Prospective identification of hematopoietic lineage choice by deep learning

Felix Buggenthin[1,6], Florian Buettner[1,2,6], Philipp S Hoppe[3,4], Max Endele[3], Manuel Kroiss[1,5], Michael Strasser[1], Michael Schwarzfischer[1], Dirk Loeffler[3,4], Konstantinos D Kokkaliaris[3,4], Oliver Hilsenbeck[3,4], Timm Schroeder[3,4], Fabian J Theis[1,5] & Carsten Marr[1]

Differentiation alters molecular properties of stem and progenitor cells, leading to changes in their shape and movement characteristics. We present a deep neural network that prospectively predicts lineage choice in differentiating primary hematopoietic progenitors using image patches from brightfield microscopy and cellular movement. Surprisingly, lineage choice can be detected up to three generations before conventional molecular markers are observable. Our approach allows identification of cells with differentially expressed lineage-specifying genes without molecular labeling.

Long-term, high-throughput time-lapse microscopy is a powerful tool for studying the differentiation processes of single cells in unprecedented temporal resolution[1]. A high frequency of brightfield imaging (typically on the scale of a few minutes) ensures that moving single cells and cell divisions can be accurately tracked and used for the construction of cellular genealogies. Additionally, fluorescence imaging (which, owing to cell phototoxicity, is typically only possible when spaced out by intervals of 30 min or more[2]) allows the quantification of molecular lineage markers[3,4]. However, molecular lineage markers are only available for specific cell types that are often already differentiated[5,6], hindering the early identification of differentiating cells.

Therefore, we set out to exploit the information in the abundant brightfield images of time-lapse experiments for prospective detection of lineage commitment. We developed our method on time-lapse experiments of primary murine hematopoietic stem and progenitor cells (HSPCs) differentiating into either the granulocytic/monocytic (GM) or the megakaryocytic/erythroid (MegE) lineage (**Fig. 1a**, **Supplementary Fig. 1**). Branches (i.e., a cell and all its predecessors) were generated by automatically linking an image patch of 27 × 27 pixels covering the mass-centered body of the cell to every time point of a manual cell track (**Fig. 1b**,**c**, **Supplementary Note 1**). We annotated the lineage commitment when the respective lineage marker was detectable in the fluorescence channel (CD16/32 for the GM and GATA1-mCherry for the MegE lineage) and assigned all tracked cells to one of three categories: (i) "annotated" cells with clear expression of the marker within the cell lifetime, (ii) "latent" cells with no immediate expression of the marker but expression in a subsequent generation, and (iii) "unknown" cells with no expression of the marker in current or subsequent generations (**Supplementary Fig. 2a**–**c**). Our data set comprised 150 genealogies from 3 independent experiments with a total of 5,922 single cells (**Supplementary Fig. 2d**–**f**). Each cell was imaged ~400 times, resulting in more than 2,400,000 image patches.

We used these millions of image patches to build a classifier that predicts the lineage choice of a stem cell's progeny toward either the GM or the MegE lineage. To efficiently leverage the information in our data set, we built on recent advances in deep neural networks for image classification. We combine a convolutional neural network (CNN) with a recurrent neural network (RNN) architecture to automatically extract local image features and exploit the temporal information of the single-cell tracks (**Fig. 1d**,**e**). Specifically, three connected convolutional layers extract image features, resulting in increasingly global representations of the image patches. As a CNN allows no direct inclusion of features other than pixel information, we introduced a concatenation layer combining the highest-level spatial features with cell displacement, which was followed by a fully connected layer that can be interpreted as patch features. To train the CNN, this layer is connected to output nodes, resulting in a lineage score for each patch. Lineage scores of 0 or 1 indicate a strong similarity to cell patches from either the MegE or GM lineage, respectively. Next, in order to classify individual cells as committed to either lineage, we used the patch features as input for the RNN. To model long-range temporal dependencies in the data without suffering from the vanishing-gradient problem[7], we used a bidirectional long short-term memory (LSTM) architecture[8,9] (**Fig. 1e**).

After filtering out of all unknown cells (containing both uncommitted cells and committed cells for which the markers had not yet switched on at the end of the experiments), the data set to train and evaluate our method consisted of 4,402 single cells (~1,700,000 image patches) with onset of the annotated or latent marker (34% MegE and 66% GM, **Supplementary Fig. 2e**,**f**). To assess the generalization power of our model to reliably predict a cell's choice of putative lineage in independent experiments, we trained our CNN–RNN on two

[1]Institute of Computational Biology, Helmholtz Zentrum München–German Research Center for Environmental Health, Neuherberg, Germany. [2]European Molecular Biology Laboratory, European Bioinformatics Institute, Wellcome Genome Campus, Hinxton, UK. [3]Department of Biosystems Science and Engineering, Eidgenössische Technische Hochschule (ETH) Zurich, Basel, Switzerland. [4]Research Unit Stem Cell Dynamics, Helmholtz Zentrum München–German Research Center for Environmental Health, Neuherberg, Germany. [5]Department of Mathematics, Technische Universität München, Garching, Germany. [6]These authors contributed equally to this work. Correspondence should be addressed to T.S. (timm.schroeder@bsse.ethz.ch) or F.J.T. (fabian.theis@helmholtz-muenchen.de) or C.M. (carsten.marr@helmholtz-muenchen.de).

RECEIVED 21 OCTOBER 2015; ACCEPTED 17 JANUARY 2017; PUBLISHED ONLINE 20 FEBRUARY 2017; DOI:10.1038/NMETH.4182

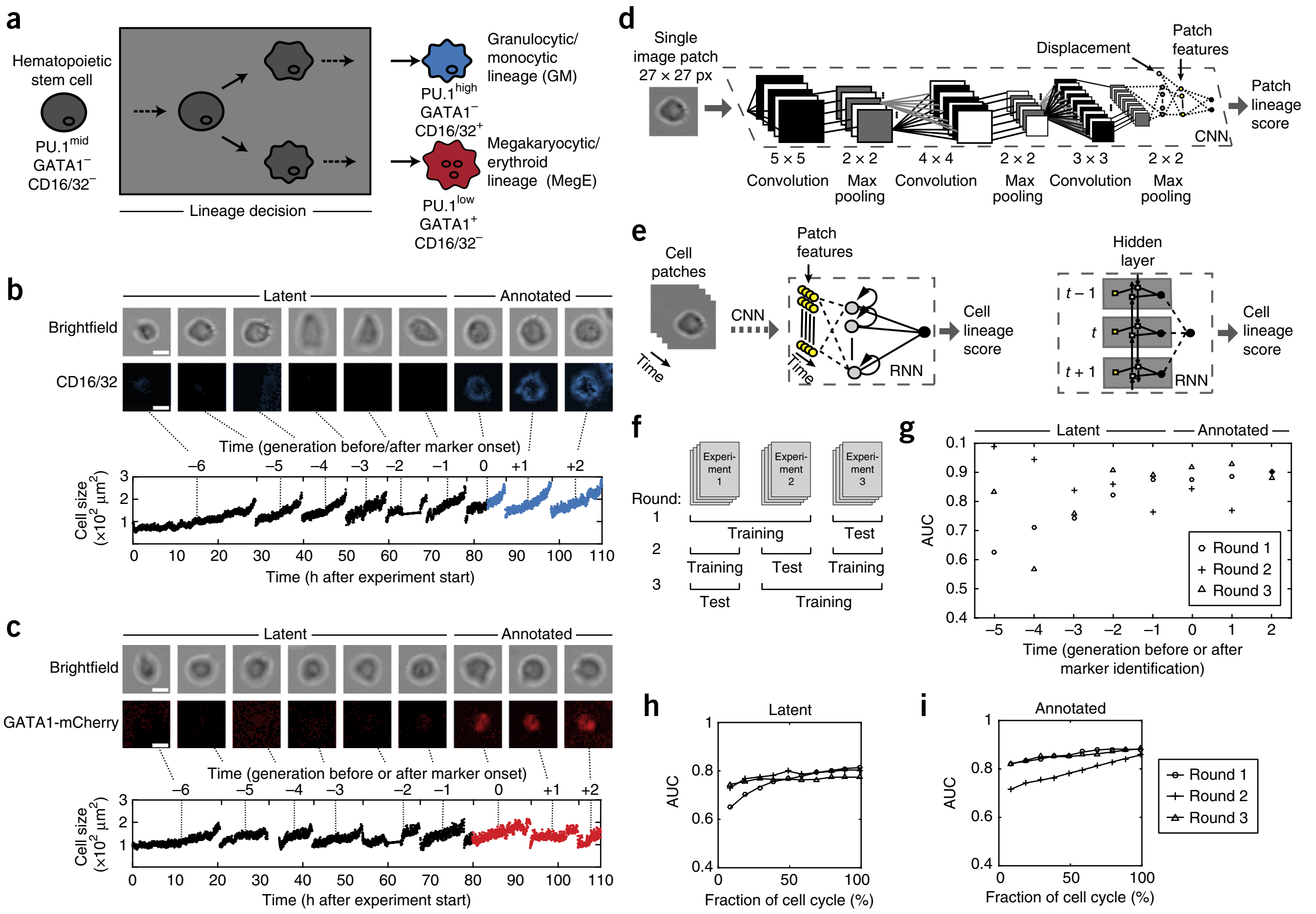

Figure 1 | Prediction of hematopoietic lineage choice up to three generations before molecular marker annotation using deep neural networks. (**a**) Hematopoietic stem cells (gray) can differentiate and are annotated as committed toward the granulocytic/monocytic (GM, blue) lineage via detection of CD16/32 or toward the megakaryocytic/erythroid lineage (MegE, red) via detection of GATA1-mCherry expression. These conventional markers appear after the lineage decision of the cell (gray box). (**b**,**c**) Exemplary image patches of a branch of single cells committing to either GM (**b**, upper row) or MegE (**c**, upper row) lineage (scale bars, 10 μm). Cells with no marker expression are "latent", and cells with marker expression are "annotated" (**b**,**c**, middle row). The graphs in the lower rows show cell size. (**d**) A schematic of the convolutional neural network (CNN) calculating a patch lineage score for each image patch (see Online Methods for details). (**e**) To account for temporal dependencies, we feed the CNN-derived patch features of a cell (yellow) in a recurrent neural network (RNN). The nodes in the hidden layer are connected to output nodes, as well as all other hidden nodes across time (left); this temporal dependency is further illustrated in an unrolled representation of the RNN (right), where yellow squares represent the patch-feature vectors at a specific time point, and forward and backward arrows reflect the bidirectional architecture of the RNN. (**f**) Schematic of round-robin training and testing. (**g**) Area under the receiver operating characteristics curve (AUC; 1.0 = perfect classification, 0.5 = random guessing) determines the performance of the trained models. Annotated cells (generations 0, +1, +2) and latent cells up to three generations before marker onset (generations −3, −2, −1) show AUCs higher than 0.77 (n = 3 rounds, 4,204 single cells in total). (**h**,**i**) AUCs when only (contiguous) subsets of image patches are used to compute the cell lineage score. AUCs over 0.75 were reached when using the first ~25% of time points in the cell cycle from latent (**h**) and annotated cells (**i**), respectively.

experiments and tested the resulting model on the third experiment; we repeated this procedure three times in a round-robin fashion (**Fig. 1f**) and evaluated the performance of the trained model by the area under the curve (AUC) of the receiver operating characteristic and F_1 score (**Supplementary Fig. 3**). Our method achieved high AUCs of 0.87 ± 0.01 (mean ± s.d., $n = 3$ rounds) on annotated cells, indicating that morphology and displacement suffice to detect the lineage choice of HSPCs. Interestingly, the reported AUCs for latent cells were also high (0.79 ± 0.02, mean ± s.d., $n = 3$ rounds), suggesting that latent cells are morphologically different before expression of an identifiable marker. We investigated this finding further by analyzing AUCs for every generation separately (**Fig. 1g**). AUCs stayed at comparable and robust levels from one to three generations before an annotated marker onset (0.84 ± 0.07, 0.86 ± 0.04 and 0.78 ± 0.05, respectively, mean ± s.d., $n = 3$ rounds). At four and five generations before the marker onset, the decline and variance of AUCs (0.74 ± 0.19 and 0.82 ± 0.19, respectively, mean ± s.d., $n = 3$) suggested that the differences in morphology and displacement were no longer sufficient to differentiate GM- and MegE-committed cells. We achieved similar performance when using our classifier on data from genetically nonmodified mice (Online Methods and **Supplementary Fig. 4**). To assess the performance of our CNN–RNN on cells in which tracking is stopped before cell cycle completion (for example, in an 'online-prediction' scenario where the lineage score is calculated while the experiment runs), we computed AUCs on the basis of only a subset of brightfield patches. While single patches are insufficient for correct prediction, AUCs over 0.75 were reached when using the first ~25% of time points in the cell cycle from latent (three, two and one generation before marker onset; **Fig. 1h**) and annotated (**Fig. 1i**) cells.

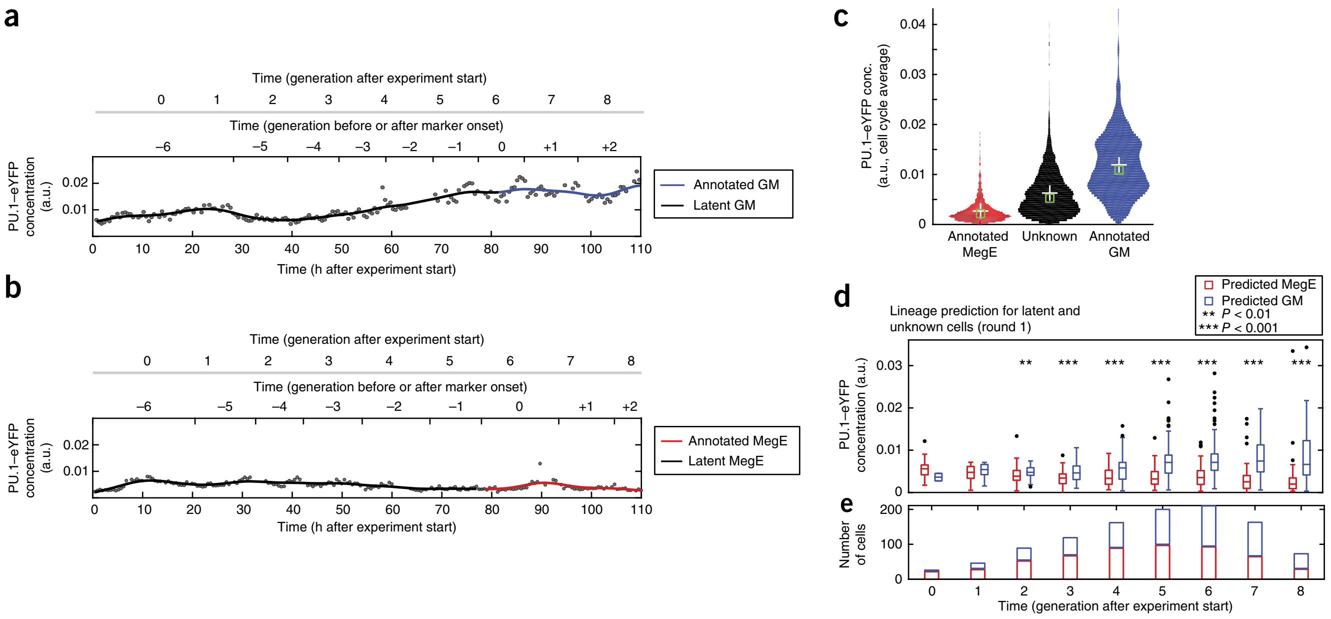

Figure 2 | Subsets of cells with differential PU.1–eYFP expression can be distinguished two generations after experiment start. (**a**) Increase of PU.1–eYFP concentration in a branch with annotated GM marker onset. a.u., arbitrary unit. (**b**) Decrease of PU.1–eYFP concentration in a branch with annotated MegE marker onset. Concentrations (dots) are fitted with a B-spline (black/colored line). (**c**) Cells without marker expression (unknown or latent fate, black) show an intermediate PU.1–eYFP concentration compared to cells annotated for GM (blue) and MegE (red) lineage. (**d**) Concentration of PU.1–eYFP for unknown and latent cells in generations after the experiment start; they are subdivided into predicted GM (blue) and MegE (red) on the basis of their CNN–RNN lineage score for round 1 (see **Supplementary Fig. 4** for rounds 2 and 3). The PU.1–eYFP concentration is significantly different in generations 2 ($P < 0.01$) and 3–8 ($P < 0.001$, unpaired Wilcoxon rank-sum test) after experiment start between the two predicted groups. Error bars extend to 1.5 times the interquartile range. (**e**) Significantly different PU.1–eYFP expression in generation 2 after experiment start is detected in 55 MegE (red) vs. 34 GM (blue) predicted cells.

The combination of single-cell transcriptomic profiling[10] with our approach would, in principle, allow comparison of the expression patterns of early committed cells after stratification in accordance with their lineage score to identify differentially expressed regulators. As the usefulness of our approach for such an experiment is hard to assess from the AUCs, we evaluated our method with the expression of PU.1, a transcription factor that is tagged with enhanced yellow fluorescent protein (eYFP) in our cells (Online Methods). As expected[6,11], PU.1–eYPF was upregulated in GM-annotated cells (**Fig. 2a**), was downregulated in MegE-annotated cells (**Fig. 2b**), and showed intermediate expression in cells without expression of annotated markers (**Fig. 2c**). If our proposed method was capable of reliably predicting a cell's early lineage choice, we should be able to stratify all cells *in silico* into either GM- or MegE-committed cells at every time point of a hypothetical experiment. These groups, in turn, are expected to differentially express PU.1–eYFP. Thus, we classified every latent and unknown cell into one of two groups by analyzing whether each cell's lineage score was above (GM) or below (MegE) a threshold of 0.5 (**Fig. 2d**, **Supplementary Fig. 5a**,**b**). We found the two groups to differentially express PU.1–eYFP from two generations after the experiment start and onwards ($P < 0.01$ in generation 2, $P < 0.001$ in generations 3–8, unpaired Wilcoxon rank-sum test, **Fig. 2d**) with at least 89 cells per generation (**Fig. 2e**). As only 2 ± 1% (mean ± s.d., $n = 3$ rounds) of GM and 15 ± 8% (mean ± s.d., $n = 3$ rounds) of MegE marker onsets were annotated earlier than four generations after experiment start, our method is clearly superior to lineage identification on the basis of traditional molecular markers in that particular time window (**Supplementary Fig. 5c**).

Different automated methods have been used for single-cell classification[12–15]. We compared the performance of our CNN–RNN method to that of several other approaches. To this end, we trained a random forest model and a support vector machine (SVM) with a set of 87 morphological features and displacement. In addition, we evaluated the algorithmic information theoretic prediction (AITP)[15], a method designed to predict the differentiation fate of retinal progenitor cells using a set of six movement and size features, as well as a conditional random field (CRF) approach based on scale-invariant feature transform (SIFT) features[13,16]. Finally, we quantified the performance of two CNN models, in which we averaged patch-wise lineage scores to obtain cell-specific predictions (**Supplementary Fig. 6**). We evaluated all methods in terms of AUC; in addition, to quantify performance in terms of precision and recall, we further compared the F_1 scores of all methods. While our CNN–RNN outperformed the SVM on annotated cells and was on par with SVM on latent cells, we found the AUCs for the random forest method to be similar on both sets (**Supplementary Fig. 7a**). However, the CNN–RNN achieved considerably higher F_1 scores than the random forest, AITP and CRF-based approaches (**Supplementary Fig. 7b**), indicating that these methods were more poorly calibrated than CNN–RNN. Using a RNN to model cell dynamics rather than using simple averaging in the CNN approach yielded a slight but consistent increase in predictive power (in terms of F_1 score). This suggests that the CNN–RNN approach yields more robust results when applied to new experiments that were not part of the training procedure.

The computation of hand-crafted features can be time-consuming and biased as appropriate features have to be chosen carefully for every new data set. Instead, our CNN–RNN takes only raw brightfield image patches as input, rendering the explicit computation of features obsolete. However, knowing which interpretable features are most important for lineage prediction could support the design of novel experiments to study hematopoietic differentiation[2]. Since the features implicitly derived within the CNN are difficult to extract and interpret, we evaluated the feature importance reported by the trained random forest model[17]. We found that multiple features—most importantly displacement and simple morphological features (maximal and mean pixel intensity and cell size)—are required for correct random forest classification (**Supplementary Fig. 8**). To investigate the relevance of the displacement feature, we retrained our CNN–RNN model omitting displacement, resulting in a somewhat lower predictive power for latent cells (from 0.79 ± 0.02 to 0.76 ± 0.04); this result illustrates that displacement is used by the network. Moreover, we found slight differences in the displacement and cell diameter of GM- vs. MegE-predicted cells (**Supplementary Fig. 9**).

Previously, computational image analysis has been used to predict the fate of retinal progenitor cells from rats[15] and to identify characteristic features of two populations of progenitor cells in the cerebral cortex[18]. In the hematopoietic system, long cell-cycle times and trailing cellular projections[19], as well as reduced proliferation and increased asynchronous divisions[20], have been identified as key features of HSPC self-renewal via time-lapse microscopy. Our method allows users to prospectively discriminate between two different lineages arising from hematopoietic progenitors and performs robustly on multiple independent time-lapse experiments. For a single cell, this prediction relies on a sequence of brightfield images and the combination of multiple features—single images and single features do not suffice. The brightfield-based prediction frees fluorescence channels that are currently used for lineage marker annotation. Moreover, the differential expression of PU.1–eYFP implies that our method can be used to identify important regulators of lineage choice when combined with single-cell profiling. While parameterizing deep neural networks requires large quantities of training data, the application of this approach matches the large amount of labeled image data that emerges from the diligent and careful annotation of time-lapse microscopy movies[4,6] and can be used for training the networks. Compared to other machine learning methods, our CNN–RNN method predicts fast and robustly for new experiments not used for training. While it is independent of a cell-type-specific, curated set of features and requires no high-level feature calculation, the interpretation of the derived features is challenging and an active field of research. Our approach is versatile and well-suited to analyze differentiation processes in biological systems where robustness is pivotal, suitable feature sets are unknown or fast prediction is required.

METHODS

Methods, including statements of data availability and any associated accession codes and references, are available in the online version of the paper.

Note: Any Supplementary Information and Source Data files are available in the online version of the paper.

ACKNOWLEDGMENTS

We thank S. Pölsterl for comments on the manuscript. This work was supported by the German Federal Ministry of Education and Research (BMBF), the European Research Council starting grant (Latent Causes grant 259294, FJT), the BioSysNet (Bavarian Research Network for Molecular Biosystems, FJT), the German Research Foundation (DFG) within the SPPs 1395 and 1356 to FJT, the UK Medical Research Council (Career Development Fellowship MR/M01536X/1 to FBT), and the Swiss National Science Foundation grant 31003A_156431 and SystemsX IPhD grant 2014/244 to TS.

AUTHOR CONTRIBUTIONS

F. Buggentin developed the image processing and machine learning pipeline, tracked cells and analyzed the data. F. Buettner developed the deep neural network approach with M.K., M. Strasser and M. Schwarzfischer contributed to image processing and data analysis. P.S.H. developed and conducted all experiments and tracked cells with M.E.; T.S. developed and supervised data generation. D.L., K.D.K., and O.H. contributed to data generation and analysis. F.J.T. and T.S. initiated the study. C.M. supervised the study with F.J.T., and wrote the manuscript with F.B.U. and F.B.T. All authors commented on the manuscript.

COMPETING FINANCIAL INTERESTS

The authors declare no competing financial interests.

1. Skylaki, S., Hilsenbeck, O. & Schroeder, T. *Nat. Biotechnol.* **34**, 1137–1144 (2016).
2. Schroeder, T. *Nat. Methods* **8** (Suppl.), S30–S35 (2011).
3. Rieger, M.A. & Schroeder, T. *Cells Tissues Organs* **188**, 139–149 (2008).
4. Filipczyk, A. *et al. Nat. Cell Biol.* **17**, 1235–1246 (2015).
5. Rieger, M.A., Hoppe, P.S., Smejkal, B.M., Eitelhuber, A.C. & Schroeder, T. *Science* **325**, 217–218 (2009).
6. Hoppe, P.S. *et al. Nature* **535**, 299–302 (2016).
7. Bengio, Y., Simard, P. & Frasconi, P. *IEEE Trans. Neural Netw.* **5**, 157–166 (1994).
8. Graves, A. & Schmidhuber, J. *Neural Netw.* **18**, 602–610 (2005).
9. Graves, A., Jaitly, N. & Mohamed, A.-r. in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)* 273–278 (IEEE, 2013).
10. Sandberg, R. *Nat. Methods* **11**, 22–24 (2014).
11. Hoppe, P.S., Coutu, D.L. & Schroeder, T. *Nat. Cell Biol.* **16**, 919–927 (2014).
12. Veta, M. *et al. Med. Image Anal.* **20**, 237–248 (2015).
13. Liu, A.-A., Li, K. & Kanade, T. *IEEE Trans. Med. Imaging* **31**, 359–369 (2012).
14. Huh, S., Ker, D.F.E., Bise, R., Chen, M. & Kanade, T. *IEEE Trans. Med. Imaging* **30**, 586–596 (2011).
15. Cohen, A.R., Gomes, F.L.A.F., Roysam, B. & Cayouette, M. *Nat. Methods* **7**, 213–218 (2010).
16. Liu, A.-A., Li, K. & Kanade, T. in *2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro* 580–583 (IEEE, 2010).
17. Breiman, L. *Mach. Learn.* **45**, 5–32 (2001).
18. Winter, M.R. *et al. Stem Cell Rep.* **5**, 609–620 (2015).
19. Dykstra, B. *et al. Proc. Natl. Acad. Sci. USA* **103**, 8185–8190 (2006).
20. Lutolf, M.P., Doyonnas, R., Havenstrite, K., Koleckar, K. & Blau, H.M. *Integr. Biol.* **1**, 59–69 (2009).

ONLINE METHODS

Generation of knock-in mice. The generation of knock-in mouse lines with reading frames for yellow (enhanced yellow fluorescent protein; eYFP) and red (mCherry) fluorescent proteins knocked into the gene loci for *PU.1* and *Gata1*, respectively, has previously been described[6]. The resulting PU.1^{eYFP} and GATA1^{mCherry} mice were mated to create male PU.1^{eYFP}GATA1^{mCherry} mice with no discernible phenotype[6].

Purification of primary murine hematopoietic stem and progenitor cells. Femurs, tibiae and ilia were removed from mice aged 12–14 weeks, and bone marrow was extracted. HSPCs were sorted to both a technical purity of >95% and an expected functional purity of at least 40–60% by flow cytometry[21,22]. Directly after sorting, cells were incubated with CD16/32 Alexa Fluor 647 antibody and seeded on a plastic slide (μ-slide VI coated with Fibronectin, Integrated BioDiagnostics GmbH, Munich, Germany) with physically separated channels in serum-free medium (StemSpan SFEM, StemCell Technologies) supplied with cytokines that only promote differentiation toward myeloid cells. Animal experiments were approved by veterinary office of Canton Basel-Stadt, Switzerland and Regierung von Oberbayern, Germany.

Long-term time-lapse microscopy data. For each experiment, channels of a plastic slide were subdivided into 72–78 overlapping fields of view. Each field of view corresponds to a 1388 × 1040 pixel image that was saved in 8-bit png format. Images were acquired using Axio Observer Z1 microscopes (Zeiss), equipped with a 0.63× TV-adaptor (Zeiss), an AxioCamHRm camera (Zeiss) and a 10× fluar objective (Zeiss). Microscopes were surrounded by an incubator to keep a constant temperature of 37 °C, and cells were maintained in 5% CO_2. Each field of view was imaged in intervals of 60–120 s (brightfield channel), 25–40 min (PU.1–eYFP and GATA1–mCherry channels) and 120–240 min (CD16/32 channel) for up to 8 d (**Supplementary Fig. 1**). Automatic focusing was achieved using a hardware autofocus (Zeiss), which was set to 18 μm below the optimal focal plane to acquire slightly blurred images that are optimal for cell detection[23].

Time-lapse experiments from PU.1^{eYFP}GATA1^{mCherry} mice (3 experiments, 150 genealogies, 5,922 cells, 2,477,784 cell patches, **Supplementary Fig. 2**) and non-genetically modified C57BL/6J mice (1 experiment, 29 genealogies, 266 cells, 157,384 cell patches) were used in this study, comprising a total size of ~1TB of disc space.

Single-cell tracking and annotation of lineage commitment. Single cells and their progeny were manually followed over time using the tTt software[24] (**Supplementary Figs. 1** and **2** and **Supplementary Video 1**). Next, we combined automated segmentation with cell tracks by linking image patches centered on the cell's center of mass to the nearest track coordinate. To that end, we extended our previously developed automated image processing pipeline that identifies and segments single cells with high accuracy in time-lapse brightfield microscopy[25] (**Supplementary Video 2**). For a detailed description of tracking, identification and segmentation, see **Supplementary Note 1** and **Supplementary Figure 10**.

Lineage commitment was initially annotated by experts by visual inspection of the fluorescence signal of CD16/32 (for the GM lineage) and GATA1-mCherry (for the MegE lineage). We amended these annotations by automatically quantifying the concentration of CD16/32 and GATA1-mCherry using a self-written user interface (see **Supplementary Fig. 11**). Galleries of cells with marker expression are shown in **Supplementary Figure 12**. For the genetically non-modified mice, we used CD16/32 (for the GM lineage) and a large morphology (for the MegE lineage due to the absence of GATA1-mCherry).

Deep neural networks. We combined a convolutional neural network (CNN) that extracts shape-based features, with a recurrent neural network (RNN) architecture that models the dynamics of the cells.

For the CNN, we extended a model from the LeNet family[26] by combining three convolutional layers (20 filters with kernel size 5, 60 filters with kernel size 4, and 100 filters with kernel size 3) with two fully connected layers (500 and 50 nodes). The last hidden layer that is fully connected (yellow nodes in **Fig. 1d**) can be interpreted as patch-specific features. Each convolutional layer is followed by a nonlinear activation function. We chose Rectified Linear Units (ReLU), which have been shown to introduce nonlinearities without suffering from the vanishing gradient problem[27]. In addition, we used max-pooling layers, which reduce variance and increase translational invariance by computing the maximum value of a feature over a region[28], and dropout layers following the fully connected layer to avoid over-fitting. Here, we largely follow Ciresan *et al.*[29], where it is shown that this combination of layers results in fast training times and good performance on a variety of image classification data sets. We further use the recently proposed batch normalization strategy to normalize outputs after each layer[30]. Finally, in order to be able to account for non-image-based features, we introduce a concatenation layer that combines spatial features with cell displacement (white nodes in **Fig. 1d**).

We used a softmax loss function and trained the network using stochastic gradient descend with stratified batches of 128 images. We initialized all weights in the network using the Xavier algorithm, which automatically determines the scale of the initialization based on the number of input and output nodes[31]. We used standard values for the base rate of learning (0.01), momentum (0.9) and the learning rate policy (stepwise policy decreasing the learning rate every 10,000 iterations[32]).

We then passed the output of the first layer that was fully connected together with the displacement feature (which we interpret as patch features extracted by the CNN) to a bidirectional long short-term memory (LSTM) recurrent neural network (**Fig. 1e**). Specifically, we trained a LSTM RNN with 20 hidden nodes using the rprop algorithm[33] with cross-entropy loss function, taking the mean across all time points. We trained the RNN for up to 15 epochs with standard positive update parameter of 1.2 and negative update parameter of 0.5.

In order to avoid over-fitting, we divided the training data into a training set and a validation set, and we optimized the weights of both the CNN and the LSTM RNN until the performance on the validation set started to degrade (early stopping). All images were normalized to mean zero and unit variance to normalize for possible batch effects.

Quantification of morphodynamics and fluorescence signals. On every extracted image patch (27 × 27 px around the cell's

doi:10.1038/nmeth.4182

center of mass), 87 features (14 basic measurements as provided by MATLAB regionprops method, 27 Zernike moments[34], 3 Ray features[35], 13 Haralick texture features[36], 2 Gabor wavelet features[37], 5 Tamura features[38] and histograms of oriented Gradients[39] with 27 bins) were computed. If a fluorescence image was available, the fluorescence concentration was quantified by summing up all pixels within the segmented cell in the background-corrected fluorescence image and dividing by cell size. Quantification errors (clumped cells, dirt, falsely identified as cell, cells lost due to border contact and over-segmented cell fragments) were detected by fitting a B-spline to the cell size over time. We then discarded those time points with residue differences beyond the 98th or below the 2nd percentile of all time points. We computed cell displacement $s_{i,t}$ for cell i at time point t as the root mean squared displacement between the frame t and $t-1$ divided by the time difference between the frames,

$$s(x,y)_{i,t} = \frac{\sqrt{(x_{i,t} - x_{i,t-1})^2 + (y_{i,t} - y_{i,t-1})^2}}{T_t - T_{t-1}},$$

where x and y are the spatial coordinates, and T_t is the absolute time after experiment start for frame t. We computed $s(x,y)$ for all pairs of adjacent frames for every cell using the track coordinates for the full cell trajectory.

Feature-based classification. A random forest classifier was trained with 200 trees (default parameters) and evaluated by out-of-bag prediction. We chose BudgetedSVMs[40] with the Pegasos algorithm and a radial basis function kernel as a support vector machine (SVM) framework that was able to deal with the millions of single image patches in our data set. We used a grid search with fivefold cross-validation for every train-test combination to determine optimal hyperparameters. The best-performing model was then used for predicting the testset. Note that the hyperparameters for SVM had to be determined for every train-test run individually. We trained both methods with a set of 87 morphological features and cell displacement (see above). We applied the same train-test procedure for model evaluation as for the CNN.

AITP was trained as described[15], using a set of 6 features for each cell (movement, net movement, movement direction, area and eccentricity of fitted convex hull). As AITP was not able to process the full data set in a single run, we generated three subsets (n = 400 cells per subset) for every train-test round, which we evaluated separately. We used the averaged evaluation results for comparison. As the used version of AITP reported class labels and no prediction scores, we used the macro-averaged F_1 score for performance evaluation. It is worth noting that in contrast to all other methods, AITP inherently uses the full cell trajectory for training and prediction.

Recently, conditional random field (CRF) based models have been proposed for sequence labeling in the context of mitosis sequence detection[13,14]. We adapted the approach of Liu *et al.*[13] and trained a CRF using SIFT features for our cells[41]. The CRF was implemented using the pystruct library[42].

Evaluation of model performance. Performance of the trained classification models was determined by receiver-operator characteristics and macro-averaged F_1 score.

The receiver-operator characteristic is a function that evaluates the change in true positive (TP) rate with respect to the false positive (FP) rate of a predicted class label in accordance to all possible thresholds of a classification score that can be interpreted as probabilities (as is the case for random forest, SVM and CNN). The area under the curve (AUC) falls in the interval of [0,1] (1 = perfect classification, 0.5 = random guessing) and gives an impression of the general performance of the classifier.

The F_1 score combines precision and recall in a single score

$$F_1 = 2 * \frac{TP}{2TP + FP + FN}, F_1 \in [0,1],$$

where FN is the false negative rate. A perfect classifier would reach a score of 1 and a random classifier would reach a score of 0.5. To account for the classification performance of both classes, the F_1 score can be calculated for each class and then averaged resulting in the macro-averaged F_1 score

$$\text{macro}_{F1} = \frac{1}{|C|} \sum_{i=1}^{|C|} F_1(C_i),$$

for all class labels C. To determine class membership (i.e., commitment of a cell to GM or MegE lineage) a threshold of 0.5 for the cell lineage score was used for all models.

Implementation. Single-cell identification and quantification was implemented using MATLAB (R2014a). Code from ref. 43. was used to compute histograms of oriented gradients. All quantifications were parallelized on single-cell level and processed on a computation cluster (sun grid engine version 6.2u5). The average node architecture was equal to an Intel Xeon 2 GHz, 4 GB RAM running a 64-bit linux-based operating system. Random forest classification was conducted with the python-based scikit-learn package (v 0.15). The support vector machine was trained using the code provided with the original publication[40]. AITP was trained using the latest version (April 1st, 2014) from the website of the authors after slight adaptation of input/output functionality to fit our data. To implement the CNN, we used the caffe framework[32] and trained it on a standard PC equipped with an Intel Core i7-4770 CPU, 32 GB working memory and a 6 GB Geforce GTX Titan Black graphics card. The RNN was implemented in Theano[44] and trained on that same machine. SIFT features were calculated using VLFeat[41], and the CRF was implemented using the pystruct library[42].

Data availability. Data and code for cell detection and neural network training and cell fate prediction is available as **Supplementary Software**; updated versions are available via https://github.com/QSCD/HematoFatePrediction. Source data files for **Figures 1** and **2** are available online.

21. Osawa, M., Hanada, K.-I., Hamada, H. & Nakauchi, H. *Science* **273**, 242–245 (1996).
22. Kiel, M.J. *et al. Cell* **121**, 1109–1121 (2005).
23. Selinummi, J. *et al. PLoS One* **4**, e7497 (2009).
24. Hilsenbeck, O. *et al. Nat. Biotechnol.* **34**, 703–706 (2016).

25. Buggenthin, F. *et al. BMC Bioinformatics* **14**, 297 (2013).
26. Lecun, Y., Bottou, L., Bengio, Y. & Haffner, P. *Proc. IEEE* **86**, 2278–2324 (1998).
27. Nair, V. & Hinton, G.E. in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)* 807–814 (ICML, 2010).
28. Ranzato, M., Huang, F.J., Boureau, Y.-L. & LeCun, Y. in *IEEE Conference on Computer Vision and Pattern Recognition, 2007 (CVPR '07)* 1–8 (IEEE, 2007).
29. Ciresan, D.C., Meier, U., Masci, J., Gambardella, L.M. & Schmidhuber, J. in *IJCAI Proceedings–International Joint Conference on Artificial Intelligence* **22**, 1237 (2011).
30. Ioffe, S. & Szegedy, C. Batch normalization: accelerating deep network training by reducing internal covariate shift. Preprint at https://arxiv.org/abs/1502.03167 (2015).
31. Glorot, X. & Bengio, Y. in *International Conference on Artificial Intelligence and Statistics,* 249–256 (2010).
32. Jia, Y. *et al.* in *Proceedings of the 22nd ACM International Conference on Multimedia* 675–678 (ACM, 2014).
33. Braun, H. & Riedmiller, M. in *Proceedings of the International Symposium on Computer and Information Science VII* (1992).
34. Zernike, F. *Physica* **1**, 689–704 (1934).
35. Smith, K., Carleton, A. & Lepetit, V. in *Proceedings of the International Conference on Computer Vision (ICCV)* (2009).
36. Haralick, R.M. *IEEE Trans. Syst. Man Cybern.* **SMC-3**, 610–621 (1973).
37. Gabor, D. *J. Instrum.* **93**, 429–441 (1946).
38. Tamura, H., Mori, S. & Yamawaki, T. *IEEE Trans. Syst. Man Cybern.* **8**, 460–473 (1978).
39. Dalal, N. & Triggs, W. in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR05)* **1**, 886–893 (IEEE, 2004).
40. Djuric, N., Lan, L., Vucetic, S. & Wang, Z. *J. Mach. Learn. Res.* **14**, 3813–3817 (2013).
41. Vedaldi, A. & Fulkerson, B. VLFeat: An Open and Portable Library of Computer Vision Algorithms. (2008).
42. Müller, A.C. & Behnke, S. *J. Mach. Learn. Res.* **15**, 2055–2060 (2014).
43. Junior, O.L., Delgado, D., Goncalves, V., Nunes, U. & Ludwig, O. in *2009 12th International IEEE Conference on Intelligent Transportation Systems* 1–6 (IEEE, 2009).
44. The Theano Development Team. *et al.* Theano: a Python framework for fast computation of mathematical expressions. Preprint available at https://arxiv.org/abs/1605.02688 (2016).