

Deep learning enables fast and dense single-molecule localization with high accuracy

Artur Speiser^{1,2,3,4,12}, Lucas-Raphael Müller^{5,6,12}, Philipp Hoess^{1D}⁵, Ulf Matti^{1D}⁵, Christopher J. Obara⁷, Wesley R. Legant^{8,9,10}, Anna Kreshuk^{1D}⁵, Jakob H. Macke^{1,2,3,11,13}✉, Jonas Ries^{1D}^{5,13}✉ and Srinivas C. Turaga^{1D}^{7,13}✉

Single-molecule localization microscopy (SMLM) has had remarkable success in imaging cellular structures with nanometer resolution, but standard analysis algorithms require sparse emitters, which limits imaging speed and labeling density. Here, we overcome this major limitation using deep learning. We developed DECODE (deep context dependent), a computational tool that can localize single emitters at high density in three dimensions with highest accuracy for a large range of imaging modalities and conditions. In a public software benchmark competition, it outperformed all other fitters on 12 out of 12 datasets when comparing both detection accuracy and localization error, often by a substantial margin. DECODE allowed us to acquire fast dynamic live-cell SMLM data with reduced light exposure and to image microtubules at ultra-high labeling density. Packaged for simple installation and use, DECODE will enable many laboratories to reduce imaging times and increase localization density in SMLM.

Single-molecule localization microscopy (SMLM) (for example, PALM¹ and (d)STORM^{2,3}) has become an invaluable super-resolution method for biology, as it can resolve cellular structures with nanometer precision. It is based on acquiring a large number of camera frames, in each of which only a tiny fraction of the emitters are stochastically activated into a bright ‘on’ state, so that their images do not overlap. This allows precise localization of the emitter coordinates by fitting a model of the point spread function (PSF). A super-resolution image is then reconstructed from these coordinates. This principle of SMLM is at the same time one of its main limitations: the need for sparse activation leads to long acquisition times. This results in low throughput, poor time resolution when imaging dynamic processes, low labeling densities and a reduced choice of fluorophores. Additionally, long acquisition times in combination with high excitation laser intensities needed for single-molecule imaging can cause strong phototoxicity in live-cell SMLM.

All of these limitations can be mitigated by activating emitters at a higher density. In this ‘multi-emitter’ setting, PSFs are no longer well-separated but may overlap, making both the detection of multiple nearby emitters and their accurate localization computationally challenging. This is not adequately addressed by existing algorithms: current ‘multi-emitter’ fitting algorithms^{4–6} work reasonably well on two-dimensional (2D) samples where all emitters have the same *z* coordinate and thus produce identical PSFs. These algorithms, however, have had limited success for realistic three-dimensional (3D) biological structures. In a software competition that benchmarked SMLM algorithms using realistic computer-generated data, simple single-emitter fitters outperformed dedicated high-density fitters on 3D samples even in the high-density regime⁷.

Deep learning is revolutionizing biological image analysis^{8–10}. For SMLM, deep learning holds promise to extract emitter coordinates and additional parameters under conditions and densities too complex for traditional fitters. With enough training data, deep networks are flexible function approximators that can be trained to recognize patterns in the image and thus transform images directly into predicted emitter configurations, even for challenging high densities of emitters. While ground-truth data to train the neural network are typically not available, synthetic training data can be generated by numerically simulating the imaging process^{11,12}. Convolutional neural networks (CNNs, a class of deep networks suitable for image data) have recently been used to extract parameters describing single isolated emitters such as color, emitter orientation, *z* coordinate, background or aberrations^{13–16} and to design optimized PSFs¹⁷. Two recent studies (DeepSTORM3D¹⁷ and DeepLoco¹⁸) used CNNs for extracting emitter coordinates, and outperformed traditional single-emitter fitting algorithms at densities higher than the single-molecule regime. These studies illustrate the potential of deep learning for SMLM, however, they have only been demonstrated either for exotic engineered PSFs or on simulated data.

Here we present the DECODE (deep context dependent) method for deep learning-based single-molecule localization that achieves high accuracy across a wide range of emitter densities and brightness levels. DECODE uses a deep network output representation, architecture, and cost function, which are optimized for simultaneous detection and subpixel localization of single emitters. Uniquely, DECODE is able to predict both the probability of detection and the uncertainty of localization for each emitter. As the timing and duration

¹Machine Learning in Science, Excellence Cluster Machine Learning, Tübingen University, Tübingen, Germany. ²Computational Neuroengineering, Department of Electrical and Computer Engineering, Technical University of Munich, Munich, Germany. ³Research Center Caesar, Max Planck Society, Bonn, Germany. ⁴International Max Planck Research School Brain and Behavior, Bonn, FL, USA. ⁵Cell Biology and Biophysics Unit, European Molecular Biology Laboratory, Heidelberg, Germany. ⁶Ruprecht Karls University of Heidelberg, Heidelberg, Germany. ⁷HHMI Janelia Research Campus, Ashburn, VA, USA. ⁸Joint Department of Biomedical Engineering, UNC, Chapel Hill, NC, USA. ⁹NCSU Raleigh, Raleigh, NC, USA. ¹⁰Department of Pharmacology, University of North Carolina, Chapel Hill, NC, USA. ¹¹Department Empirical Inference, Max Planck Institute for Intelligent Systems, Tübingen, Germany.

¹²These authors contributed equally: Artur Speiser, Lucas-Raphael Müller. ¹³These authors jointly directed this work: Jakob H. Macke, Jonas Ries, Srinivas C. Turaga. ✉e-mail: Jakob.Macke@uni-tuebingen.de; jonas.ries@embl.de; turagas@janelia.hhmi.org

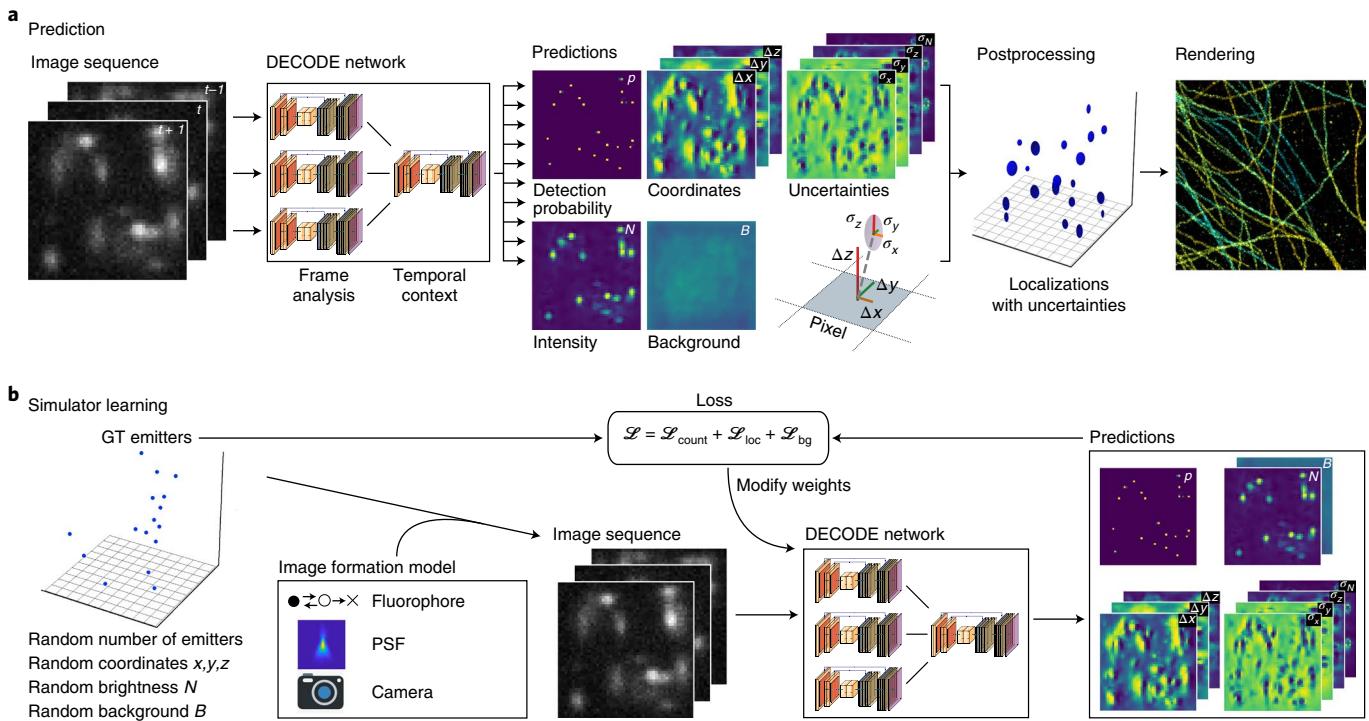


Fig. 1 | DECODE for high-density single-molecule localization. **a**, DECODE architecture. The DECODE network uses information from multiple frames to predict output maps representing for each pixel the probability of detecting an emitter and the emitter's subpixel spatial coordinates ($\Delta x, \Delta y, \Delta z$), brightness (N), the uncertainty related to those predictions ($\sigma_x, \sigma_y, \sigma_z$), and an optional background map (B). **b**, Training DECODE. The DECODE network is trained by simulator learning. Ground-truth (GT) emitter coordinates are generated randomly, and synthetic images simulated from a forward model of the image formation process are passed through the DECODE network. The loss quantifies the probability that the GT explains the output predictions and is optimized during training.

of emitter activations are stochastic, they regularly persist over several imaging frames. The DECODE architecture can integrate information across neighboring frames ('temporal context'), which improves emitter detection and localization.

In the public SMLM challenge⁷, DECODE outperformed all existing methods on 12 out of 12 datasets. Compared to previous deep learning-based high-density fitters¹⁷, DECODE is ten times faster and up to twice as accurate, and can be applied to a wide range of PSFs. We demonstrate on biological structures that DECODE allows for fivefold higher labeling densities or tenfold faster imaging compared to imaging in the single-emitter regime, and thus enables fast live-cell SMLM with reduced light exposure and visualization of dynamic processes. We show the versatility of DECODE by reanalyzing a published lattice light sheet (LLS) point accumulation for imaging of nanoscale topography labeling (PAINT) dataset¹⁹ for which we could substantially improve fluorophore detection and localization accuracy. DECODE is packaged for simple use and can be easily trained and used by nonexpert users, without having to design new network architectures. Thus, it will enable the entire community to overcome the need of sparse activation as one of the main bottlenecks in SMLM.

Results

DECODE network. DECODE introduces a new output representation and architecture for detecting and localizing emitters. For each image frame, it predicts multiple channels with the same dimensions as the input image (Fig. 1a). The first two channels indicate the probability p that an emitter exists near that pixel, as well as its brightness N (number of photons emitted by the emitter in the frame). The next three channels describe the coordinates of the emitter with respect to the center of the pixel,

$\Delta xyz = (\Delta x, \Delta y, \Delta z)$. An additional channel predicts the background intensity B in each pixel.

This architecture overcomes limitations of current deep learning^{17,18} and non-deep learning-based high-density approaches in three ways. First, DECODE predictions scale only with the number of imaged pixels (not super-resolution voxels as in DeepSTORM3D), resulting in over 20-fold improvement in prediction speeds and the use of continuous subpixel coordinates eliminates a voxel size dependent limit on precision. The local output representation used by DECODE also avoids the potentially challenging nonlocal mapping of pixels to global coordinates used in DeepLoco.

Second, DECODE has four additional output channels that estimate the uncertainty of the localization along each coordinate given by $\sigma_{xyz} = (\sigma_x, \sigma_y, \sigma_z)$ and of the brightness σ_N . These predicted localization uncertainties can be used to filter out poorly localized detections to improve the rendering of super-resolution images. In addition, training the network to additionally predict the localization uncertainty corresponding to each detection also helps to improve the quality of the detection probabilities p by implicitly grouping all the detections corresponding to the same emitter. In contrast, standard output representations that only indicate the probability of detecting an emitter on a per-voxel basis make it more challenging to correctly group detection probability voxels corresponding to the same emitter in high emitter-density and high localization-uncertainty scenarios.

Third, the DECODE network integrates information across multiple frames with a two-stage design. The first stage (frame analysis module) analyses single imaging frames using a 2D multi-resolution convolutional network based on the 'U-Net' architecture²⁰ to compute a feature representation of the single frame (Extended Data Fig. 1). The second stage (temporal context module) integrates the

feature representations of the frame with those of the previous and next imaging frame using a second 2D U-Net to produce the final predictions. As emitters persist over several frames, this improves detection and localization accuracy.

Training the DECODE network using simulator learning. We train DECODE to simultaneously detect and localize emitters in SMLM measurements. Ground-truth data for supervised learning are not easily available for SMLM. However, it is possible to simulate realistic images of activated emitters as the physics of imaging single molecules is well understood¹². We train the DECODE network by generating a large amount of simulated data. To avoid structural bias⁸, we place emitters at random coordinates, and calculate simulated images with a realistic image formation model that includes dye photophysics, a measured PSF and camera noise (Methods).

We trained the DECODE network to predict the probability of detection, along with the subpixel localization and localization uncertainty of each detected emitter. Our loss function has three terms: (1) a count loss that compares the true and detected number of emitters in the image; (2) a localization loss that trains the network to correctly localize the detected emitters and estimate the localization uncertainty and emitter brightness and (3) an optional background loss. The count and localization loss functions were derived together as an approximation to a spatial point process probability distribution. They work together to correctly train the DECODE network to predict one detection per emitter, and to correctly assign the localization uncertainty of each emitter to the corresponding detection. Together, they constitute a new loss for counting, detecting and localizing sets of discrete point-like objects.

The count loss first constructs a Gaussian approximation to the predicted number of emitters by summing the mean and the variance of the Bernoulli detection probability map, and then maximizes the probability of the true number of emitters under this distribution. Uncertain detections will lead to large predicted count variance, while confident detections will result in low variance. Thus, the count loss encourages a detection probability map with sparse but confident predictions. The localization loss models the distribution of subpixel localizations Δxyz with a coordinate-wise independent Gaussian probability distribution²¹ with standard deviation σ_{xyz} . For imprecise localizations, this probability is maximized for large σ_{xyz} , for precise localizations for small σ_{xyz} . The distribution of all localizations over the entire image is approximated as a weighted average of individual localization distributions, where the weights correspond to the probability of detection. By optimizing both the probability of detection, the subpixel localization Δxyz and σ_{xyz} simultaneously, the network learns not only the best predictions for the coordinates of the emitters, but also the best estimate for their localization uncertainties. The emitter brightness predictions N and their uncertainties σ_N are optimized similarly. Finally, the optional background loss computes the mean squared error between the true and predicted background images B . While the network only uses camera images to make predictions, the network training procedure does require PSF calibration measurements.

DECODE achieves high accuracy for a wide range of simulated data. *Performance metrics.* The quality of SMLM data analysis is commonly quantified by two factors: first, the detection accuracy quantifies the fraction of emitters that are detected. The metric we use here is the Jaccard Index (JI)⁷, that sets the true positives (TP) in relation to the false positives (FP) and false negatives (FN), $JI = TP / (TP + FN + FP)$. The second factor is the localization error, that is how close the measured coordinates are to the true coordinates, measured here as the root mean squared error (r.m.s.e.) averaged over the dimensions (Methods). We matched the detected emitters to the ground-truth emitters in three dimensions with a lateral threshold of 250 nm and an axial threshold of 500 nm.

There is a natural trade-off between JI and localization error: discarding all but the brightest and best separated emitters will result in a good (low) localization error but a bad (low) JI. Conversely, including also poorly localized emitters might improve JI, but deteriorates the localization error. The optimal operating point between these two extremes will depend on the experimental conditions and the scientific question. Because DECODE also provides uncertainties for each localization, it offers a straightforward way to filter localizations and thus set the desired balance between the number of detected emitters and the localization error that can be tolerated.

The Cramér–Rao lower bound (CRLB) gives the minimum achievable localization error for an optimal fitter given a known PSF, background and noise model²². Most commonly, it is calculated under idealized conditions (that is, nonoverlapping PSFs, homogeneous background, assuming the chosen PSF model to be the true model) and we use it here for comparison as a best-case limit for localization error.

DECODE approaches the CRLB for low densities. We simulated 100,000 frames with exactly one emitter per frame at random coordinates with a constant brightness and background, and trained DECODE without temporal context. On these data with sparse activations, DECODE approaches the single-emitter CRLB, that is the theoretical limit of precision (Fig. 2a). It thus performs as well as maximum likelihood estimation (MLE) based fitters, which have also been shown to reach the CRLB²³ in this regime.

DECODE’s uncertainty estimates are well calibrated. In the high-density regime, DECODE’s σ predictions correlate closely to the measured localization error (Fig. 2b), much better than the single-emitter CRLB estimate that assumes isolated emitters (correlation coefficient 0.86 for σ versus 0.07 for single-emitter CRLB). For the low-density regime, the uncertainty estimates are in line with the measured error and the single-emitter CRLB (Fig. 2a).

Temporal context improves localization error and detection. DECODE’s temporal context module pools information across multiple (we used three) frames, to model the fact that emitters can persist in multiple subsequent frames. Use of this context module improves both the detection accuracy (JI) and the localization error (Fig. 2c). The increase in JI is apparent for all densities and signal to noise ratios (SNRs). In addition, the r.m.s.e. is reduced by up to 20 nm. Overall, the temporal context has a large impact across imaging conditions, and is also more powerful than ‘grouping’ approaches that are often applied to localizations in a postprocessing step (Extended Data Fig. 2).

DECODE architecture outperforms a voxel-based network architecture and a multi-emitter fitter. To assess how the DECODE network architecture performs against other deep learning-based and iterative methods, we directly compared to DeepSTORM3D¹⁷ and CSpline⁴, a matching pursuit style multi-emitter fitter based on MLE, using the code provided by the authors. To minimize the risk of suboptimal training, we trained DeepSTORM3D on data sampled from our generative model using the same parameters we used for the training of DECODE. For both DeepSTORM3D and CSpline we performed a parameter grid search over user-defined parameters to maximize their performance (measured as efficiency score⁷). To facilitate the comparison of localization precision, we filtered out DECODE localizations with the highest inferred uncertainties such that the remaining number match DeepSTORM3D. DECODE outperforms the other methods across all densities and SNRs (Fig. 2e and Extended Data Fig. 3) even without temporal context. When we use temporal context, DECODE reduces the localization error up to twofold compared to DeepSTORM3D. Although both methods are based on deep learning, this performance improvement is

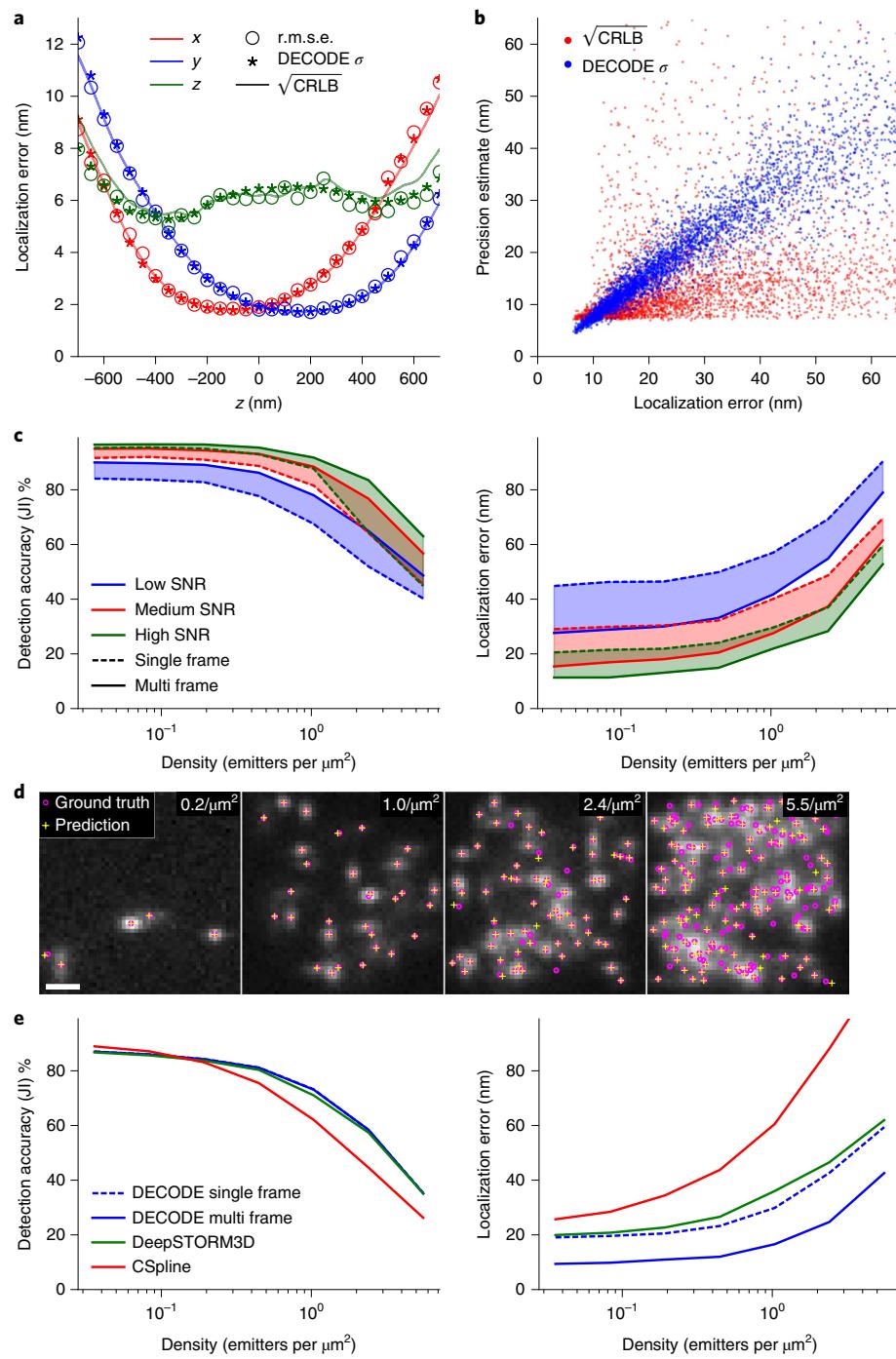


Fig. 2 | Performance of DECODE on simulated data. **a**, DECODE reaches the single-emitter CRLB for isolated emitters. The r.m.s.e. and DECODE σ averaged over 50 nm bins (additional comparisons in Extended Data Fig. 4). **b**, Comparison of the predicted localization uncertainty, σ , and measured localization error for densely activated emitters. We simulated the same dense emitter configuration 100 times and calculated the measured localization error as the r.m.s.e. of the predictions of the coordinates. Also shown is the (square root of the) single-emitter CRLB. See Supplementary Fig. 1 for comparisons of individual axes. **c**, Impact of temporal context on detection performance and localization error. Detection accuracy and localization error of DECODE trained with (multiframe) and without (single frame) temporal context quantified as a function of emitter density on simulations with low, medium and high SNR. **d**, Representative simulated frames with ground-truth coordinates (magenta circles) and predicted coordinates (yellow crosses) for the densities used in **c** and medium SNR. **e**, Comparison of DECODE with CSpline and DeepSTORM3D over a wide range of densities. See Extended Data Figs. 2 and 3 for additional comparisons with different conditions and metrics. The standard error of the mean (s.e.m.) on the localization error lies between 0.2 and 0.4 nm. See Methods and Supplementary Table 1 for additional details on training and evaluation.

based on the differences in output representation and loss function between DECODE and DeepSTORM3D. The localization error of DeepSTORM3D is limited by the super-resolution voxel size¹⁷

(Extended Data Fig. 4), which prevents the method from achieving the single-emitter CRLB, unlike DECODE that has no such limitation. Because DECODE has multiple output maps it is also able to

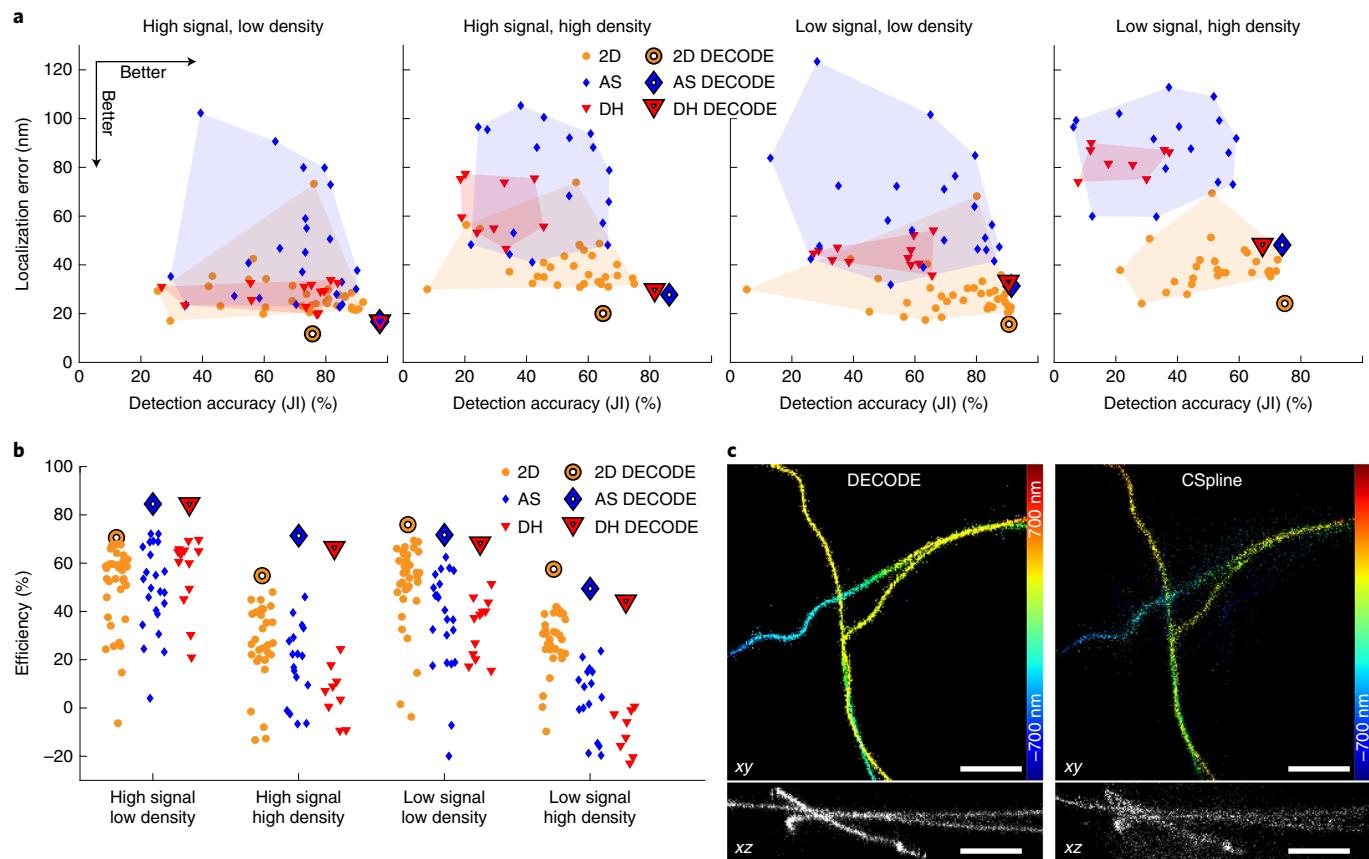


Fig. 3 | Performance comparison on the SMLM 2016 challenge. **a**, Performance evaluation on the 12 test datasets with low or high density, low or high SNR and different modalities (2D, AS, astigmatic; DH, double helix) using the detection accuracy (JI, higher is better) and localization error (lower is better) as metrics. Each marker indicates a benchmarked algorithm, large markers indicate DECODE. **b**, Efficiency scores (higher is better) where each marker indicates performance for one method. Metrics were calculated by the SMLM 2016 challenge and downloaded from the challenge website (<http://bigwww.epfl.ch/smlm/challenge2016/leaderboard.html>). **c**, Reconstructions by DECODE and the CSpline algorithm on the high density, low signal double-helix challenge training data. Upper panels show the xy view, color coded by the z coordinate and the lower panels show the xz reconstructions. Scale bars, 1 μm. See Supplementary Figs. 5 and 6 for additional comparisons with DeepSTORM3D on training datasets.

provide accurate estimates of the signal photon counts and background values (Extended Data Fig. 9).

Notably, DECODE performs favorably in fitting time (Extended Data Fig. 6), taking less than 1.5 s to analyze 1,000 frames of 64 × 64 pixels, while DeepSTORM3D requires between 34 and 54 s and CSpline requires between 14 and 2,680 s, which is up to 1,900-fold slower than DECODE. Training the DECODE network to convergence on a NVIDIA RTX2080Ti graphical processing unit (GPU) requires around 10 h while DeepSTORM3D takes around 50 h.

DECODE outperforms all fitters on a public SMLM benchmark. The 2016 SMLM challenge is an on-going and continuously updated second generation comprehensive benchmark evaluation developed for the objective, quantitative evaluations of the plethora of available localization algorithms^{7,24}. It offers synthetic datasets for training, created to emulate various experimental conditions. To avoid overfitting, evaluations are carried out on data not shared with contestants. It calculates various quality metrics, among them r.m.s.e. lateral or volume localization error, as applicable for 2D and 3D data, respectively, the JI quantifying detection accuracy and a single ‘efficiency’ score that combines r.m.s.e. and JI. The performance of DECODE in the SMLM 2016 challenge, including extensive evaluations and side by side comparisons, is available online (<http://bigwww.epfl.ch/smlm/challenge2016/leaderboard.html>). DECODE outperformed

all 39 algorithms on 12 out of 12 datasets, often by a substantial margin (Fig. 3, data from challenge website, current as of 1 October, 2020). The datasets included high (N1) and low (N2, N3) SNRs, with low or high emitter densities, with 2D, astigmatism and double-helix PSF-based imaging modalities.

DECODE achieves an average efficiency score of 66.6% out of the best possible score of 100% (achievable only by a hypothetical algorithm that accurately detects 100% emitters with 0 nm localization error). This is compared to an average score of 48.3 and 45.6% for all second and third place algorithms, respectively. The difference is particularly large under difficult imaging conditions, when high emitter densities and low SNR can conspire to make detection and localization challenging, particularly so for the double-helix PSF. For example, compared to the second-best algorithm (SMAP2018) in the low-SNR/high-density/double-helix condition, DECODE improves the localization error from 75.2 to 48.4 nm and the JI from 30.0 to 67.5%.

DECODE enhances super-resolution reconstructions by improving both the detection and the localization of single molecules. An example of this can be seen in Fig. 3c, where we compare the reconstruction obtained with DECODE and CSpline⁴ on a high-density 3D double-helix dataset (using settings provided by the authors, github.com/ZhuangLab/storm-analysis). Other deep learning-based approaches have not yet submitted their results. However, we performed comparisons to DeepSTORM3D on

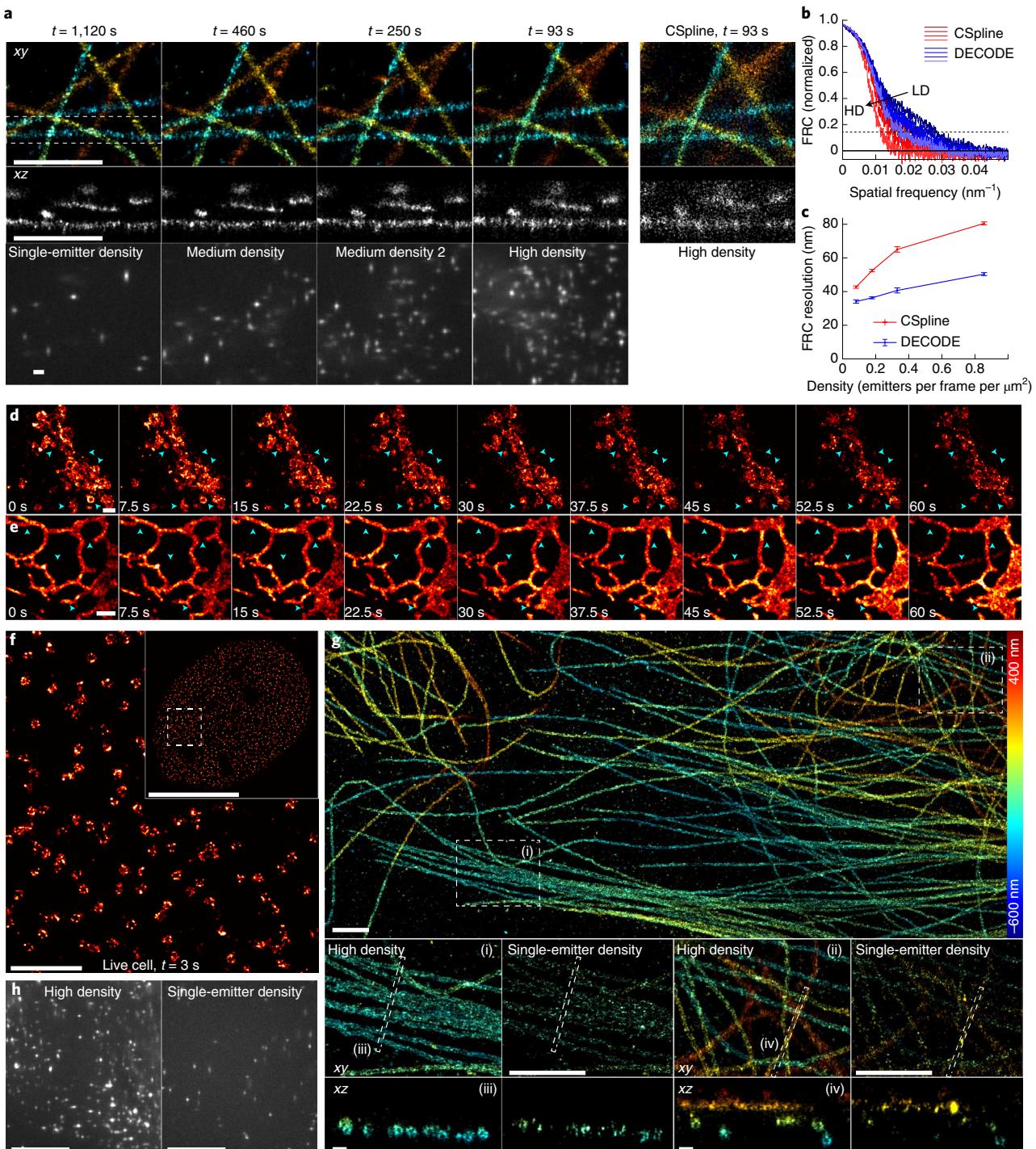


Fig. 4 | DECODE enables high-speed and live-cell SMLM and ultra-high labeling densities. **a**, DECODE can reduce acquisition times by one order of magnitude. The same sample of microtubules, labeled with anti- α -tubulin primary and AF647 secondary antibodies, imaged with different ultraviolet activation intensities to result in different emitter densities per frame, between 0.08 and $0.86 \mu\text{m}^{-2}$ and acquisition times between 93 and $1,120 \text{ s}$, while keeping the total number of localizations the same. For high-density activation, we show a comparison with CSpline. **b**, Fourier ring correlation (FRC) curves for DECODE and CSpline for different emitter densities. HD, high density; LD, low density. **c**, Resolution estimates obtained using the FRC and 0.143 criterion across densities for both methods. **d**, Fast live-cell SMLM on the Golgi apparatus labeled with α -mannosidase II-mEos3.2 (Supplementary Video 1). **e**, Fast live-cell SMLM on the endoplasmic reticulum labeled with calnexin-mEos3.2 (Supplementary Video 2 and Supplementary Fig. 3). **f**, Fast live-cell SMLM on the nuclear pore complex protein Nup96-mMaple acquired in 3 s . **g**, DECODE enables ultra-high labeling densities. Microtubules labeled with a high concentration of anti- α and anti- β -tubulin primary and Alexa Fluor 647 secondary antibodies. **g(i),(ii)**, Magnified regions as indicated in **g**. Data acquired with high-density labeling show continuous structures. As a comparison, the same sample was acquired after prebleaching of the fluorophores to reach the single-molecule blinking regime. Here, single labels are resolved in the super-resolution reconstruction and lead to a sparse decoration of the microtubules. **g(iii),(iv)**, Side view reconstructions of regions as indicated in **g(i),(ii)** resolving the hollow, cylinder-like structure of immunolabeled microtubules. **h**, Representative raw camera frames for the high-density and single-emitter acquisitions, respectively. Scale bars, $10 \mu\text{m}$ (**f** inset, **h**), $1 \mu\text{m}$ (**a,d,e,f,g(i),(ii)**) and 100 nm (**g(iii),(iv)**).

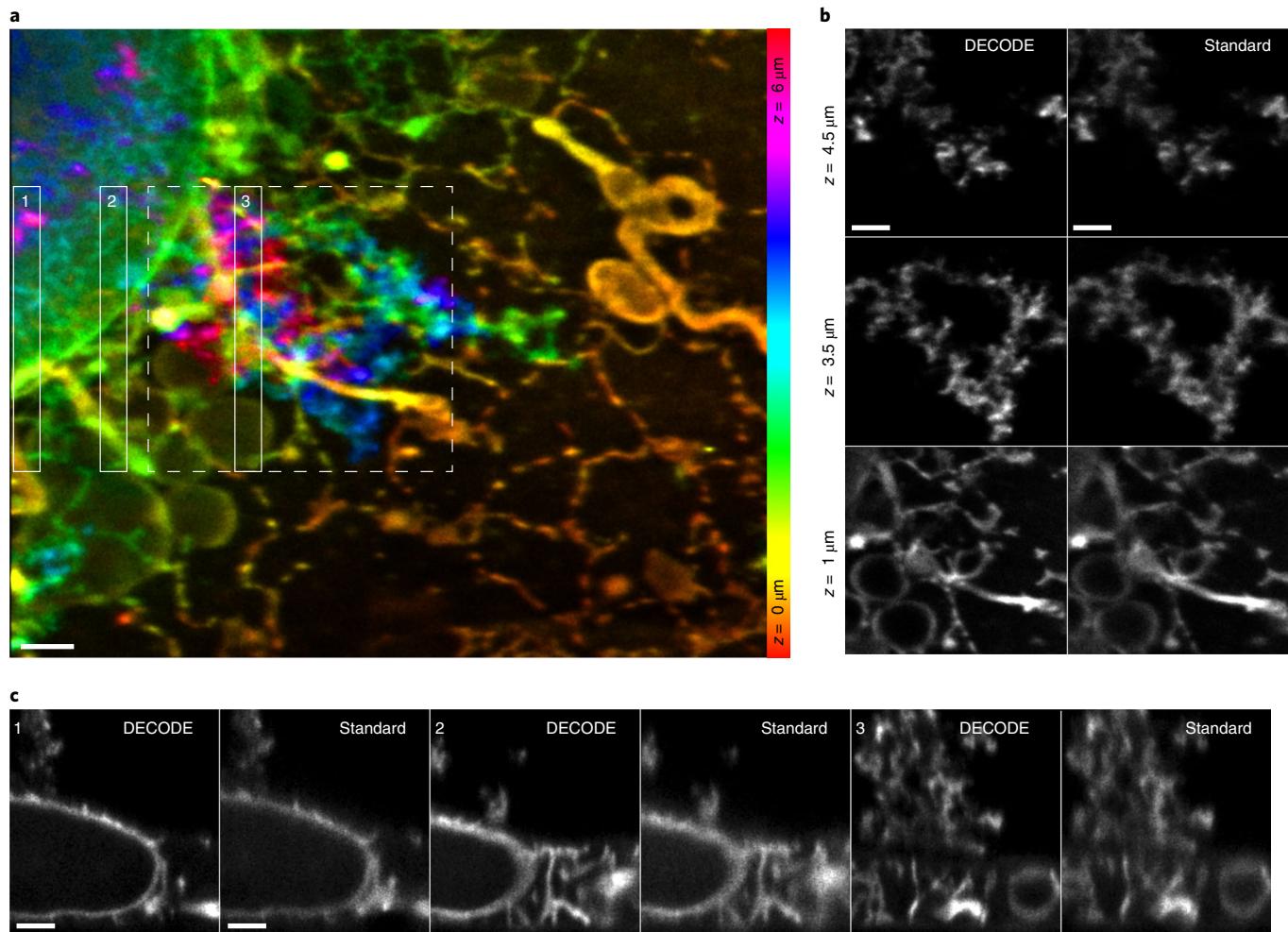


Fig. 5 | DECODE improves resolution in LLS PAINT. **a**, COS-7 cell imaged with LLS PAINT microscopy, overview. Data from Legant et al.³⁰, 70,000 volumes imaged over 2.7 days. **b**, 500-nm-thick slices of the region indicated in **a** (dashed line), comparing DECODE analysis and the original analysis using MLE fitting (standard analysis). **c**, Perpendicular (side view) reconstructions of 500-nm-thick regions as indicated in **a** comparing DECODE and standard analysis. Scale bars, 1 μm. See Extended Data Fig. 7 for additional comparisons.

low-SNR high-density training datasets and again achieved superior results (efficiency score of 51 against 32% on double helix and 45 against 31% on astigmatism data, Supplementary Figs. 5 and 6). Thus, DECODE is setting new quantitative standards for localization algorithms, across both low and high SNRs and densities.

Considerations. As with any fitter, DECODE relies on an accurate PSF model and proper parameters, otherwise artifacts will dominate the predictions. When the localization uncertainty is large, for very dim and dense localizations far from the focal plane, DECODE has a bias toward predicting localizations close to the pixel center. This effect can be overcome by filtering out localizations with large predicted uncertainty, or by rendering every localization with a Gaussian the size of the localization uncertainty, effectively dispersing these large uncertainty localizations over the pixel (Extended Data Fig. 8 and Methods).

DECODE reduces imaging times by one order of magnitude. By enabling accurate emitter localization at high densities of more than $2.5 \mu\text{m}^{-2}$ per frame (Fig. 2c), DECODE can yield high-quality super-resolution reconstructions with much shorter imaging times. We demonstrate this by imaging and reconstructing the same sample of labeled microtubules at four different activation laser

powers using STORM (stochastic optical reconstruction microscopy)^{2,3}. This results in different emitter densities per frame between 0.08 and $0.86 \mu\text{m}^{-2}$. The imaging time was chosen to result in the same number of total localizations and decreased from 1,120 to 460 and 250 and 93 s for stronger activation.

We trained and applied one common DECODE model to all four datasets (Fig. 4a). Whereas CSpline reconstructions quickly degrade with high emitter densities, DECODE consistently yields reconstructions with high accuracy even for the densest sample. We quantified the lateral resolution using Fourier Ring Correlation (FRC)²⁵, which estimates resolution by measuring the correlation of two different reconstructions of the same image across spatial frequencies. DECODE consistently improves the x,y resolution by 20–30 nm over CSpline across all imaging densities (Fig. 4b,c) while detecting around 30% more localizations.

DECODE enables fast live-cell SMLM with reduced light exposure. Fast imaging is especially relevant for live-cell SMLM where the dynamics of the biological system under investigation dictate the necessary time resolution. At the same time, fast imaging usually requires high laser powers, deteriorates resolution²⁶ and leads to substantial phototoxicity²⁷. As DECODE allows activating emitters to high density, it enables faster imaging with decreased light dose

for a given number of localizations. We were able to image dynamic changes of the Golgi apparatus (Fig. 4d) and the endoplasmic reticulum (Fig. 4e) with 7.5 s temporal resolution. We imaged nuclear pore complexes in living cells²⁸ within only 3 s (Fig. 4f), seven times faster than our previous speed-optimized live-cell SMLM²⁶ and with a 70% reduced light dose.

DECODE enables ultra-high labeling densities. Labeling densities in SMLM are fundamentally limited by the fraction of emitters that are in the bright state. For the best performing fluorophore Alexa Fluor 647, even without ultraviolet activation about 0.05% of the emitters are in the bright state²⁹ due to activation by the red imaging laser and spontaneous activation. For the single-emitter blinking regime (activated emitter density $<0.1 \mu\text{m}^{-2}$), this limits the number of total emitters to about $200 \mu\text{m}^{-2}$. For higher labeling, prebleaching can be used to reduce the number of emitters to this regime, but the resulting low labeling limits the resolution¹⁹ and in the super-resolution reconstructions sparse individual emitters become dominant (Fig. 4g). With DECODE, we can now image densely labeled samples that previously were inaccessible. We demonstrated this on immunolabeled microtubules that were labeled about five-fold higher than compatible with single-emitter fitting, resulting in much smoother and denser decoration of the microtubules (Fig. 4g). In 50-nm-thick orthogonal reconstructions, only the densely labeled microtubules were resolved as hollow cylinders, whereas after prebleaching to single-emitter blinking, these reconstructions only showed individual emitters (Fig. 4g(iii)(iv)). Additional comparisons with DeepSTORM3D highlight that the superior output representation and loss function of DECODE are critical to reach the optimal resolution for this dataset (Extended Data Fig. 5).

DECODE enables high fidelity reconstructions of 3D LLS PAINT. To illustrate the general applicability of DECODE, we applied it to 3D LLS microscopy combined with the PAINT technique¹⁹. In PAINT microscopy, the fluorophore labeling a sample stochastically binds and unbinds from the sample, providing dense labeling. In LLS microscopy, thick volumes are imaged at high resolution by scanning a thin ($1.1 \mu\text{m}$) light sheet, with axial localization within the sheet enabled by astigmatism.

Single-molecule localization in LLS PAINT is usually performed frame-wise using MLE fitting³⁰. However, an emitter is visible in several adjacent z planes in the volumetric dataset. Thus, similar to exploiting the temporal context, we now use the same spatio-temporal context by analyzing three adjacent frames in the z stack at the same time to improve detection accuracy and localization error.

We reconstructed a previously reported dataset of a chemically fixed COS-7 cell with intracellular membranes labeled by azepanyl-rhodamine (AzepRh)^{19,30} consisting of 70,000 3D volumes comprising more than 10 million 2D images acquired in 270-nm steps. DECODE detected 500 million emitters, compared to 200 million emitters detected by the original algorithm. Thus, for a comparable quality of the reconstruction, only half of the frames are needed, reducing imaging times by over a day from 2.7 to 1.35 days (Extended Data Fig. 7). At the same time, improved accuracy of DECODE results in sharper reconstructions (Fig. 5).

Discussion

We presented DECODE, a new deep learning-based method for single-molecule localization that performs exceptionally well on dense 3D data. DECODE differs from traditional localization algorithms by simultaneously performing detection and localization of emitters. It can be used in a flexible and general manner for a wide range of imaging parameters (including arbitrary PSFs and noise models) and imaging modalities such as 3D LLS PAINT imaging. In a publicly available benchmark challenge, it is the best performing algorithm in every condition, and often improves both localization

and detection accuracy by a large margin. By making use of the temporal context, DECODE improves detection accuracy and localization error of emitters that are active across multiple imaging frames. Temporal context is also used by postprocessing steps in SMLM relying on ‘merging’ or ‘grouping’ of localizations, in which localizations occurring in consecutive images that are closer to each other than a fixed threshold are assumed to belong to the same emitter and their coordinates are averaged, weighted by the uncertainty of each localization. However, grouping does not improve detection of emitters, and it fails for dense or dim emitters whose localizations cannot be linked unambiguously across frames.

DECODE not only predicts coordinates of emitters, but also their uncertainty. This is highly useful for filtering out imprecise localizations, for reconstruction of super-resolution images in which every localization is rendered as a Gaussian with a size proportional to the coordinate uncertainty and as weights for quantitative coordinate-based analysis of SMLM data.

We demonstrated the performance of DECODE on various experimental SMLM datasets. We could show that the excellent performance on high-density data can increase the achievable localization density or decrease imaging times by one order of magnitude. This allowed us to perform live-cell measurements on nuclear pore complexes with high temporal resolution and reduced light exposure, and to achieve ultra-high labeling on microtubules. LLS PAINT data analyzed with DECODE showed markedly improved resolution due to substantial improvements in emitter detection and localization error.

Prediction of coordinates with DECODE can be as fast as GPU-based MLE fitters for sparse activation, but greatly outperforms those for high densities, as the computational complexity of DECODE depends only on the size of the image and not the number of emitters in each imaging frame. However, it requires the training of a new neural network whenever the optical properties of the microscope change. This training can currently take over 10 h on a single GPU, but after just 2 h of training time, the localization error is within 1 nm and the JI within 2% of the final value (Extended Data Fig. 10). To reduce training times further, one can likely take an existing network and fine-tune its parameters using a smaller number of simulations, rather than training it from scratch. Ultimately, it may be possible to train a single network across multiple parameter settings or even PSFs, so that the same network can ‘amortize’ inference across multiple experimental settings. To make DECODE easily usable by the entire community, we distribute it as a Python-based open-source software package based on the PyTorch³¹ deep learning library. We provide precompiled, easily installable code, along with detailed tutorials and integration into the SMAP SMLM analysis software³². To enable anyone to directly use DECODE for training and prediction without relying on previous programming knowledge and dedicated local hardware, we deploy these Jupyter notebooks in Google Colab, complementing a recent initiative to make deep learning-based image analysis tools accessible to nonexperts at minimal cost³³. Thus, DECODE will enable a large community to directly perform SMLM in a new high-density regime with greatly increased imaging speeds or localization densities and excellent localization and detection accuracy.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at <https://doi.org/10.1038/s41592-021-01236-x>.

Received: 23 October 2020; Accepted: 7 July 2021;
Published online: 3 September 2021

References

1. Betzig, E. et al. Imaging intracellular fluorescent proteins at nanometer resolution. *Science* **313**, 1642–1645 (2006).
2. Rust, M. J., Bates, M. & Zhuang, X. Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM). *Nat. Methods* **3**, 793–796 (2006).
3. Van de Linde, S. et al. Direct stochastic optical reconstruction microscopy with standard fluorescent probes. *Nat. Protocols* **6**, 991–1009 (2011).
4. Babcock, H. P. & Zhuang, X. Analyzing single molecule localization microscopy data using cubic splines. *Sci. Rep.* **7**, 552 (2017).
5. Babcock, H., Sigal, Y. M. & Zhuang, X. A high-density 3d localization algorithm for stochastic optical reconstruction microscopy. *Opt. Nanoscopy* **1**, 6 (2012).
6. Ovesny, M., Krizek, P., Borkovec, J., Svindrych, Z. & Hagen, G. M. Thunderstorm: a comprehensive ImageJ plug-in for palrn and storm data analysis and super-resolution imaging. *Bioinformatics* **30**, 2389–2390 (2014).
7. Sage, D. Super-resolution fight club: assessment of 2D and 3D single-molecule localization microscopy software. *Nat. Methods* **16**, 387–395 (2019).
8. Belthangady, C. & Royer, L. A. Applications, promises, and pitfalls of deep learning for fluorescence image reconstruction. *Nat. Methods* **16**, 1215–1225 (2019).
9. Ching, T. Opportunities and obstacles for deep learning in biology and medicine. *J. R. Soc. Interface* **15**, 20170387 (2018).
10. Weigert, M. Content-aware image restoration: pushing the limits of fluorescence microscopy. *Nat. Methods* **15**, 1090 (2018).
11. Le, T. A., Baydin, A. G., Zinkov, R., and Wood, F. Using synthetic data to train neural networks is model-based reasoning. In *Proc. International Joint Conference on Neural Networks (IJCNN)* 3514–3521 (IEEE, 2017).
12. Möckl, L., Roy, A. R. & Moerner, W. E. Deep learning in single-molecule microscopy: fundamentals, caveats, and recent developments. *Biomed. Opt. Express* **11**, 1633–1661 (2020).
13. Zhang, P. et al. Analyzing complex single-molecule emission patterns with deep learning. *Nat. Methods* **15**, 913–916 (2018).
14. Kim, T., Moon, S. & Xu, K. Information-rich localization microscopy through machine learning. *Nat. Commun.* **10**, 996 (2019).
15. Möckl, L., Roy, A. R., Petrov, P. N. & Moerner, W. E. Accurate and rapid background estimation in single-molecule localization microscopy using the deep neural network bgnet. *Proc. Natl Acad. Sci. USA* **117**, 60–67 (2020).
16. Zelger, P. et al. Three-dimensional localization microscopy using deep learning. *Opt. Express* **26**, 33166–33179 (2018).
17. Nehme, E. et al. DeepSTORM3D: dense 3D localization microscopy and PSF design by deep learning. *Nat. Methods* **17**, 734–740 (2020).
18. Boyd, N., Jonas, E., Babcock, H. P. & Recht, B. Deeploco: fast 3D localization microscopy using neural networks. Preprint at *bioRxiv* <https://doi.org/10.1101/267096> (2018).
19. Chen, B.-C. Lattice light-sheet microscopy: imaging molecules to embryos at high spatiotemporal resolution. *Science* **346**, 1257998 (2014).
20. Ronneberger, O., Fischer, P. & Brox, T. U-net: convolutional networks for biomedical image segmentation. In *Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention* (eds Navab, N. et al.) 234–241 (Springer, 2015); https://doi.org/10.1007/978-3-319-24574-4_28
21. Rieger, B. & Stallinga, S. The lateral and axial localization uncertainty in super-resolution light microscopy. *Chem. Phys. Chem.* **15**, 664–670 (2014).
22. Chao, J., Ward, E. S. & Ober, R. J. Fisher information theory for parameter estimation in single molecule microscopy: tutorial. *JOSA A* **33**, B36–B57 (2016).
23. Li, Y. et al. Real-time 3D single-molecule localization using experimental point spread functions. *Nat. Methods* **15**, 367–369 (2018).
24. Small, A. & Stahlheber, S. Fluorophore localization algorithms for super-resolution microscopy. *Nat. Methods* **11**, 267–279 (2014).
25. PJ. Nieuwenhuizen, R. et al. Measuring image resolution in optical nanoscopy. *Nat. Methods* **10**, 557–562 (2013).
26. Diekmann, R. et al. Optimizing imaging speed and excitation intensity for single-molecule localization microscopy. *Nat. Methods* **17**, 909–912 (2020).
27. Wälchen, S., Lehmann, J., Klein, T., Van De Linde, S. & Sauer, M. Light-induced cell damage in live-cell super-resolution microscopy. *Sci. Rep.* **5**, 15348 (2015).
28. Thevathasan, J. V. et al. Nuclear pores as versatile reference standards for quantitative superresolution microscopy. *Nat. Methods* **16**, 1045–1053 (2019).
29. Dempsey, G. T., Vaughan, J. C., Chen, K. H., Bates, M. & Zhuang, X. Evaluation of fluorophores for optimal performance in localization-based super-resolution imaging. *Nat. Methods* **8**, 1027–1036 (2011).
30. Legant, W. R. et al. High-density three-dimensional localization microscopy across large volumes. *Nat. Methods* **13**, 359–365 (2016).
31. Paszke, A. et al. Pytorch: an imperative style, high-performance deep learning library. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)* Vol. 32, 8024–8035 (2019).
32. Ries, J. SMAP: a modular super-resolution microscopy analysis platform for SMLM data. *Nat. Methods* **17**, 870–872 (2020).
33. von Chamier, L. et al. Democratising deep learning for microscopy with ZeroCostDL4Mic. *Nat. Commun.* **12**, 2276 (2021).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© The Author(s), under exclusive licence to Springer Nature America, Inc. 2021, corrected publication 2021

Methods

DECODE network architecture for probabilistic single-molecule detection and localization. Our architecture consists of two stacked U-nets²⁰ (Extended Data Fig. 1), each with two up- and downsampling stages and 48 filters in the first stage. Each stage consists of three fully convolutional layers with 3×3 filters. In each downsampling stage, the resolution is halved, and the number of filters is doubled, vice versa in each upsampling stage. Upsampling is performed using nearest neighbor interpolation to avoid checkerboard artifacts³⁴. For multiframe DECODE, three consecutive frames are processed by the first frame analysis U-net (with parameters shared for every frame), and the outputs are concatenated and passed to the second temporal context U-net. The entire DECODE network is always trained end-to-end by gradient descent.

For each camera pixel k , the DECODE network predicts (1) a Bernoulli probability map p_k that an emitter was detected near that pixel, (2) the coordinates of the detected emitter $\Delta x_k, \Delta y_k, \Delta z_k$ relative to the center of the pixel x_k, y_k, z_k , (3) a nonnegative emitter brightness ('photon count') N_k and (4) the uncertainties associated with each of these predictions, $\sigma_{x,k}, \sigma_{y,k}, \sigma_{z,k}, \sigma_{N,k}$. For each of these outputs, we use two additional convolutional layers that follow the second U-net. We used the exponential linear unit activation function³⁵ for all hidden units, and the logistic sigmoid nonlinearity for the nonnegative detection probability p , brightness N and the uncertainty outputs $\sigma_x, \sigma_y, \sigma_z, \sigma_N$ (scaled by a prefactor of three). For the coordinate outputs $\Delta x, \Delta y, \Delta z$ we use the hyperbolic tangent nonlinearity, which limits their range to $[-1, 1]$ (that is, to twice the size of a pixel). This way, even though the network can at most predict one emitter per pixel, when necessary, the neighboring pixels can each contribute to place multiple localizations within a single pixel.

New loss function for simultaneous detection, localization and uncertainty estimation. Given a set of E simulated emitters active in each imaging frame with locations for each emitter e given by x_e, y_e, z_e and brightness N_e , and a background image map B_k simulated as described below, we developed a loss function that trains the DECODE network to detect the correct number of emitters, to predict the subpixel localization and brightness for each detection (along with the uncertainty), and to predict the image background. Our loss function is a sum of three terms: a count loss $\mathcal{L}_{\text{count}}$, a localization loss \mathcal{L}_{loc} and a background loss \mathcal{L}_{bg} ,

$$\mathcal{L} = \mathcal{L}_{\text{count}} + \mathcal{L}_{\text{loc}} + \mathcal{L}_{\text{bg}}. \quad (1)$$

The count loss $\mathcal{L}_{\text{count}}$ is a function of the detection probability map p_k with K total pixels and the total number of true emitters E . Interpreting p_k as a Bernoulli detection probability for a single-emitter, we can compute the mean and variance of the predicted total number of emitters detected, if we were to independently sample binary detections from each p_k . While the predicted count distribution $P(E|\{p_k\})$ over the number of emitters detected by this Bernoulli sampling procedure follows an intractable Poisson binomial distribution, we can approximate this predicted distribution as a Gaussian distribution,

$$P(E|\{p_k\}) \approx P(E|\mu_{\text{count}}, \sigma_{\text{count}}^2) = \frac{1}{\sqrt{2\pi}\sigma_{\text{count}}} \exp\left(-\frac{1}{2}\frac{(E - \mu_{\text{count}})^2}{\sigma_{\text{count}}^2}\right). \quad (2)$$

The mean of a sum of Bernoulli random variables is the sum of the means $\mu_{\text{count}} = \sum_{k=1}^K p_k$, and the variance is the sum of the variances of each independent Bernoulli random variable $\sigma_{\text{count}}^2 = \sum_{k=1}^K p_k(1 - p_k)$. This count loss maximizes the log probability of the true number of emitters E under the Gaussian approximation of the predicted count probability distribution. This loss is minimized when μ_{count} correctly matches E , sparsely predicting only one nonzero p_k per detected emitter, and when σ_{count}^2 is small, which happens when p_k are confident and so nearly binary,

$$\mathcal{L}_{\text{count}} = -\log P(E|\mu_{\text{count}}, \sigma_{\text{count}}^2) = \frac{1}{2} \frac{(E - \mu_{\text{count}})^2}{\sigma_{\text{count}}^2} + \log\left(\sqrt{2\pi}\sigma_{\text{count}}\right). \quad (3)$$

The localization loss \mathcal{L}_{loc} is a function of the true emitter locations, and the predicted detection probability map, and the subpixel localizations $\Delta x_k, \Delta y_k, \Delta z_k$, brightness N_k , along with the associated uncertainties $\sigma_{x,k}, \sigma_{y,k}, \sigma_{z,k}, \sigma_{N,k}$ for each detected emitter. For each pixel k , we predict a four-dimensional Gaussian distribution $P(\mathbf{u}_k|\mu_k, \Sigma_k)$ over the absolute position and brightness of an emitter $\mathbf{u} = [x, y, z, N]$ detected in pixel k corresponding to the mean and uncertainty in the subpixel localization and brightness of the emitter detected in pixel k , with mean $\mu_k = [x_k + \Delta x_k, y_k + \Delta y_k, z_k + \Delta z_k, N_k]$ and diagonal covariance matrix $\Sigma_k = \text{diag}(\sigma_{x,k}^2, \sigma_{y,k}^2, \sigma_{z,k}^2, \sigma_{N,k}^2)$,

$$P(\mathbf{u}|\mu_k, \Sigma_k) = \frac{1}{\sqrt{(2\pi)^4 \det(\Sigma_k)}} \exp\left(-\frac{1}{2}(\mu_k - \mathbf{u})^\top \Sigma_k^{-1} (\mu_k - \mathbf{u})\right). \quad (4)$$

Here, the x_k, y_k and z_k are the absolute coordinates for the center of pixel k , so $x_k + \Delta x_k$ corresponds to the absolute coordinates of the emitter to subpixel precision. We note that the localization loss defined below ignores the predicted localization and brightness for pixels where no emitter is detected, that is p_k is zero.

At any given point in training, the true number of emitters will not necessarily match the detected number of emitters perfectly, and we will not have a perfect correspondence between predicted emitters and true emitters. A full probabilistic loss function would sum over all possible assignments of true emitters to detected emitters to correctly evaluate $P(\mathbf{u}|\mu_k, \Sigma_k)$. And since p_k will not necessarily be sparse, the correct cost function would include an intractably large sum over $\binom{K}{E}$ terms. We approximate this by constructing a Gaussian mixture model over the predicted per pixel distributions $P(\mathbf{u}_k|\mu_k, \Sigma_k)$ with mixture weights equal to $p_k/\sum_{j=1}^K p_j$ where the denominator is a sum of the detection probability over all pixels in the image.

The resulting approximation leads to the following localization loss function, which maximizes the probability of the true absolute coordinates and brightness of each ground-truth emitter \mathbf{u}_e^{GT} under the weighted mixture of per pixel probabilities,

$$\mathcal{L}_{\text{loc}} = -\frac{1}{E} \sum_{e=1}^E \log \sum_{k=1}^K \frac{p_k}{\sum_j p_j} P(\mathbf{u}_e^{\text{GT}}|\mu_k, \Sigma_k). \quad (5)$$

The background loss \mathcal{L}_{bg} computes the simple squared error between the predicted and true background maps,

$$\mathcal{L}_{\text{bg}} = \sum_k (B_k^{\text{GT}} - B_k^{\text{pred}})^2. \quad (6)$$

Obtaining localizations and postprocessing. The DECODE network predicts the probabilities p_k of an emitter being located at a specific pixel k . To get deterministic, fast and precise final localizations we use a variant of spatial integration. A detection is considered at pixel k if one of two conditions is met. (1) $p_k > 0.6$. (2) $p_k > 0.3$ and it is a local maximum of a four-connected neighborhood. These candidates are then registered as detections if the cumulative probability of p_k and its four nearest neighbor pixels is >0.7 . Therefore, if the network predicts high confidence detection probability (>0.6) in two adjacent pixels, two emitters will be considered to be detected. However, if a cluster of pixels has low predicted probability, their probabilities will be clustered toward the local maximum, if the local maximum has probability >0.3 , and an emitter will be considered to have been detected if the integrated probabilities of the cluster are >0.7 . The algorithm can be expressed purely in the form of pooling and convolution operations and therefore runs efficiently on a GPU.

For difficult imaging conditions when the predicted localization uncertainties are large, that is high densities, low-SNR values, and large offsets from the focal plane, the subpixel coordinates $\Delta x, \Delta y$ and Δz can be biased toward the center of the pixels (Extended Data Fig. 8). This is because with large predicted localization uncertainty, the predicted mean location is poorly constrained. This bias toward zero (pixel center) scales with the uncertainty of the predictions and can produce artifacts in the reconstructed image depending on how the reconstruction is performed. If a reconstruction uses only the coordinates while ignoring the uncertainty, poorly localized emitters will cluster toward the pixel centers. A more expensive rendering procedure that renders a Gaussian localization distribution with variance proportional to the estimated uncertainty corresponding to each emitter will reduce the impact of this artifact since the bias is usually small relative to the localization uncertainty. Also, filtering out localizations with high uncertainty removes this artifact (Extended Data Fig. 8).

Simulating training data. Training samples are continuously generated in an asynchronous fashion and each frame is only used once as a target. For this reason, the network cannot overfit to specific frames. The performance of our approach will depend on an accurate generative model and could show reduced performance when there is a mismatch between the simulated and experimental data. Thus, we developed a realistic model for the image formation process that incorporates dye blinking behavior, a realistic PSF model and realistic camera read noise.

Structural prior. While incorporating prior structural information has shown to be beneficial^{36,37}, there are concerns that these priors could potentially bias the model to the training data, which could result in the presence of misleading structures after the fitting procedure. We therefore sample the coordinates of the emitters from a 3D homogeneous spatial Poisson point process distribution with density as specified in the text, limits corresponding to the size of the image and the z range for which the PSF was calibrated.

Photophysical prior. In contrast to previous work, DECODE can directly incorporate temporal context into the detection and localization of emitters, rather than as a postprocessing step. We simulate the temporal dynamics of emitters, at least over the short time scale of three imaging frames corresponding to the temporal context of the DECODE network.

For each emitter, the time of initial appearance t_0 is sampled from a continuous random distribution. The on-time of the emitter follows an exponential distribution parametrized by λ . For each emitter, we draw a photon flux from a Gaussian distribution $N(\mu_{\text{flux}}, \sigma_{\text{flux}})$. Together with the amount of time the emitter

is active in each frame, this determines the total number of photons emitted in a frame. Since the input to our model is only a window of three frames, we argue that it is not necessary to model long range temporal correlations that are part of a more detailed photoactivation model³⁸, such as an emitter in the dark state that reappears many frames later. The aforementioned parameters are estimated by a prefit procedure as described in Estimating simulation parameters.

Point spread function. The PSF is a fundamental characteristic of a microscope, specifying the image formed by a single point emitter, and we approximate it to be spatially invariant across the field of view. Given the object $O(\mathbf{r})$ in the object plane, and $\text{PSF}(\mathbf{r})$, the image $I(\mathbf{r})$ results in

$$I(\mathbf{r}) = O(\mathbf{r}) \otimes \text{PSF}(\mathbf{r}), \quad (7)$$

where \otimes denotes the convolution operator. While Gaussian approximations of the PSF are frequently used for both 2D and 3D^{5,6} data, (cubic) spline functions have been shown to achieve more accurate results and can mimic almost arbitrary PSFs^{4,23}. Following Li et al.²³ and Babcock et al.⁴ a 3D PSF can be modeled as

$$f_{i,j,k}(x, y, z) = \sum_{m=0}^3 \sum_{n=0}^3 \sum_{p=0}^3 a_{i,j,k,m,n,p} \left(\frac{x - x_i}{\text{dx}} \right)^m \left(\frac{y - y_j}{\text{dy}} \right)^n \left(\frac{z - z_k}{\text{dz}} \right)^p, \quad (8)$$

where i, j, k are the voxel indices, dx, dy are the pixel sizes; dz is the step size in the axial dimension; x_i, y_j, z_k are the corner coordinates of the voxel (i, j, k) in the respective directions and $a_{i,j,k,m,n,p}$ are the respective spline coefficients, which amounts to 64 coefficients per pixel and per z slice. In a bead calibration routine, the coefficients $a_{i,j,k,m,n,p}$ are estimated and account for varying experimental conditions. Because of the simple form of equation (8), the CRLB with respect to the fitting parameters x, y, z can be calculated easily as the diagonal elements of the inverse of the Fisher information matrix²².

Camera model. All real datasets presented in this work were recorded with an electron multiplying charge-coupled device (EMCCD) camera, with the exception of the LLS data that were recorded with an sCMOS camera. The measured camera signal is subject to various noise sources, which we will discuss in the following:

Shot noise originates from the stochastic nature of photons when interacting with the camera chip. The expected number of detected electrons is

$$\lambda_k = \lambda_{0,k} \cdot \text{QE} + c_s. \quad (9)$$

Here, $\lambda_{0,k}$ is the expected number of photons that are collected in pixel k , QE is the quantum efficiency, and c_s the spurious charge, measured in electrons. The probability $p_{\text{shot}}(s_k)$ of observing the signal s_k in pixel k follows a Poisson distribution,

$$p_{\text{shot}}(s_k) = \frac{\lambda_k^{s_k} e^{-\lambda_k}}{s_k!}. \quad (10)$$

EMCCD amplification noise stems from the amplification of photo electrons that pass through the gain register and stochastically generate additional electrons. For our EMCCD camera noise model, we follow Huang et al.³⁹. EMCCD amplification noise can be described approximately by a Gamma distribution,

$$\rho_{\text{EM}}(x|s_k, \theta) = \frac{1}{\Gamma(s_k)\theta^{s_k}} x^{s_k-1} e^{-\frac{x}{\theta}}. \quad (11)$$

$\rho_{\text{EM}}(x|s_k, \theta)$ denotes the probability that s_k input photo electrons in pixel k with an electron multiplying gain of θ create x output electrons after the gain register.

Read noise stems from the process of converting electrons into a digital signal. In this process, the signal is usually multiplied by a gain factor g and an offset o is added to avoid negative signal. In this work, we convert the input camera image to photon units before inference by subtracting o and dividing by g . In addition, when using EMCCD cameras we divide by the electron multiplying gain θ , thus the units of the read noise are photo electrons. We approximate the read noise (both for sCMOS and EMCCD cameras) by a zero mean additive Gaussian distribution with variance σ^2 ,

$$\rho_{\text{read}}(x|0, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}. \quad (12)$$

Training details. Training was performed on 40×40 pixel-sized regions that are directly simulated or randomly selected from larger simulated images at each iteration. We used the AdamW optimizer⁴⁰ with a group learning rate of 6×10^{-4} for the network parameters. We reduce the learning rate by a factor of 0.9 after every 1,500 iterations with a batch size of 64. To stabilize training, we use gradient norm clipping with a maximum norm of 0.03.

Very dim emitters with less than 50 photons are excluded from the ground-truth targets (but still rendered) so that the network is discouraged to make predictions for practically invisible emitters.

Estimating simulation parameters. For training DECODE, a proper parametrization of the simulation is needed to match the real data distribution. In a prefitting step, the main parameters, that is the emitter on-time, emitter brightness and background, can be determined. The prefitting can be performed with a single-emitter MLE fitter after filtering the log-likelihood value to exclude data from overlapping PSFs. This step is incorporated in the SMAP software for the sake of ease of use³². We observed that the precise values of the simulation parameters of the emitters' photophysics (that is, lifetime and brightness) and density are not crucial, as the stochastic nature of the emitters' positions, brightness levels and appearance times presents the network with data that match the real experiments under different conditions and effectively covers a broad range of these parameters.

The camera parameters are usually given by the manufacturer.

The given network architecture and training parameters are effective across different real and simulated datasets and in our experience do not have to be optimized by the end user.

Evaluating localization error and reconstruction resolution. To evaluate performance on the challenge datasets, as well as our own simulations, we use two metrics.

First, instead of the Euclidean distance, we use the localization error, measured in nm, which is the r.m.s.e. averaged over the dimensions:

$$\text{r.m.s.e.}_d = \left(\frac{1}{\text{TP}} \sum_{i=1}^{\text{TP}} \sum_{k=1}^d (x_{i,k} - x_{i,k}^{\text{GT}})^2 / d \right)^{1/2} \quad (13)$$

TP is the number of localizations that are matched to ground-truth (GT) coordinates, d is the dimension (two for 2D data, three for 3D data), $x_k = x, y, z$ are the predicted coordinates and $x_k^{\text{GT}} = x, y, z$ the ground-truth coordinates.

Second, the detection accuracy or JI, which quantifies how well an algorithm does at detecting all the emitters while avoiding false positives:

$$\text{JI} = \text{TP}/(\text{FN} + \text{FP} + \text{TP}) \quad (14)$$

TP are the true positives, FN the false negatives and FP the false positives.

Localizations are matched to ground-truth coordinates when they are within a circle of 250 nm radius and the distance in z is less than 500 nm. As a single metric that evaluates the ability to reliably infer emitters with high precision, we use the efficiency metric as defined in ref.⁷:

$$E = 1 - \sqrt{(1 - \text{JI})^2 + \alpha^2 d \cdot \text{r.m.s.e.}_d^2} \quad (15)$$

Lateral and axial efficiency are calculated based on r.m.s.e._2 and r.m.s.e._1 with alpha values of $\alpha = 1 \times 10^{-2}$ and $\alpha = 0.5 \times 10^{-2} \text{ nm}^{-1}$, respectively and then averaged to obtain the overall efficiency. Detection accuracy is expressed in units of 0 to 1 (or 0–100%), the efficiency ranges up to 1 (or 100%) for a perfect fitting algorithm.

The Fourier ring correlation^{25,41} (FCR) in Fig. 4a was calculated by dividing the data in ten blocks of equal number of frames and constructing super-resolution images from even and odd blocks (pixel size 5 nm).

Simulating data for performance evaluation. To simulate data for performance evaluation and comparison shown in Fig. 2, we assumed an ideal camera without EMCCD or read noise and an image size of 64×64 pixels. We used the PSF model that was acquired for the dataset in Fig. 4a. Data used to test the effect of the SNR and density were simulated using the structural and photophysical prior previously described with an average on-time of two frames. Precise simulation parameters can be found in Supplementary Table 1.

The CRLB is evaluated as the diagonal elements of the inverse of the Fisher information matrix²² with the simulated parameters and spline interpolated experimental PSF model and was calculated with the SMAP software³². A bootstrap estimate ($N = 10,000$) of the r.m.s.e. was used to estimate the s.e.m. on the localization error.

Sample preparation and localization microscopy. See Supplementary Note for details on sample preparation and localization microscopy.

Reporting Summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

All data can be downloaded from <https://doi.org/10.25378/janelia.14674659>. Raw data and bead frames are available for Figs. 2a–e and 4a–h and Extended Data Figs. 2, 3, 4a, 5 and 8. Localizations and performance metrics (for DECODE and CSpline/DeepSTORM3D when applicable) are available for Figs. 2a–e, 4a–h and 5 and Extended Data Figs. 2, 3, 4a, 7 and 8. The parametrization of the simulation for Fig. 2a–e is available and can be used to generate data. Raw data and bead frames, as well as performance metrics for Fig. 3, are publicly available at <http://bigwww.epfl.ch/smlm/challenge2016/>. Raw data and bead frames for Fig. 5 and Extended Data Fig. 7 are available on request from the authors of ref.³⁰. All other data

supporting the findings of this study are available from the corresponding authors upon reasonable request. Source data are provided with this paper.

Code availability

DECODE is available as Supplementary Software. Updated versions can be found at <https://github.com/TuragaLab/DECODE>.

References

34. Odena, A., Dumoulin, V. & Olah, C. Deconvolution and checkerboard artifacts. *Distill* <https://distill.pub/2016/deconv-checkerboard/> (2016).
35. Clevert, D.-A., Unterthiner, T. & Hochreiter, S. Fast and accurate deep network learning by exponential linear units (ELUs). Preprint at <https://arxiv.org/abs/1511.07289> (2016).
36. Ouyang, W., Aristov, A., Lelek, M., Hao, X. & Zimmer, C. Deep learning massively accelerates super-resolution localization microscopy. *Nat. Biotechnol.* **36**, 460–468 (2018).
37. Weigert, M. Content-aware image restoration: pushing the limits of fluorescence microscopy. *Nat. Methods* **15**, 1090–1097 (2018).
38. Annibale, P., Vanni, S., Scarselli, M., Rothlisberger, U. & Radenovic, A. Quantitative photo activated localization microscopy: unraveling the effects of photoblinking. *PLoS ONE* **6**, e22678 (2011).
39. Huang, F. Video-rate nanoscopy using scmos camera-specific single-molecule localization algorithms. *Nat. Methods* **10**, 653–658 (2013).
40. Loshchilov, I. & Hutter, F. Decoupled weight decay regularization. Preprint at <https://arxiv.org/abs/1711.05101> (2019).
41. Banterle, N., Bui, K. H., Lemke, E. A. & Beck, M. Fourier ring correlation as a resolution criterion for super-resolution microscopy. *J. Struct. Biol.* **183**, 363–367 (2013).
42. Perlin, K. An image synthesizer. *Comput. Graph. (ACM)* **19**, 287–296 (1985); <https://doi.org/10.1145/325165.325247>

Acknowledgements

S.C.T. is supported by the Howard Hughes Medical Institute. J.R., L.-R.M. and P.H. were supported by the European Molecular Biology Laboratory, the European Research

Council (grant no. CoG-724489 to J.R.) and the National Institutes of Health Common Fund 4D Nucleome Program (grant no. U01 EB021223 to J.R.). J.H.M. and A.S. were supported by the German Research Foundation (DFG) through Germany's Excellence Strategy (EXC-Number 2064/1, project no. 390727645) and the German Federal Ministry of Education and Research (BMBF, project no. 'ADMIMEM', FKZ 01IS18052). W.R.L. acknowledges support from the Searle Scholars Program, the Beckman Young Investigator Program, an National Institutes of Health New Innovator Award (no. DP2GM136653) and the Packard Fellows Program. We thank D. Sage for useful discussions, U. Boehm, D. Greenberg and P. Ramesh for comments on the manuscript, and E. Betzig and J. Lippincott Schwartz for kindly sharing data with us. We are grateful to C. Leterrier for extensive testing and useful suggestions for improving the DECODE library, and to U. Boehm for helping create DECODE tutorials. We thank D. Olbris and D. Kutra for assistance with automation of building and deploying DECODE across multiple platforms.

Author contributions

A.S., L.-R.M., J.H.M., J.R. and S.C.T. conceived the project, analyzed the results and wrote the paper with input from all authors. A.S. and L.-R.M. wrote the software. U.M., P.H. and J.R. acquired and analyzed the biological data. A.K. provided supervision. C.J.O. and W.R.L. provided the LLS data and helped with analysis.

Competing interests

The authors declare no competing interests.

Additional information

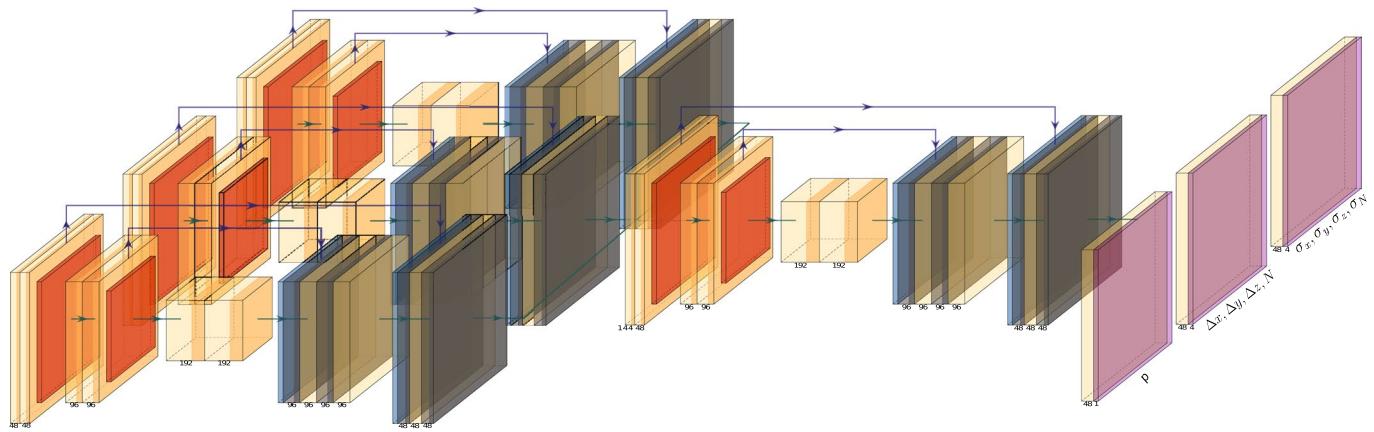
Extended data is available for this paper at <https://doi.org/10.1038/s41592-021-01236-x>.

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41592-021-01236-x>.

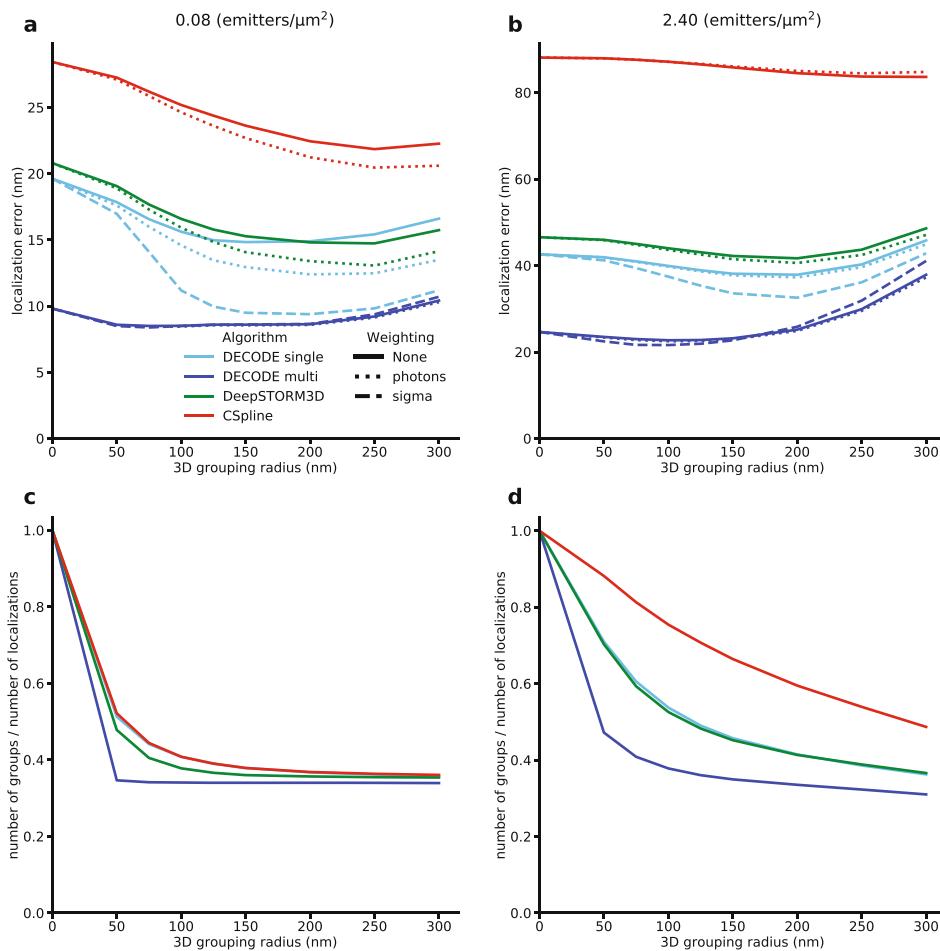
Correspondence and requests for materials should be addressed to J.H.M., J.R. or S.C.T.

Peer review information *Nature Methods* thanks Alex Herbert and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

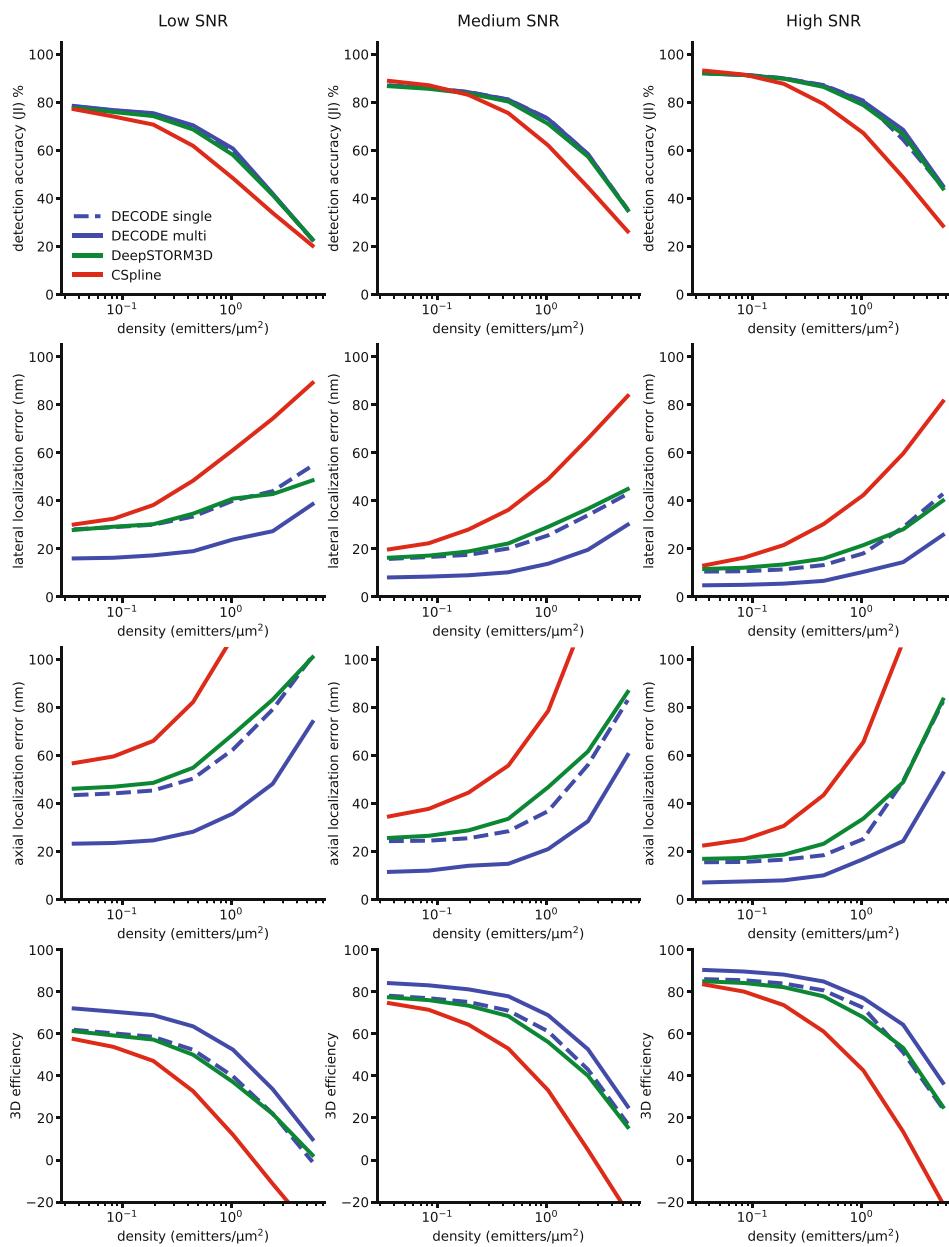
Reprints and permissions information is available at www.nature.com/reprints.



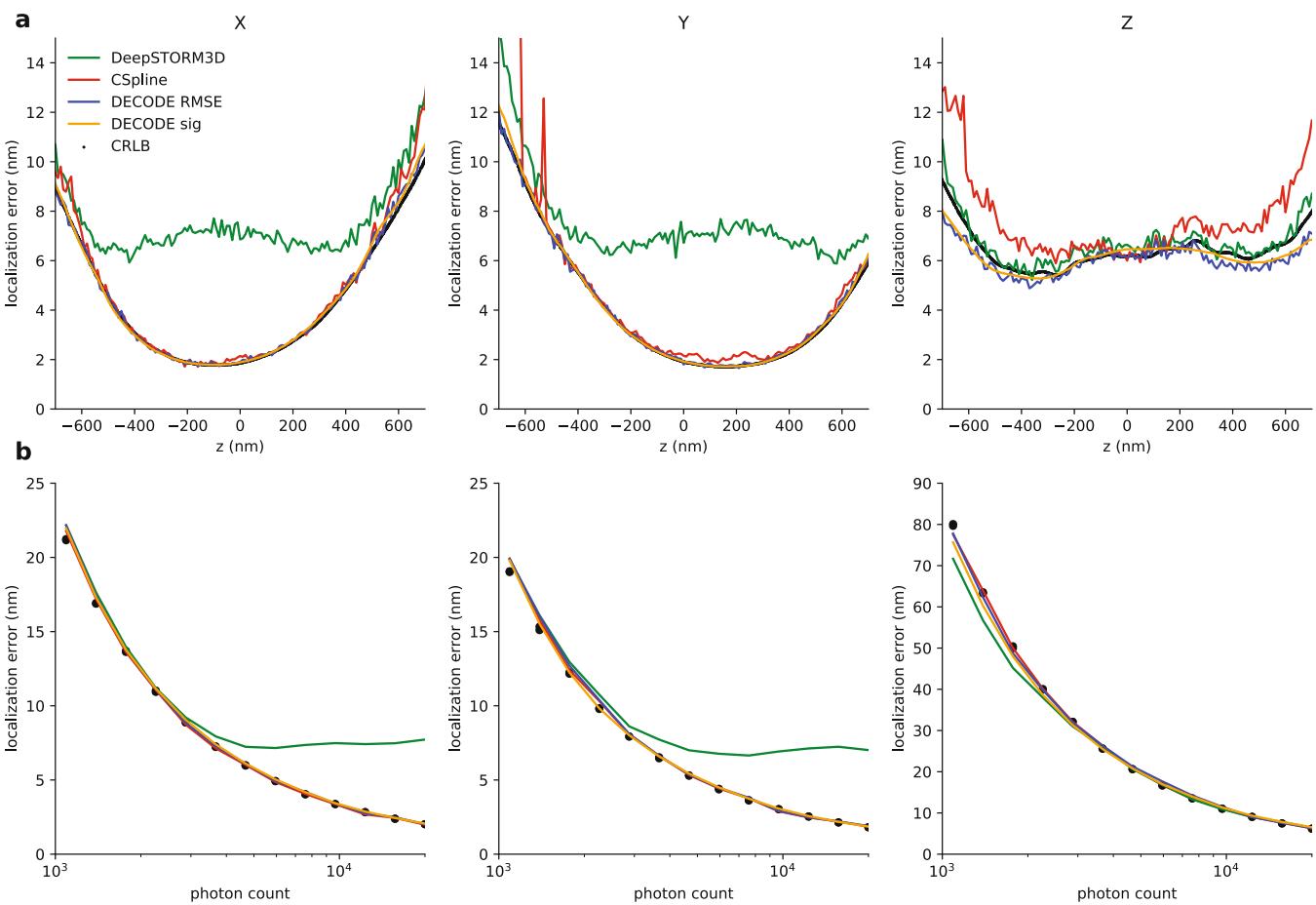
Extended Data Fig. 1 | Architecture. The DECODE network consists of two stacked U-Nets²⁰ with identical layouts (the three networks depicted on the left share parameters). The *frame analysis module* extracts informative features from three consecutive frames. These features are integrated by the *temporal context module*. Both U-Nets have two up- and downsampling stages and 48 filters in the first stage. Each stage consists of three fully convolutional layers with 3×3 filters. In each downsampling stage, the resolution is halved, and the number of filters is doubled, vice versa in each upsampling stage. Blue arrows show skip connections. Following the *temporal context module* three output heads with two convolutional layers each produce the output maps which have the same spatial dimensions as the input frames. The first head predicts the Bernoulli probability map p , the second head the spatial coordinates of the detected emitter $\Delta x, \Delta y, \Delta z$ and its intensity N and the third head the associated uncertainties $\sigma_x, \sigma_y, \sigma_z, \sigma_N$. An optional fourth output head can be used for background prediction.



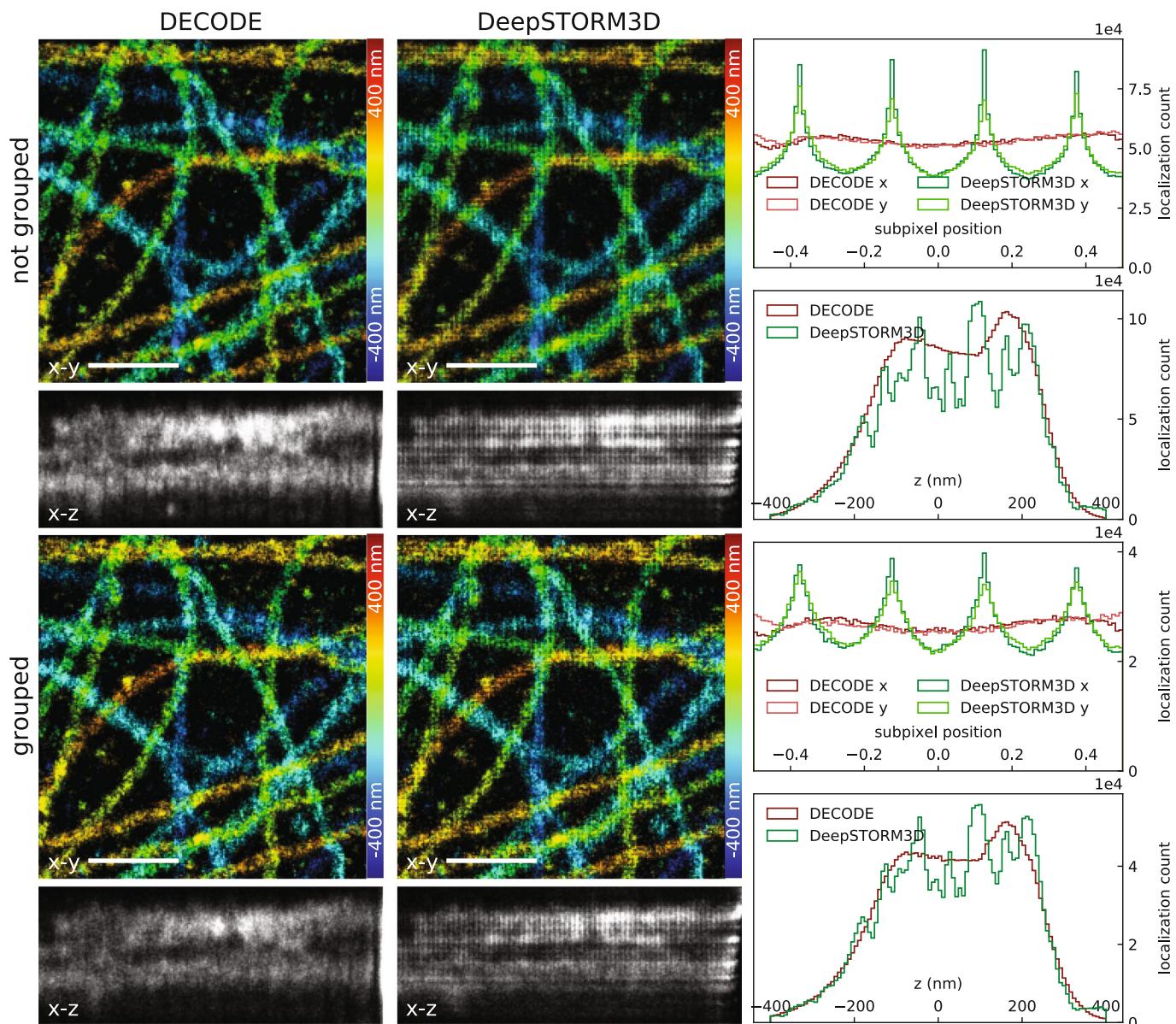
Extended Data Fig. 2 | Impact of grouping across grouping radius for different averaging weights. Predictions in consecutive frames are grouped when they are closer to each other than the given grouping radius. A grouping radius of 0 nm corresponds to not performing any grouping. Predictions within a group are assigned a common set of emitter coordinates which is calculated as weighted average of their individual coordinates. We compare three different options for the weighted average: Uniform weighting ('None', solid lines); Weighting by the inferred number of photons for CSpline and DECODE or the inferred confidence for DeepSTORM3D ('photons', dotted line); Weighting by the predicted DECODE σ values, where the x,y and z values are individually weighted by $\sigma_{x,y,z}^{-2}$. **a, b:** 3D efficiencies across grouping radii. Grouping is especially useful in the low density setting (a) where DECODE without temporal context (DECODE single) with a correctly set grouping radius can match the performance of DECODE with temporal context (DECODE multi) without grouping. This is, however, only the case when weighting by the uncertainty estimates that DECODE provides. Using grouping on top of DECODE multi offers little additional benefit. **c, d:** Number of groups divided by the number of localizations. Detecting all emitters and correctly grouping them would result in a ratio of 1:3 as on average each emitter is visible in three consecutive frames. See methods and Supplementary Table 1 for additional details on training and evaluation.



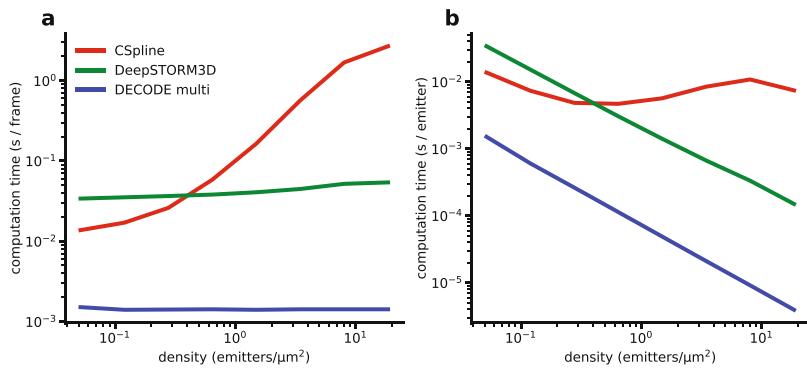
Extended Data Fig. 3 | Comparison of performance metrics across densities and SNRs. DECODE outperforms DeepSTORM3D and CSpline across densities and SNRs. See methods and Supplementary Table 1 for additional details on training and evaluation.



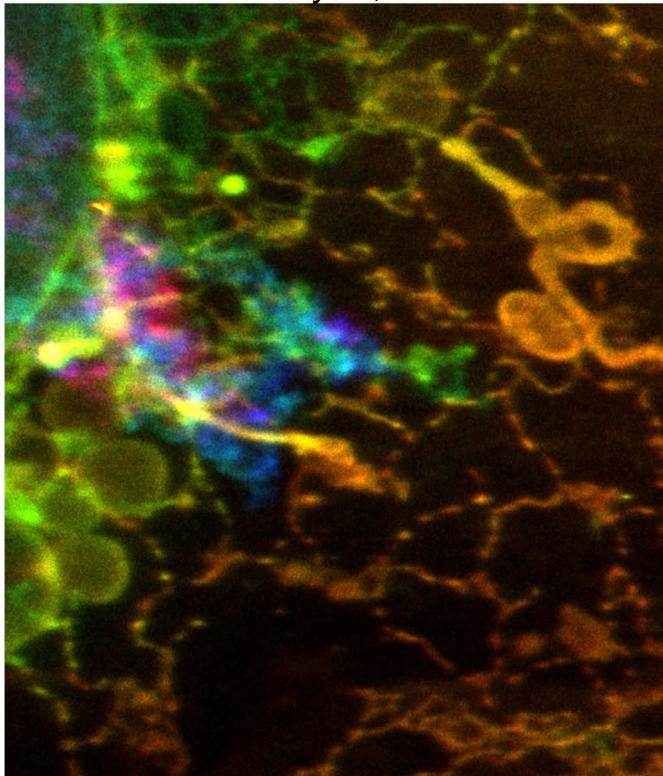
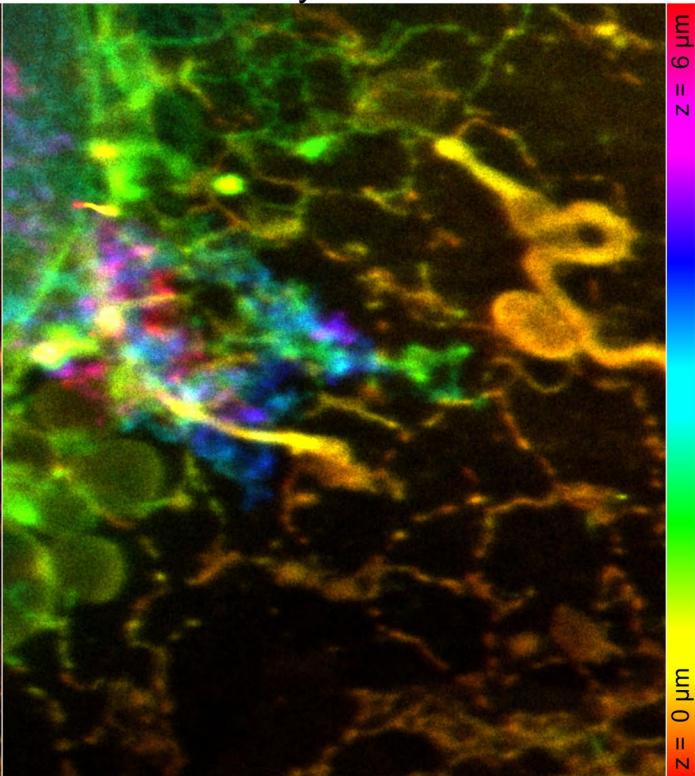
Extended Data Fig. 4 | Comparison of localization error and CRLB for single-emitter fitting. The r.m.s.e. achieved by DECODE and its predicted σ values closely match the single emitter CRLB in every dimension. CSpline is also able to achieve the CRLB, which has been shown for iterative MLE fitters before. In contrast the resolution that DeepSTORM3D can achieve is limited by its output representation and the size of the super-resolution voxels. **a**): Data simulated with high SNR (20,000 photons) and random z. r.m.s.e. and DECODE σ averaged over 10 nm bins. **b**): Data simulated with fixed z (0 nm) and varying SNR levels. See methods and Supplementary Table 1 for additional details on training and evaluation.



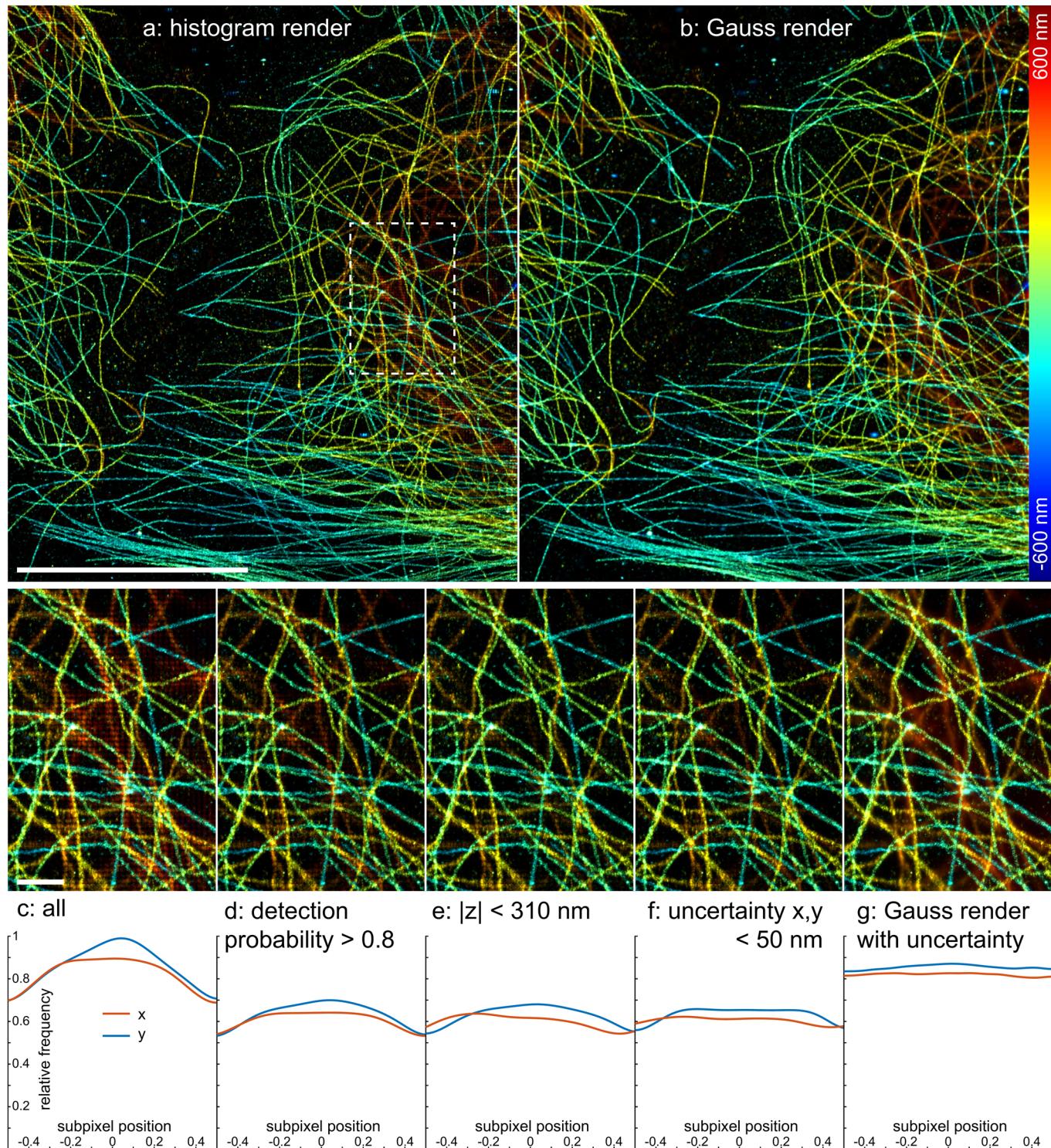
Extended Data Fig. 5 | Comparison of reconstruction quality on experimental STORM data. Reconstructions by DECODE and the DeepSTORM3D on a subset of data shown in Fig. 4g. Histograms show within pixel distribution of localizations in x and y as well as the z coordinate in nm. DeepSTORM3D has 4 significant peaks in the subpixel distribution, corresponding to the fourfold upsampling it uses for its network output. These are visible as grid artifacts in the reconstructions. In contrast the DECODE localizations are evenly distributed and no artifacts are visible. Scale bars 0.5 μm.



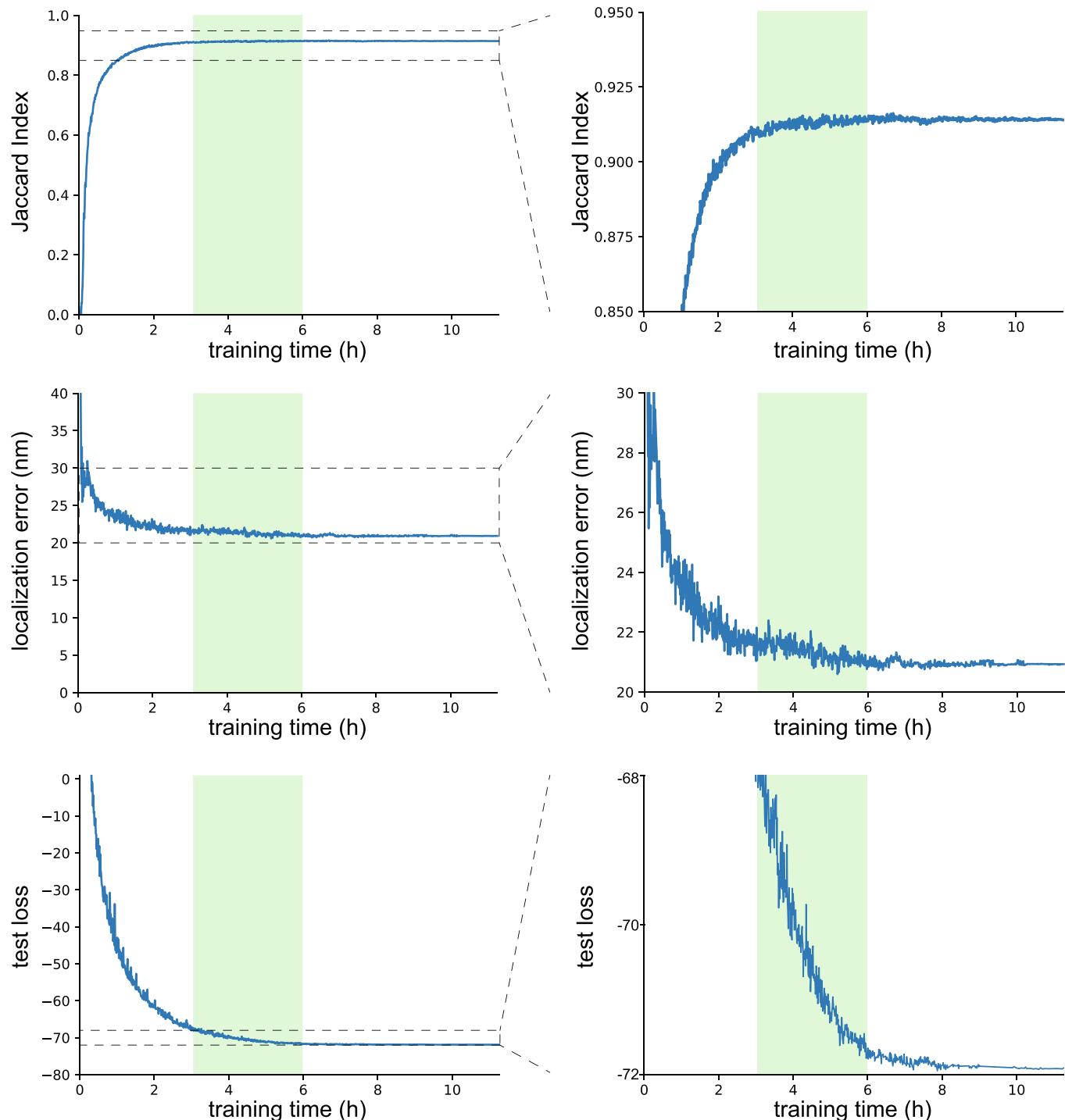
Extended Data Fig. 6 | Comparison of computation times. **a**) Measured as the time it takes to analyze a 64×64 pixel frame with varying emitter densities. Trained DECODE and DeepSTORM3D models were evaluated using a NVIDIA RTX2080Ti GPU. Computation time includes the network forward pass and postprocessing and does not include training time. CSpline was evaluated on an Intel(R) Xeon(R) CPU E5-2697 v3. **b**) Computation time per simulated emitter. The computation time of CSpline scales with the number emitters while the two deep learning based approaches scale with the number (and size) of the analyzed frames. GPU-based DECODE is about 20 times faster than GPU-based DeepSTORM3D and outperforms CPU-based CSpline even at low densities.

a DECODE analysis, 35k frames**b** standard analysis 70k frames

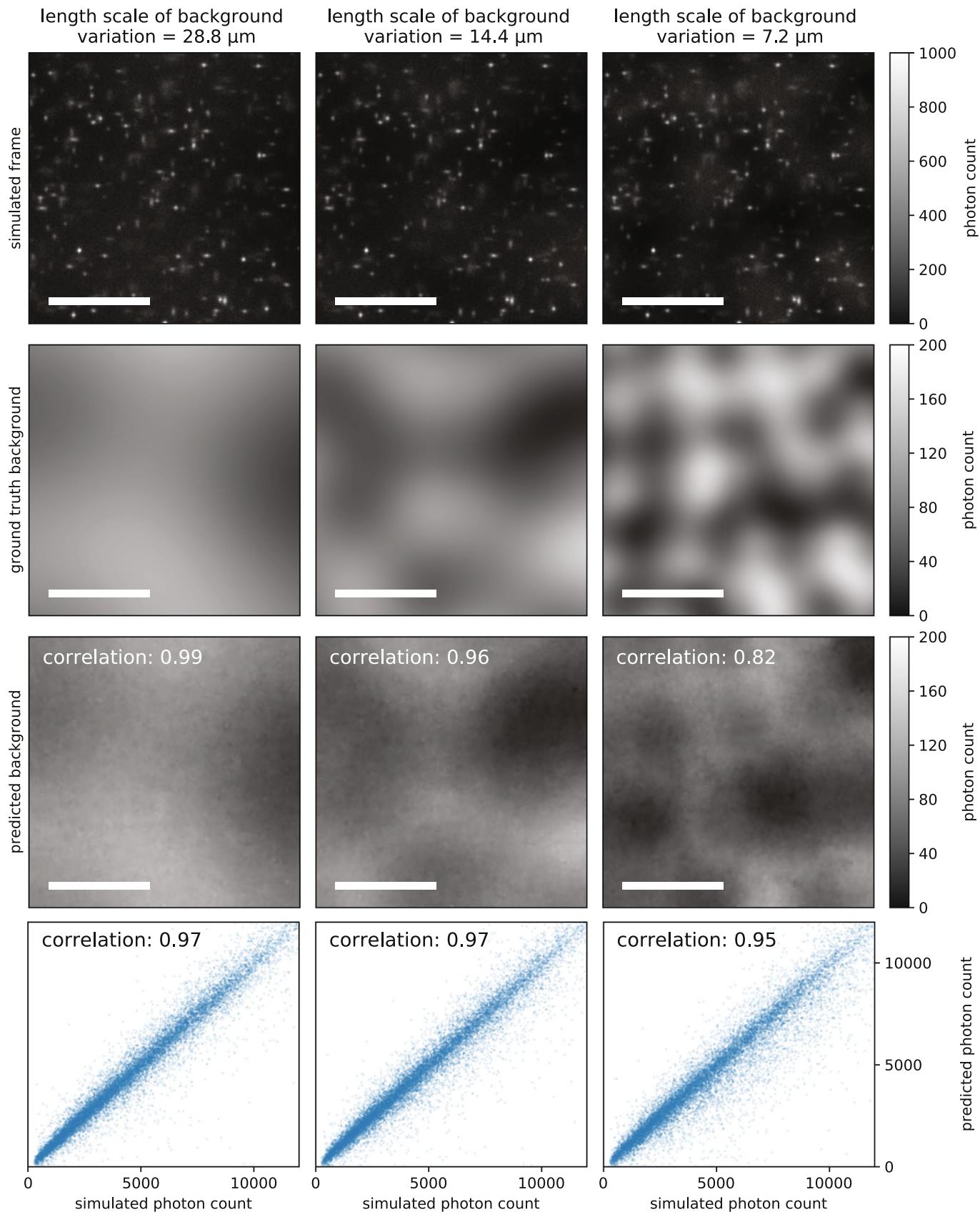
Extended Data Fig. 7 | DECODE reduces acquisition times in LLS-PAINT. DECODE reconstruction of 35,000 frames (**a**) results in the same number of localizations as the Standard reconstruction of 70,000 frames (**b**). As DECODE detects twice as many localizations as the traditional analysis, it needs only approx. half of the frames for a high-quality reconstruction.



Extended Data Fig. 8 | Removing Pixelation artifacts. Dim, dense out-of-focus localizations have a bias towards the pixel center (**a,c**). This is apparent as a non-uniform distribution of the sub-pixel positions in x and y (bottom row). This bias is not visible if every localization is rendered as a Gaussian with a standard deviation equal to the predicted uncertainty s (**b,g**). Filtering according to the detection probability reduces the artifact (**d**). Filtering according to the predicted uncertainty σ or the fluorophore z-position (**e**) also removes the pixelation artifact. Scale bars 10 μm (**a,b**) and 1 μm (**c-g**). The overview images (**a,b**) are rendered with a pixel size of 10 nm, the zoom-ins (**c-g**) with a pixel size of 4 nm. The camera used to record the data has a pixel size of 117 \times 127 nm.



Extended Data Fig. 9 | Performance as a function of deep network training time. Convergence of the accuracy of DECODE for several performance metrics. Runtimes are measured on a single nVidia RTX 2080 Ti GPU. The estimated training achievable with the maximum of 12 hours possible on the free tier of Google Colab is shown in green range (assuming that a Google Colab GPU is $2\times$ – $4\times$ slower than the nVidia RTX 2080 Ti GPU). This suggests that acceptable performance is achievable using DECODE and Google Colab at minimal cost, no GPU needed. Metrics evaluated for prediction >0.5 detection probability estimate without sigma filtering. Training data was simulated at high SNR (as described in Fig. 2c) at an average density of $1\mu\text{m}^{-2}$.



Extended Data Fig. 10 | DECODE provides accurate background and signal predictions. Shown on simulated data with inhomogeneous background of various length scales. First row: sample frames. Second row: background values simulated using Perlin noise⁴². Third row: background values inferred by a DECODE network that was trained on 40×40 pixel sized simulations with uniform background. Fourth row: Scatter plot of inferred photon counts over simulated photon counts. Scale bars are $10 \mu\text{m}$.

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our [Editorial Policies](#) and the [Editorial Policy Checklist](#).

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a Confirmed

- The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement
- A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
- The statistical test(s) used AND whether they are one- or two-sided
Only common tests should be described solely by name; describe more complex techniques in the Methods section.
- A description of all covariates tested
- A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
- A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
- For null hypothesis testing, the test statistic (e.g. F , t , r) with confidence intervals, effect sizes, degrees of freedom and P value noted
Give P values as exact values whenever suitable.
- For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
- For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
- Estimates of effect sizes (e.g. Cohen's d , Pearson's r), indicating how they were calculated

Our web collection on [statistics for biologists](#) contains articles on many of the points above.

Software and code

Policy information about [availability of computer code](#)

Data collection

SMLM Data collection was performed on custom microscopes detailed in the manuscript, using a custom written μManager (1.4.22 or μManager 2 gamma) plugin.

Data analysis

Data was curated and analyzed on custom code written in Python (DECODE), and also with other software which has been published previously: MATLAB R2019a (SMAP, github.com/jries/SMAP) and Python (CSpline, github.com/ZhuangLab/storm-analysis). DECODE is freely available at github.com/TuragaLab/DECODE

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research [guidelines for submitting code & software](#) for further information.

Data

Policy information about [availability of data](#)

All manuscripts must include a [data availability statement](#). This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

All data can be downloaded from <https://doi.org/10.25378/janelia.14674659>.

Raw data and bead frames are available for Fig. 2a-e, 4a-h and Extended Data Figure 2, 3, 4a, 5, 8.

Localizations and performance metrics (for DECODE and CSpline/DeepSTORM3D when applicable) are available for Fig. 2a-e, 4a-h, 5, Extended Data Figure 2, 3, 4a, 7 and 8.

The parametrization of the simulation for Fig. 2a-e is available and can be read out.

Raw data and bead frames, as well as performance metrics for Fig 3, are publically available at (<http://bigwww.epfl.ch/smlm/challenge2016/>). Raw data and bead frames for Fig. 5 and Extended Data Figure 7 are available on request from the authors of Legant et. al. 2016.
All other data supporting the findings of this study are available from the corresponding authors upon reasonable request.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

- Life sciences Behavioural & social sciences Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

| | |
|-----------------|--|
| Sample size | No hypothesis based experiment was performed. Therefore, the sample size was not predetermined. For all simulations, the data size was chosen large enough that statistical errors are minimal compared to the effect. The variation in performance between repeated training of the neural network was low. |
| Data exclusions | No data was excluded. |
| Replication | All code is open source to help with the reproducibility of computational analysis. Experimental findings were reproduced successfully across different datasets. |
| Randomization | Samples were not randomized as this work presents an image analysis software. We used a wide array of different simulations and datasets to systematically test different imaging conditions. |
| Blinding | The data analyzed in this study was used to illustrate different possible applications and not to test a specific hypothesis. Therefore, blinding is not relevant for this work. |

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

| Materials & experimental systems | | Methods | |
|-------------------------------------|---|-------------------------------------|---|
| n/a | Involved in the study | n/a | Involved in the study |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> Antibodies | <input checked="" type="checkbox"/> | <input type="checkbox"/> ChIP-seq |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> Eukaryotic cell lines | <input checked="" type="checkbox"/> | <input type="checkbox"/> Flow cytometry |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Palaeontology and archaeology | <input checked="" type="checkbox"/> | <input type="checkbox"/> MRI-based neuroimaging |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Animals and other organisms | | |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Human research participants | | |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Clinical data | | |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Dual use research of concern | | |

Antibodies

| | |
|-----------------|---|
| Antibodies used | anti-a-tubulin (MS581; NeoMarkers, Fremont, CA, USA), anti-b-tubulin (T5293; Sigma-Aldrich), anti-mouse Alexa Fluor 647 (A21236; Invitrogen, Carlsbad, CA, USA) |
| Validation | Highly specific staining during SMLM imaging validates the antibodies. The antibodies showed highly specific staining in super-resolution microscopy. |

Eukaryotic cell lines

Policy information about [cell lines](#)

| | |
|--------------------------|--|
| Cell line source(s) | U-2 OS cells (ATCC HTB-96), U-2 OS Nup96-mMaple (300461, CLS Cell Line Service, Eppelheim, Germany), |
| Authentication | None of the cell lines were authenticated. |
| Mycoplasma contamination | Cell lines have been tested and are negative for mycoplasma contamination. |

