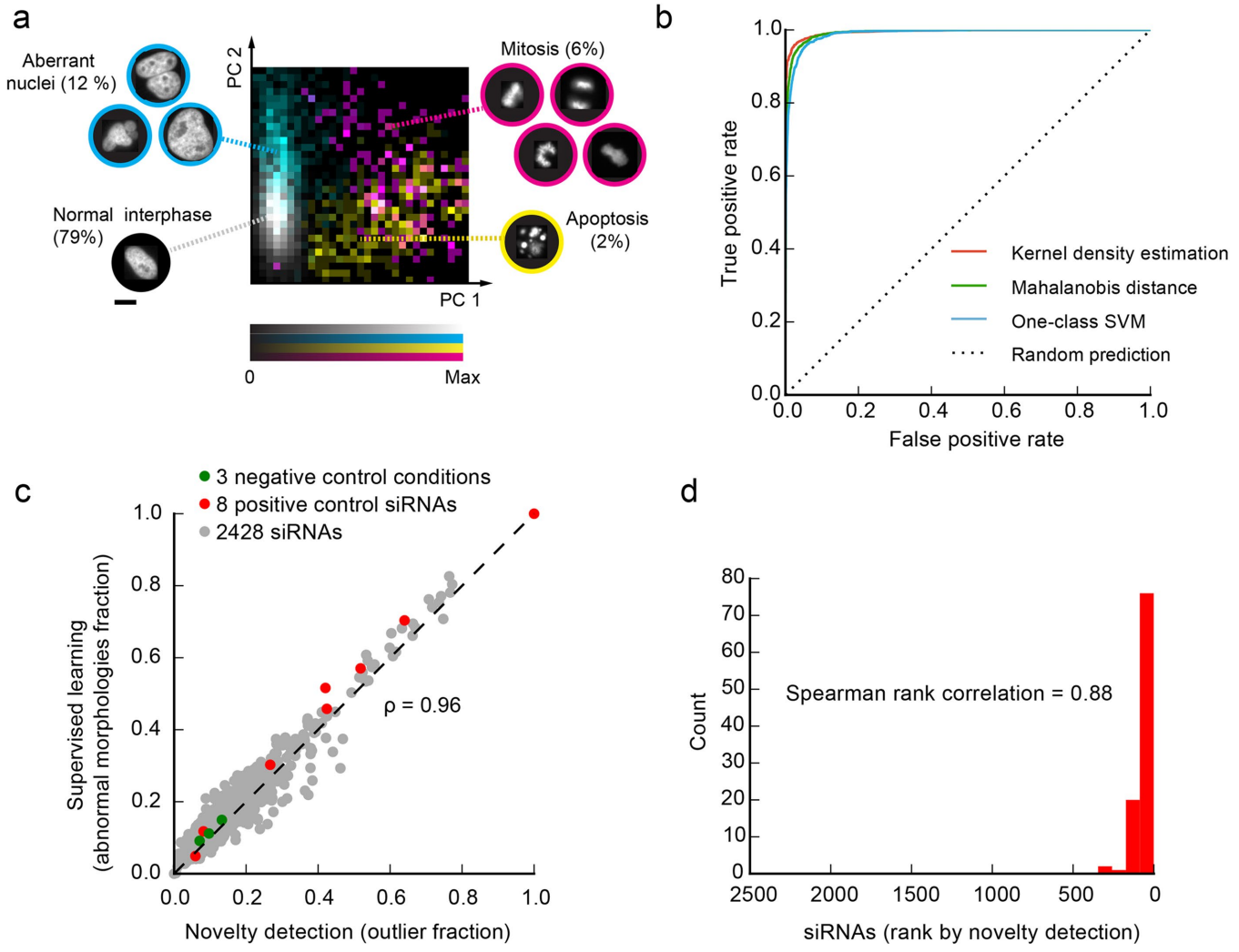

FIGURE 2: Novelty detection with *CellCognition Explorer*. (a) Phenotype distribution in principal component subspace of 10,000 manually annotated HeLa cells stably expressing H2B-mCherry. The cell images were randomly selected from 2428 RNAi experiments to establish a representative collection of all cellular phenotypes. The nine phenotype classes illustrate the most common nuclear and mitotic chromatin morphologies. Colors indicate phenotype groups as illustrated by examples. (b) Performance of different novelty detection methods. True positive and false positive rates (receiver operating characteristic curve) were calculated for the indicated methods based on the data shown in a. (c) Comparison of phenotype scoring by conventional supervised learning (Held *et al.*, 2010) and novelty detection. Each dot represents one of 2428 different siRNAs; negative controls: no-targeting siRNAs; positive controls: siRNAs causing strong mitotic phenotypes. Phenotype scores are defined as the fraction of HeLa-H2B-mCherry cells deviating from normal interphase morphology. For detailed analysis results, see Supplemental Table S2. (d) Phenotype scoring of 2428 siRNAs as in a by novelty detection using *CellCognition Explorer*. Red bars indicate the distribution of the top 100 ranked siRNA hits identified by conventional supervised learning; see c.

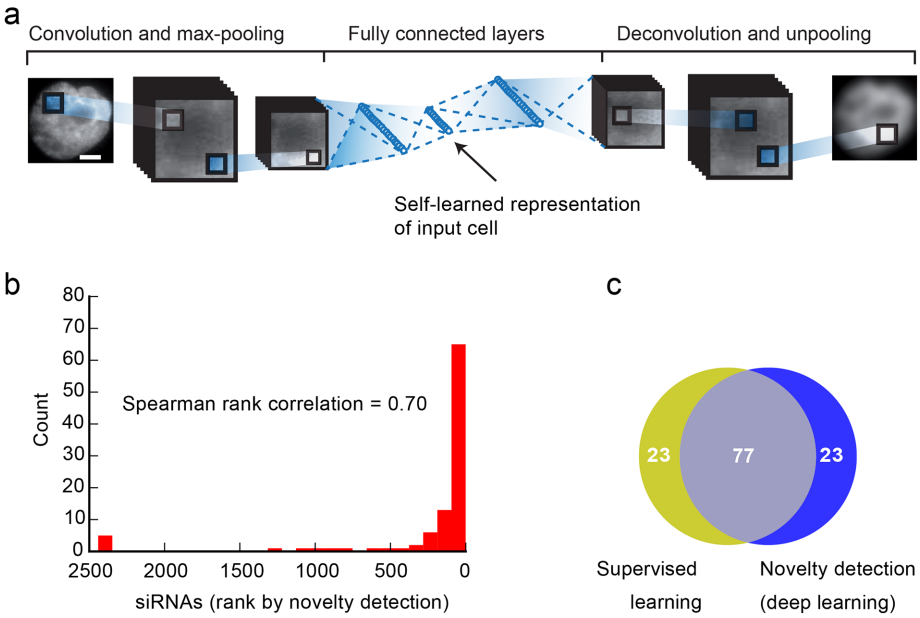

FIGURE 3: Self-learning of cell object features with *CellCognition Deep Learning Module*. (a) Schematic illustration of deep learning using an AE with convolutional, pooling, and fully connected layers. (b) Phenotype scoring of 2428 siRNAs (see Figure 1a) by novelty detection and deep learning using *CellCognition Explorer*. Red bars indicate the distribution of the top 100 ranked siRNA hits identified by conventional supervised learning as in Held *et al.* (2010). (c) Comparison of the top 100 screening hits determined either by novelty detection and deep learning of object features (blue) or by supervised learning and conventional features (yellow) for 2428 siRNAs as in a and b. For comparison of novelty detection with conventional and deep learning features, see Supplemental Figure S4a. Scale bars, 10 µm.

Supplemental Table S1, full data set available at http://software.cellcognition-project.org/explorer/). In the principal component subfeature space (Pimentel *et al.*, 2014), interphase nuclei appeared as a single compact cluster, whereas the different outlier phenotype groups scattered broadly in different regions.

All three novelty detection methods implemented in *CellCognition Explorer* accurately classified normal interphase nuclei as inlier objects and other morphologies as outliers, consistent with phenotype scoring with supervised analysis (Figure 2b). To extend the performance tests to the full data set, we next quantified the abundance of abnormal cell phenotypes in each of the 2428 RNAi conditions. The fraction of cells with outlier morphologies calculated with novelty detection methods consistently matched the reference state-of-the-art analysis using supervised learning by support vector machines (Held *et al.*, 2010) (SVMs; Figure 2, c and d, and Supplemental Figure S3). Thus, novelty detection with *CellCognition Explorer* accurately identifies abnormal cell morphology phenotypes without the need for extensive data annotation.

Deep learning of cell features

The *CellCognition Deep Learning* Module automatically extracts numerical feature sets that adjust to specific cell morphology markers used in an assay. This is achieved by a convolutional autoencoder, a multilayered artificial neural network (Hinton and Salakhutdinov, 2006) that learns a representation (encoding) for a collection of images. This method requires only center coordinates of cell objects as an input and is thus independent of the accurate object segmentation contours that are normally necessary for conventional user-curated feature sets to calculate shape features. The features derived by deep learning serve as an input for the novelty detection method (see above) and can also be used for conventional supervised machine learning. The implementation of different analysis pipelines combining supervised and unsupervised methods is facilitated by the interactive visualization platform for cell objects (Figure 1).

We evaluated the accuracy of phenotype scoring based on deep learning–derived features using the reference data as described above. We trained deep autoencoder neural networks to reduce the high-dimensional image-pixel data to low-dimensional compressed code (Figure 3a; for details see Supplemental Tables S5–S7). The parameters of such networks are iteratively adjusted by minimizing the discrepancy between the original data and its reconstruction based on the compressed code. The learned features then serve as an input for novelty detection as described above. The top-scoring RNAi phenotypes obtained by this method matched well the reference scoring by supervised learning (Figure 3, b and c). The total accuracy achieved by deep learning features was only slightly lower than that of classical feature collection (compare Supplemental Figures S4b and S4c), which has been highly optimized for the specific chromatin morphology assay. Thus, unsupervised deep learning can derive informative features for fully automated phenotype scoring by novelty detection, thereby overcoming dependence on manually curated feature sets.

Application to genomewide screening

To test the applicability of *CellCognition Explorer* to high-throughput screening, we aimed to recompute phenotype scores for published primary image data of a genomewide RNAi screen for mitotic regulators (Neumann *et al.*, 2010). The original phenotype scoring of these screening data was based on classification using supervised learning (Neumann *et al.*, 2010), in which the development of a reliable classifier required a multimonth phase of iterative pilot screening and visual inspection for classifier training (Neumann *et al.*, 2006, 2010). The analysis of millions of images during pilot screening had ultimately led to the discovery of many previously unknown phenotypes (Figure 4a). Identifying all abnormal cell morphologies prior to classifier training would abrogate the need to visually inspect the full data set, thereby solving a major bottleneck in data analysis.

We used the *CellCognition Deep Learning Module* to extract self-learned features for phenotype scoring based on novelty detection. The top-ranked siRNAs according to phenotype penetrance were very similar to the originally published hit list (Figure 4b), even though information on abnormal phenotype morphologies from pilot screening was not taken into account. Hit lists generated by our new methods are hence highly enriched for abnormal cell morphologies, which greatly facilitates further phenotype analysis by visual inspection, supervised classification, or unsupervised clustering.

To probe the versatility of our methods, we next reanalyzed primary image data from another large-scale RNAi screen for DNA damage repair regulators (Gudjonsson *et al.*, 2012). The primary assay probed the regulation of DNA damage repair protein

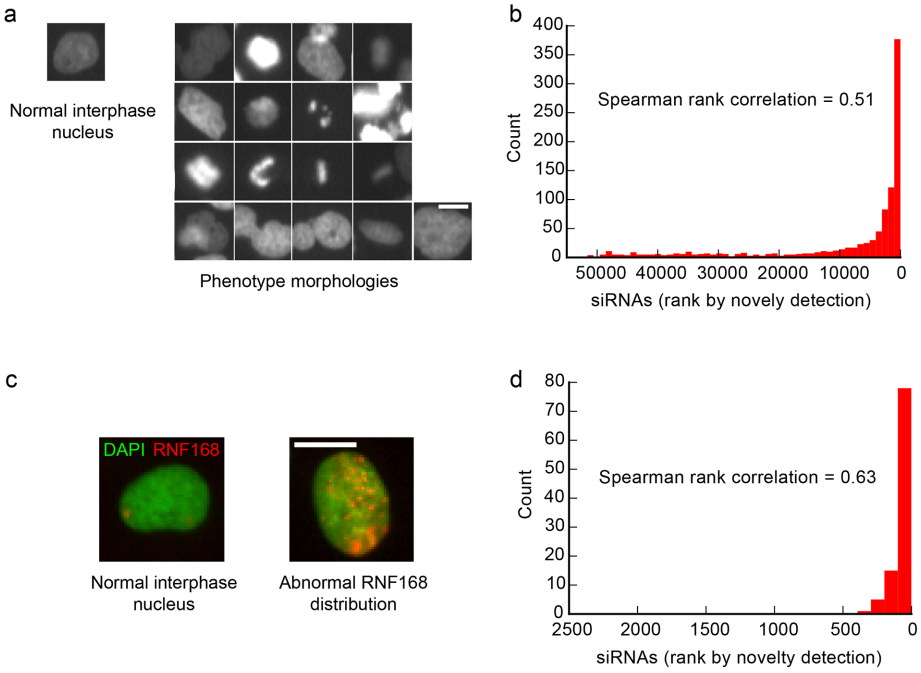

FIGURE 4: Application of *CellCognition Explorer* to high-throughput RNAi screening. (a) Genomewide RNAi screen for mitotic regulators based on live-cell microscopy of HeLa cells expressing H2B-mCherry (Neumann *et al.*, 2010). Images show representative examples for a normal interphase morphology and various phenotype morphology classes as defined in Neumann *et al.* (2010). (b) Phenotype scoring of 51,748 siRNAs by novelty detection and deep learning with *CellCognition Explorer*, using the primary image data of Neumann *et al.* (2010). Red bars indicate the distribution of the top 1000 ranked siRNA hits identified by conventional supervised learning as in Neumann *et al.* (2010). For details on analysis, see Supplemental Table S3. (c) Large-scale RNAi screen for DNA damage repair regulators (Gudjonsson *et al.*, 2012). GFP-tagged RNF168 (red) visualizes spontaneous DNA damage in U2OS cells by accumulation in bright nuclear foci, which increase in number and intensity upon perturbed DNA damage repair (abnormal RNF168 distribution in siTRIP12 RNAi cell). DNA (green) is counterstained by DAPI. (d) Phenotype scoring of 2423 siRNAs by novelty detection and deep learning with *CellCognition Explorer*, using the primary image data of Gudjonsson *et al.* (2012). Red bars indicate the distribution of the top 100 ranked siRNA hits as identified in Gudjonsson *et al.* (2012). For details on analysis, see Supplemental Table S4. Scale bars, 10 µm.

RNF168, which accumulates in bright nuclear foci (Figure 4c). The distribution of top-ranked siRNA hits derived from deep-learned features and novelty detection was very similar to that in the original conventional supervised learning analysis (Figure 4d). Hence, *CellCognition Explorer* yields reliable phenotype scores in large-scale screening data without prior user definition of aberrant phenotype morphologies.

To ensure broad applicability in biological laboratories, we designed our software to achieve high-throughput data processing independent of expensive computing infrastructure. We therefore implemented the computationally intense deep learning module as separate software that runs on standard consumer-grade graphics hardware on a desktop computer (see *Materials and Methods*).

With this, we completely processed both large-scale screening data sets. The genome-wide RNAi screen (Figure 4, a and b) was processed in less than 34 h in total. The autoencoder training on 200,000 cell objects from negative control conditions took 2 h. After training, 19 million cells were classified with the novelty detection procedure (including feature generation) in 32 h, hence with a throughput of 593,750 cells/h. Thus, deep learning and novelty detection with *CellCognition Explorer* provide a versatile and powerful solution for rapid phenotype scoring in large-scale high-content screening.

DISCUSSION

CellCognition Explorer bypasses the need for extensive pilot screening to capture all possible phenotype morphologies and thereby solves a major bottleneck of conventional supervised approaches. *CellCognition Explorer* yields a list of top-ranked screening targets directly from large-scale screening data, which can be used for mechanistic follow-up studies or further subclassified using conventional supervised methods.

By avoiding human bias arising from supervised classifier training (Zhong *et al.*, 2012), novelty detection with *CellCognition Explorer* helps to improve the consistency between different screens. Furthermore, the independence from manually curated feature sets facilitates the development of new cell biological assays. The inference of image features by deep learning does not take the image segmentation contours into account and therefore has the potential to facilitate the analysis of cellular markers that are difficult to segment.

We here demonstrate how *CellCognition Explorer* can be used to detect novel phenotypes in nuclear morphology screening data. *CellCognition Explorer* supports processing of multichannel images and we are currently extending the software for segmentation of other cell compartments and cell tracking for kinetic readouts (Hoefler, unpublished observations).

Overall, the simple graphical user interface of *CellCognition Explorer* and the independence of centralized large-scale computing infrastructure is well suited to applying deep learning and novelty detection to diverse questions in cell biology. Hence, *CellCognition Explorer* provides new opportunities for high-content screening and for– exploratory research in diverse biological fields.

MATERIALS AND METHODS

Cell culture

A HeLa cell line stably expressed histone H2B fused to mCherry and lamin B1 fused to EGFP (Daigle *et al.*, 2001) was generated from a HeLa Kyoto cell line as previously described (Schmitz and Gerlich, 2009). HeLa cells were cultured in DMEM (Life Technologies) supplemented with 10% (vol/vol) fetal bovine serum (FBS; Life Technologies), 1% (vol/vol) penicillin–streptomycin (Sigma-Aldrich), 500 µg ml^{-1} G418 (Life Technologies), and 0.5 µg ml^{-1} puromycin (Calbiochem). Homogeneous expression levels of the two transgenes were ensured by fluorescence-activated cell sorting. The cell line tested negatively for mycoplasm contamination in our quarterly testing routine. The parental HeLa cell line ("Kyoto strain") was obtained from S. Narumiya (Kyoto University, Japan) and validated by a Multiplex human Cell line Authentication test (MCA).

siRNA library transfection

A total of 1114 genes of the MitoCheck genome-wide RNAi screen validation data set (Neumann *et al.*, 2010) were targeted by two

siRNAs. To test target specificity, all siRNAs were mapped against the 2013 human genome (ENSEMBL V70), resulting in a unique match. siRNAs were delivered in 384-well imaging plates (Falcon) using solid-phase reverse transfection. Cells were seeded on the imaging plates using a Multidrop Reagent Dispenser (Thermo Scientific).

Automated microscopy

At 40 h after cell seeding, plates were imaged on a Molecular Devices ImageXpressMicro XL screening microscope using a magnification 20×, 0.75 NA, S Fluor dry objective (Nikon). Four positions with two Z-sections each (4 µm offset) were acquired in each well. Images were flat-field-corrected with the Metamorph software (Molecular Devices) using images acquired in empty wells to compensate for inhomogeneous illumination.

Image preprocessing for reference annotation with conventional supervised learning

To detect individual cells, we used local adaptive thresholding followed by a watershed split-and-merge segmentation as provided in the *CellCognition* framework (Held *et al.*, 2010). For each cell, the segmentation outline, its center of mass, and its bounding box were computed and saved to the cellH5 format (Sommer *et al.*, 2013). Then 239 cellular morphology features describing the texture, shape, and intensity distribution of each single cell object were computed as in Held *et al.* (2010). As a reference annotation in the chromatin morphology screen, we trained a multiclass SVM on nine phenotype classes with 592 training examples, achieving a cross-validation accuracy of 85.1%.

Outlier detection

An outlier is an observation that deviates so much from other observations as to arouse suspicions that it was generated by a different process (Hawkins, 1980). Novelty detection methods aim to identify outliers by inference based on intrinsic properties of the data without any further annotations. We implemented three state-of-the-art novelty detection methods: Mahalanobis distance (MD; Mahalanobis, 1936), kernel density estimation (Pimentel *et al.*, 2014), and one-class SVM (OC-SVM; Scholkopf *et al.*, 2001). All methods find novelty based on preextracted morphology features or on data-adaptive learned feature representations by deep learning autoencoders.

Mahalanobis distance

MD is based on the parametric assumption that the extracted features follow a multivariate Gaussian distribution (Mahalanobis, 1936). The Mahalanobis distance (Eq. 1) incorporates the mean µ and the covariance matrix **S** (multidimensional spread) of the data:

$$\mathrm{MD}(x) = \sqrt{(x-\mu)^T S^{-1} (x-\mu)} \tag{1}$$

For estimation of the mean and covariance, data samples were drawn from negative control conditions. Negative control conditions will also contain aberrant phenotypes (e.g., polypoid cells). To reduce the effect of strong outliers on the estimation process, some instances were excluded based on their univariate feature distributions (i.e., cells with the most extreme single feature values) prior to the estimation of the multivariate mean and covariance.

The mean feature vector represents the ideal normal cell, whereas the covariance estimates its multivariate spread. Once the mean and the covariance of the multivariate Gaussian distributions are estimated, an empirical cutoff distance is estimated (relative to negative control data) in such a way that 15% of the data points have a greater Mahalanobis distance than the chosen cutoff and therefore are considered abnormal. Cells from other experimental conditions are then assigned to the normal category (inliers) when their Mahalanobis distance is smaller than the calculated cutoff, or to the abnormal category (outliers) if the distance is greater. For all experiments shown in the article, >50,000 cells were sampled from negative control conditions for estimation of the parameters for outlier detection. For the chromatin morphology screen (Figure 2, c and d), a 10% cutoff was used. Negative control conditions in the screen for damage repair regulators (Figure 4, c and d) contained only very few outliers; a 5% cutoff value was hence chosen. The varying thresholds were empirically determined based on the expected number of outliers in the data sets.

Kernel density estimation

KDE is nonparametric estimation of a probability distribution (Pimentel *et al.*, 2014). Conceptually, KDE is related to histograms. Instead of quantizing a discrete probability density by counting how many data points fall into each bin, KDE does not require binning and outputs a continuous function. KDE places a zero-centered kernel function *K* of a bandwidth *h* on each data point followed by summing all these kernels:

$$\mathrm{KDE}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) \tag{2}$$

A multivariate Gaussian density distribution was used as the kernel *K*. The bandwidth *h* is given by the variance of the Gaussian distribution. In places where many data points are located, the estimated density will be high, while at places where no or few data points lie, the density will approach zero. Similarly to the cutoff parameter of Mahalanobis distance, a separation into abnormal (outlier) and normal (inlier) locations was achieved by thresholding so that the resulting estimated density had 15% of the data points classified as novel and 85% classified normal. The choice of an appropriate bandwidth is the key to KDE, because it determines the smoothness of the resulting density (e.g., too small bandwidths lead to spiky nonoverlapping densities). Therefore, a cross-validation strategy was used to estimate appropriate bandwidth from holdout data.

One-class support vector machine

Another nonparametric and nonlinear way to estimate densities in high-dimensional spaces is the OC-SVM (Scholkopf *et al.*, 2001). Similarly to the standard two-class SVM (Vapnik and Lerner, 1963) the OC-SVM yields a separating hyperplane with a maximal margin to distinguish the majority of the data from the origin. When a Gaussian kernel is used in the SVM optimization framework, the hyperplane implicitly corresponds to nonlinear decision boundaries in input space (Boser *et al.*, 1992). With this, the OC-SVM models arbitrary densities in high dimensions as opposed to more simplified ellipsoidal distributions when MD is used. Similarly to KDE, a parameter *g* (inverse SD of the Gaussian kernel) accounts for the smoothness of the resulting enclosing hyperplane. In the experiments shown in the article, *g* was optimized empirically by a grid search on a logarithmic scale. The parameter *n* in OC-SVM–based novelty detection is the maximal fraction of data points in the training set that can be assigned being an outlier. In accord with the methods described previously, we set $n = 0.15$.

Self-learned feature representation with autoencoders

The novelty detection algorithms described above can be trained on a manually curated feature set (e.g., as in Held *et al.*, 2010). Such

features are defined a priori and do not depend on the actual data at hand. Hence, information for distinguishing novel phenotypes from normal morphologies might not be sufficiently expressed by general feature sets.

To overcome this limitation, we implemented a class of artificial neural networks (ANNs) termed autoencoders (AE) to learn a data-dependent features representation directly from pixel data from individual cropped-out cells in an unsupervised manner. The bounding box size for cropping can be chosen freely by the user. An AE is an ANN that adjusts its internal weights to reconstruct its input as closely as possible (subject to a given objective function) by first learning how to encode its inputs effectively in the encoding layer and then decoding this back to the original domain with an inverse decoding layer (Hinton and Salakhutdinov, 2006). Once an AE has learned how to encode cell images into a code, this process is iterated to stack AEs in a hierarchical manner to generate deeper networks called stacked AEs (Hinton and Salakhutdinov, 2006). To ensure that an AE is not learning the simple identity function, the internal parameters have fewer degrees of freedom than its input/output (contractive autoencoding).

Recent progress in designing deep learning networks for various tasks in computer vision has led to the incorporation of *convolutional* and *pooling layers* (e.g., Krizhevsky *et al.*, 2012). Convolutional layers refer to learnable image filters usually of size 3×3 or 5×5 applied to the input images. Like other network weights in an ANN, these filters are simultaneously adapted during optimization to yield feature maps, which improve the overall network performance. To reduce the overall spatial dimensionality, convolutional layers are combined with max-pooling layers, which effectively downsamples the input image by selecting only the maximum pixel value for a defined neighborhood region (Krizhevsky *et al.*, 2012)—in our experiments, 2×2. We combined the stacked-contractive-AE model with convolutional and pooling layers to achieve more expressive codes in the AE paradigm. For faster AE training, *dropout layers* have been proposed (Vincent *et al.*, 2010), which randomly set a fixed percentage of inputs in each layer to zero. This compels the network to learn to reconstruct from corrupted inputs, which leads to less overfitting and faster training convergence.

To analyze the data shown in the article, we used AEs with a variable sequence of one or more convolutional layers (Supplemental Tables S5–S7), followed by max-pooling layers and a central stacked AE. The convolutional and pooling layers are mirrored in the decoding part of the AE with their corresponding inverse deconvolutional and unpooling layers. Between each pair of layers in the encoding part of the AE, optional dropout layers are inserted with a dropout probability ranging from 10 to 90%. The convolutional and fully connected layers require a nonlinear activation function. In our implementation, *sigmoid (s)* and *linear-rectifier (r)* activation functions are supported.

Autoencoder layout

To analyze the screening data—chromatin morphology (Figure 3, b and c), MitoCheck RNAi screen (Figure 4, a and b), and DNA damage repair RNAi screen (Figure 4, c and d)—we designed AEs such that an input image of a typical cropped single cell (40×40 pixels) is encoded through a series of convolutional layers, pooling layers, and fully connected layers, which successively reduce the input dimensions from $40 \times 40 = 1600$ image pixels to a code length of 144 or 64. Note that the *decoding* part in each AE is composed of layer-wise inversions of the encoding part. The layout of all AEs with precise information about the convolutional and dense layers can be found in Supplemental Tables S5–S7.

Autoencoder training

AEs were trained with stochastic gradient descent (SGD) based on the sum of squared residuals of input and output as objective function (Vincent *et al.*, 2010). The SGD method resembles a stochastic approximation of the gradient descent optimization method, where the true gradient of all the data is approximated by the gradients of so-called *mini-batches*, randomly selected subsets of the training data of a certain size *m*. All weights in the network are updated according to the back-propagated gradients of the objective function. This process is repeated and each round is referred to as one training epoch. Empirically, we found that alternating two variants of SGD—Nesterov momentum (Nesterov, 1983) and adaptive gradient descent (AdaGrad; Duchi *et al.*, 2011)—leads to fast convergence. Nesterov momentum adds a momentum term to the update rule to stabilize the learning process and to avoid local minima (Bengio *et al.*, 2013). AdaGrad modifies SGD by introducing a per-parameter learning rate leading to faster convergence (Duchi *et al.*, 2011).

For the AE used in chromatin morphology screening, we used first Nesterov momentum SGD updated for 128 epochs with a mini-batch size of 128, a learning rate of 0.02, and a momentum of 0.8 followed by AdaGrad updates for 128 epochs with a mini-batch size of 128 and a base learning rate of 0.1.

For the AE depicted in MitoCheck RNAi screening, we used a mini-batch size of 128. Nesterov momentum updates with a learning rate of 0.1 and a momentum of 0.5 were followed by AdaGrad updates with a learning rate of 0.1, both for 64 epochs. Then we reiterated with slightly adjusted values: a learning rate of 0.02 l and a momentum of 0.9 for the Nesterov updates and a learning rate of 0.05 for AdaGrad.

For the AE shown in DNA damage repair RNAi screening, we used first Nesterov momentum SGD with a learning rate of 0.02, a momentum of 0.9, and a mini-batch size of 128 followed by AdaGrad with a learning rate of 0.05 and a mini-batch size of 64.

CellCognition Explorer software package. The *CellCognition Explorer* software package consists of two programs: 1) the *CellCognition Explorer* main program, from which the novelty detection methods (e.g., MD) and supervised learning algorithms (multiclass SVM) can be executed and visualized; and 2) a program for the generation of self-learned feature representations (codes).

CellCognition Explorer main program. The software imports images of cell populations and converts them into galleries of individual cells. Original multichannel raw images are preprocessed in advance: Thumbnails, features, and outlines are calculated and saved to cellH5/hdf5. The software stores experimental conditions along with cell objects to enable efficient data export and visualization of results by statistical plots, such as object class counts per condition. The software provides functions for cell segmentation and conventional feature extraction as in Held *et al.* (2010) for multiple image channels.

The graphical user interface is designed to handle a few thousand cells smoothly on a standard desktop PC. Larger data sets can be loaded in batches. The software is available as a binary standalone version for Mac OSX and Windows 7.

CellCognition Explorer Deep Learning Module. With this program, the user can train a deep learning AE from negative controls. The main input is a cellH5 file (Sommer *et al.*, 2013) containing the image data and center positions of all detected cell objects, together with a position mapping (text-)file indicating negative control positions used for randomly sampling cells for the training process.

After training, the resulting AE model can be used in the *encoding* step. The output file of the encoding step can be loaded directly into the *CellCognition Explorer* main program for novelty detection and/or supervised learning. *CellCognition Explorer Deep Learning Module* provides a graphical user interface to specify all processing parameters (Supplemental Figure S2). The software is provided as source code (LGPL) and as a binary installer using the docker framework. For installation in MacOS X, a docker execution script is provided.

Implementation details

CellCognition Explorer is implemented in Python 2.7, and the deep learning module makes use of the Theano (ver. 0.7.0) and lasagna (ver. 0.2) library for deep learning and GPU computing interfacing with NVIDIA CUDA (ver. 5.5). For the graphical user interfaces, PyQt5 (ver. 5.3.2) binding to the C++ library Qt (ver. 5.3.1) was used. Novelty detection methods are based on the sklearn library (ver. 0.16.0). Image-based screening data are segmented and processed using *CellCognition Analyzer* (ver. 1.6.0) and analysis results were saved in the cellH5 format (ver. 1.3.1).

License

CellCognition Explorer is released under the GNU General Public License version 3 (GPLv3).

Computing hardware

All data shown in this article were processed on an HP-Z820 desktop workstation with Intel Xeon CPU E5-2665-0 @ 2.40 GHz (16 CPUs) equipped with 64 GB of RAM and an NVIDIA GeForce GTX TITAN X graphics adapter.

ACKNOWLEDGMENTS

D.W.G. has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under Grant Agreements 241548 (MitoSys) and 258068 (Systems Microscopy), from an ERC Starting Grant under Agreement 281198 (DIVIMAGE), and from the Austrian Science Fund (FWF) Project SFB F34-06 (Chromosome Dynamics). We thank the IMBA/IMP BioOptics facility for technical support, Claudia Lukas, Thomas Walter, Jean-Karim Hériché, Beate Neumann, and Jan Ellenberg for providing original image data of published RNAi screens, Thomas Walter for assistance with image preprocessing, Sara Cuylen for assistance with the graphical user interface design, and the life science editors for editorial support.

REFERENCES

Abramoff MD, Magalhaes PJ, Ram SJ (2004). Image processing with ImageJ. Biophoton Int 11, 36–42.

Bakal C, Aach J, Church G, Perrimon N (2007). Quantitative morphological signatures define local signaling networks regulating cell morphology. Science 316, 1753–1756.

Bengio Y, Boulanger-Lewandowski N, Pascanu R (2013). Advances in optimizing recurrent networks. Paper presented at: IEEE International Conference on Acoustics, Speech and Signal Processing.

Boland MV, Murphy RF (2001). A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of HeLa cells. Bioinformatics 17, 1213–1223.

Boser BE, Guyon I, Vapnik V (1992). A training algorithm for optimal margin classifiers. COLT '92: Proceedings of the Fifth Annual Workshop on Computational Learning Theory.

Boutros M, Heigwer F, Laufer C (2015). Microscopy-based high-content screening. Cell 163, 1314–1325.

Carpenter AE, Jones TR, Lamprecht MR, Clarke C, Kang IH, Friman O, Guertin DA, Chang JH, Lindquist RA, Moffat J, *et al.* (2006). CellProfiler: image analysis software for identifying and quantifying cell phenotypes. Genome Biol 7, R100.

Conrad C, Gerlich DW (2010). Automated microscopy for high-content RNAi screening. J Cell Biol 188, 453–461.

Cuylen S, Blaukopf C, Politi AZ, Muller-Reichert T, Neumann B, Poser I, Ellenberg J, Hyman AA, Gerlich DW (2016). Ki-67 acts as a biological surfactant to disperse mitotic chromosomes. Nature 535, 308–312.

Daigle N, Beaudouin J, Hartnell L, Imreh G, Hallberg E, Lippincott-Schwartz J, Ellenberg J (2001). Nuclear pore complexes form immobile networks and have a very low turnover in live mammalian cells. J Cell Biol 154, 71–84.

Duchi J, Hazan E, Singer Y (2011). Adaptive subgradient methods for online learning and stochastic optimization. J Mach Learn Res 12, 2121–2159.

Durr O, Sick B (2016). Single-cell phenotype classification using deep convolutional neural networks. J Biomol Screen 1–6.

Failmezger H, Frohlich H, Tresch A (2013). Unsupervised automated high throughput phenotyping of RNAi time-lapse movies. BMC Bioinformat 14, 292.

Gentleman RC, Carey VJ, Bates DM, Bolstad B, Dettling M, Dudoit S, Ellis B, Gautier L, Ge Y, Gentry J, *et al.* (2004). Bioconductor: open software development for computational biology and bioinformatics. Genome Biol 5, R80.

Goshima G, Wollman R, Goodwin SS, Zhang N, Scholey JM, Vale RD, Stuurman N (2007). Genes required for mitotic spindle assembly in *Drosophila* S2 cells. Science 316, 417–421.

Gudjonsson T, Altmeyer M, Savic V, Toledo L, Dinant C, Grofte M, Bartkova J, Poulsen M, Oka Y, Bekker-Jensen S, *et al.* (2012). TRIP12 and UBR5 suppress spreading of chromatin ubiquitylation at damaged chromosomes. Cell 150, 697–709.

Hawkins DM (1980). Identification of Outliers. Monographs on Statistics and Applied Probability, London/New York: Chapman and Hall.

Held M, Schmitz MH, Fischer B, Walter T, Neumann B, Olma MH, Peter M, Ellenberg J, Gerlich DW (2010). CellCognition: time-resolved phenotype annotation in high-throughput live cell imaging. Nat Methods 7, 747–754.

Hinton GE, Salakhutdinov RR (2006). Reducing the dimensionality of data with neural networks. Science 313, 504–507.

Jones TR, Carpenter AE, Lamprecht MR, Moffat J, Silver SJ, Grenier JK, Castoreno AB, Eggert US, Root DE, Golland P, *et al.* (2009). Scoring diverse cellular morphologies in image-based screens with iterative feedback and machine learning. Proc Natl Acad Sci USA 106, 1826–1831.

Kraus OZ, Ba JL, Frey BJ (2016). Classifying and segmenting microscopy images with deep multiple instance learning. Bioinformatics 32, i52–i59.

Kraus OZ, Grys BT, Ba J, Chong Y, Frey BJ, Boone C, Andrews BJ (2017). Automated analysis of high-content microscopy data with deep learning. Mol Syst Biol 13, 924.

Krizhevsky A, Sutskever I, Hinton GE (2012). Imagenet classification with deep convolutional neural networks. Paper presented at Advances in Neural Information Processing Systems.

LeCun Y, Bengio Y, Hinton G (2015). Deep learning. Nature 521, 436–444.

Liberali P, Snijder B, Pelkmans L (2014). A hierarchical map of regulatory genetic interactions in membrane trafficking. Cell 157, 1473–1487.

Mahalanobis PC (1936). On the generalised distance in statistics. Proc Natl Inst Sci India 2, 49–55.

Manning S, Shamir L (2014). CHLOE: a software tool for automatic novelty detection in microscopy image datasets. J Open Res Software 2, e128.

Mattiazzi Usaj M, Styles EB, Verster AJ, Friesen H, Boone C, Andrews BJ (2016). High-content screening for quantitative cell biology. Trends Cell Biol 26, 598–611.

Misselwitz B, Strittmatter G, Periaswamy B, Schlumberger MC, Rout S, Horvath P, Kozak K, Hardt WD (2010). Enhanced CellClassifier: a multi-class classification tool for microscopy images. BMC Bioinformat 11, 30.

Murphy RF, Velliste M, Porreca G (2003). Robust numerical features for description and classification of subcellular location patterns in fluorescence microscope images. J VLSI Sig Proc Syst 35, 311–321.

Nesterov Y (1983). A method of solving a convex programming problem with convergence rate O (1/k2). Soviet Mathematics Doklady 27, 372–376.

Neumann B, Held M, Liebel U, Erfle H, Rogers P, Pepperkok R, Ellenberg J (2006). High-throughput RNAi screening by time-lapse imaging of live human cells. Nat Methods 3, 385–390.

Neumann B, Walter T, Heriche JK, Bulkescher J, Erfle H, Conrad C, Rogers P, Poser I, Held M, Liebel U, *et al.* (2010). Phenotypic profiling of the human genome by time-lapse microscopy reveals cell division genes. Nature 464, 721–727.

Parnamaa T, Parts L (2017). Accurate classification of protein subcellular localization from high throughput microscopy images using deep learning. G3 7, 1385–1392.
Pimentel MAF, Clifton DA, Clifton L, Tarassenko L (2014). A review of novelty detection. Signal Process 99, 215–249.
Rajaram S, Pavie B, Wu LF, Altschuler SJ (2012). PhenoRipper: software for rapidly profiling microscopy images. Nat Methods 9, 635–637.
Ramo P, Sacher R, Snijder B, Begemann B, Pelkmans L (2009). CellClassifier: supervised learning of cellular phenotypes. Bioinformatics 25, 3028–3030.
Sainath TN, Mohamed AR, Kingsbury B, Ramabhadran B (2013). Deep convolutional neural networks for LVCSR. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 8614–8618.
Schindelin J, Arganda-Carreras I, Frise E, Kaynig V, Longair M, Pietzsch T, Preibisch S, Rueden C, Saalfeld S, Schmid B, *et al*. (2012). Fiji: an open-source platform for biological-image analysis. Nat Methods 9, 676–682.
Schmitz MH, Gerlich DW (2009). Automated live microscopy to study mitotic gene function in fluorescent reporter cell lines. Methods Mol Biol 545, 113–134.
Schmitz MH, Held M, Janssens V, Hutchins JR, Hudecz O, Ivanova E, Goris J, Trinkle-Mulcahy L, Lamond AI, Poser I, *et al*. (2010). Live-cell imaging RNAi screen identifies PP2A-B55alpha and importin-beta1 as key mitotic exit regulators in human cells. Nat Cell Biol 12, 886–893.
Scholkopf B, Platt JC, Shawe-Taylor J, Smola AJ, Williamson RC (2001). Estimating the support of a high-dimensional distribution. Neural Comput 13, 1443–1471.
Sommer C, Gerlich DW (2013). Machine learning in cell biology—teaching computers to recognize phenotypes. J Cell Sci 126, 5529–5539.
Sommer C, Held M, Fischer B, Huber W, Gerlich DW (2013). CellH5: a format for data exchange in high-content screening. Bioinformatics 29, 1580–1582.
Taigman Y, Yang M, Ranzato M, Wolf L (2014). DeepFace: Closing the gap to human-level performance in face verification. Proc CVPR IEEE 1701–1708.
Vapnik V, Lerner A (1963). Pattern recognition using generalized portrait method. Autom Remote Control 24, 774–780.
Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 11, 3371–3408.
Yin Z, Sadok A, Sailem H, McCarthy A, Xia XF, Li FH, Garcia MA, Evans L, Barr AR, Perrimon N, *et al*. (2013). A screen for morphological complexity identifies regulators of switch-like transitions between discrete cell shapes. Nat Cell Biol 15, 860–871.
Yin Z, Zhou X, Bakal C, Li F, Sun Y, Perrimon N, Wong ST (2008). Using iterative cluster merging with improved gap statistics to perform online phenotype discovery in the context of high-throughput RNAi screens. BMC Bioinformatics 9, 264.
Zhong Q, Busetto AG, Fededa JP, Buhmann JM, Gerlich DW (2012). Unsupervised modeling of cell morphology dynamics for time-lapse microscopy. Nat Methods 9, 711–713.