Synthetic data enables human-grade microtubule analysis with foundation models for segmentation

Mario Koddenbrock[◕,1,‡], Justus Westerhoff[◕,2,‡], Dominik Fachet[3], Simone Reber[2,3], Felix Gers[2], Erik Rodner[1,4,‡]

1 KI Werkstatt, Hochschule für Technik und Wirtschaft Berlin (HTW), Berlin, Germany
2 Berliner Hochschule für Technik (BHT), Berlin, Germany
3 Max Planck Institute for Infection Biology, Berlin, Germany
4 Merantix Momentum GmbH, Berlin, Germany

◕ These authors contributed equally to this work.
‡ Corresponding authors: `mario.koddenbrock@htw-berlin.de`, `justus.westerhoff@bht-berlin.de`, `erik.rodner@htw-berlin.de`

Abstract

Studying microtubules (MTs) and their mechanical properties is central to understanding intracellular transport, cell division, and drug action. While important, experts still need to spend many hours manually segmenting these filamentous structures. The suitability of state-of-the-art methods for this task cannot be systematically assessed, as large-scale labeled datasets are missing. We address this gap by presenting the synthetic dataset `SynthMT`, produced by tuning a novel image generation pipeline on real-world interference reflection microscopy (IRM) frames of *in vitro* reconstituted MTs without requiring human annotations. In our benchmark, we evaluate nine fully automated methods for MT analysis in both zero- and Hyperparameter Optimization (HPO)-based few-shot settings. Across both settings, classical algorithms and current foundation models still struggle to achieve the accuracy required for biological downstream analysis on *in vitro* MT IRM images that humans perceive as visually simple. However, a notable exception is the recently introduced SAM3 model. After HPO on only ten random `SynthMT` images, its text-prompted version SAM3Text achieves near-perfect and in some cases super-human performance on unseen, real data. This indicates that fully automated MT segmentation has become feasible when method configuration is effectively guided by synthetic data. To enable progress, we publicly release the dataset, the generation pipeline, and the evaluation framework at `DATEXIS.github.io/SynthMT-project-page`.

Author summary

Understanding the behavior of microtubules — stiff filaments inside cells — is essential for studying fundamental cell biological processes and for developing therapies for diseases such as cancer and neurodegenerative conditions. Yet, analyzing microtubule images is slow and labor-intensive, as researchers must manually trace these thin, overlapping filaments, which can take hours and is prone to errors. We therefore asked whether current fully automated segmentation methods are ready to replace manual microtubule analysis, and how synthetic data can be used to rigorously evaluate and improve them.

To address these questions, we created a synthetic dataset that mimics real microtubule images by capturing the appearance and variability of real microscopy. Importantly, this dataset can be generated without any manual annotations. Using this dataset, we evaluated a range of segmentation methods and found that most of them struggled to accurately identify filaments.

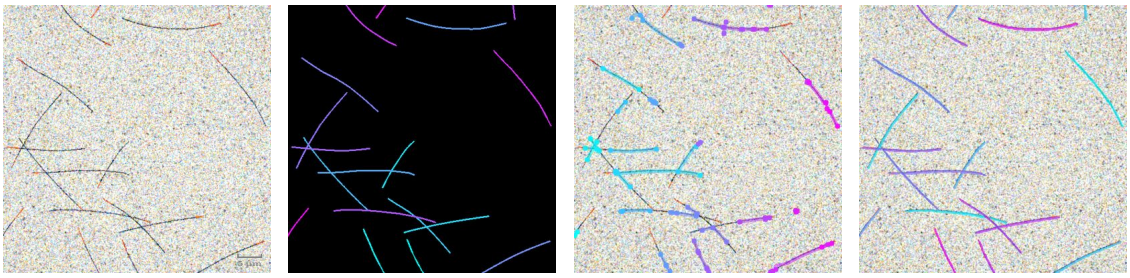
However, we discovered that a recent foundation model, when guided by a simple text instruction and tuned on only a few synthetic images, can achieve near-perfect, human-level performance on previously unseen real microtubule imaging data.

Our work demonstrates that fully automated microtubule analysis is now possible and provides a reproducible framework that other researchers can use to evaluate and improve their methods. This opens the door to faster, more consistent studies of microtubules, ultimately accelerating discoveries in cell biology and therapeutic research.

1 Introduction

Microtubules (MTs) are cytoskeletal filaments essential for cellular processes such as chromosome segregation, intracellular transport, and cell motility. They are formed by head-to-tail polymerization of $\alpha\beta$-tubulin heterodimers into protofilaments that laterally associate to form a hollow cylinder with a diameter of about $25\,nm$ [1]. Their plus ends exhibit *dynamic instability*, stochastically switching between growth and shrinkage, a behavior exquisitely sensitive to regulatory proteins and small molecules. This sensitivity makes MTs prime targets in drug discovery: widely used chemotherapeutics (e.g., taxanes [2], vinca alkaloids [3]) exploit destabilization, while emerging neuroprotective strategies aim to stabilize MT dynamics in neurodegenerative diseases [4]. To investigate how candidate compounds, cofactors, or post-translational modifications alter growth behavior, curvature, and length distributions, MTs are often reconstituted *in vitro* from stabilized nucleation seeds and visualized using microscopy techniques like total internal reflection fluorescence (TIRF) or interference reflection microscopy (IRM). A key step in such analyses is the accurate quantification of filament properties (count, length, curvature), which requires precise instance segmentation. However, manual annotation of MT images is labor-intensive and error-prone, creating a significant bottleneck in experimental and preclinical workflows.

Automated segmentation methods offer a promising solution, but their development and evaluation are hampered by a major challenge: the lack of large-scale, publicly available, and annotated datasets for *in vitro* MT images that resemble the typical noise of biological imaging data. This data scarcity has been repeatedly noted [5–7], yet no comprehensive benchmark exists.



(a) IRM-like image from SynthMT (b) Ground-truth instance masks (c) FIESTA anchor-point predictions (SKIoU = 0.19) (d) SAM3Text mask predictions (SKIoU = 0.99)

Figure 1. The `SynthMT` instance segmentation benchmark evaluates methods on synthetic interference reflection microscopy (IRM)-like images containing microtubules (MTs). (a) Synthetic image mimicking IRM of *in vitro* reconstituted MTs nucleated from fixed seeds (visualized in red), reproducing key mechanical and geometrical properties such as filament length and curvature. (b) Our pipeline generates accompanying ground-truth instance masks that enable quantitative evaluation. (c) The classical FIESTA [8] algorithm predicts anchor points for each instance (for visual clarity, only the first and last point of each instance are shown), which we connect through splines. The example demonstrates typical failure modes: filament fragmentation (single MTs split into multiple instances), incomplete segmentation, and artifacts at intersections. (d) SAM3 [9] guided by a simple text prompt ("*thin line*") produces precise, human-grade segmentation, accurately tracing intersecting MTs. This is supported by its high Skeleton Intersection over Union (SKIoU) for this specific image.

While general-purpose segmentation models, including recent foundation models like CellSAM [10], μSAM [11], and Cellpose-SAM [12], have shown impressive zero-shot performance on various microscopy datasets, they are primarily designed for round objects such as nuclei or whole cells. Their applicability to filamentous structures such as MTs remains largely unproven, and they often lack the accuracy necessary for reliable downstream biological analysis, especially when dealing with out-of-distribution data [13–16]. This raises a central question: **Are current fully automated methods ready to replace manual analysis for MT segmentation?** As sufficiently advanced methods often critically depend on their hyperparameters, we also ask to what extent synthetic data that mimics real microscopy images can enable few-shot learning through Hyperparameter Optimization (HPO), thereby improving generalization to real-world MT data.

This paper addresses these questions by introducing `SynthMT`, a synthetic benchmark designed to systematically evaluate the readiness of segmentation methods for automated MT analysis in zero- and few-shot settings (see Fig. 1).

Our main contributions and results are:

- **Synthetic data generation pipeline replicating real images.** Our publicly available pipeline generates realistic MT images along with corresponding ground-truth annotations. For tuning its internal parameters, it only requires real MT images (IRM or total internal reflection fluorescence (TIRF)) without the need for human annotations.
- **Synthetic benchmark dataset (`SynthMT`).** We apply this generation pipeline to real IRM images to construct and release the `SynthMT` dataset including instance masks for each MT. Its biological plausibility is assessed through a study with domain experts.
- **Comprehensive evaluation.** We conduct a reproducible benchmark evaluation of classical and state-of-the-art segmentation methods on `SynthMT`, including zero-shot and HPO-based few-shot adaptation experiments, establishing quantitative baselines.
- **Enabling fully automated segmentation.** We show that the newly released foundation model SAM3 achieves human-grade performance on MT images when parameter-optimized via `SynthMT`. Our synthetic data thus makes it possible to adapt this general-purpose model to the specific domain of *in vitro* MT microscopy, establishing a practical route toward fully automated segmentation for large-scale experiments and subsequent geometric analyses.

To facilitate progress, we publicly release the generation pipeline, dataset, and evaluation framework at `DATEXIS.github.io/SynthMT-project-page`, providing the community with tools to benchmark existing methods and accelerate method development.

2 Related work

Synthetic datasets for microscopy. Synthetic data has recently been explored as a way to overcome the lack of annotated microscopy datasets. For example, AnyStar [17] employs a generative model to produce synthetic 3D training volumes of star-convex objects used to train StarDist [18] models, enabling zero-shot generalization across organisms without retraining. While powerful for spherical targets such as nuclei, the star-convex shape assumption makes this approach inapplicable to filamentous structures such as microtubules (MTs), where continuity and curvature priors are essential.

The concurrently released MicSim_FluoMT [19] dataset represents an important step toward synthetic benchmarks for *in vivo* MTs. It provides microscopy-like images together with ground-truth segmentation masks, generated using Cytosim [20] for filament dynamics and ConfocalGN [21] for imaging rendering (see Fig. A.5a for examples). Their evaluation focuses on architectures trained entirely from scratch, but does not address the performance of existing foundation models.

Saguy et al. [22] propose a diffusion-based generative model for MT imaging that produces visually convincing synthetic images. However, their method does not produce corresponding

ground-truth labels, and hence the data cannot be used as a benchmark, and evaluation remains limited to qualitative inspection. Their focus lies on data augmentation for training Content-Aware Image Restoration (CARE) [23] models, rather than on providing a validated dataset for systematic comparison.

Several generative models synthesize medical or biological images by conditioning on segmentation masks or other geometric annotations. For example, SPADE [24] introduces spatially adaptive normalization to generate high-fidelity images from semantic masks, and BrainSPADE [25] applies this idea to biomedical MRI synthesis, showing that mask-to-image generation can support downstream segmentation. Similarly, ControlNet [26] guides diffusion models using additional input images, such as edges or segmentation maps, to specify where objects appear and how they are arranged, through an auxiliary control branch. SegGuidedDiff [27] extends this to anatomically informed medical image generation by concatenating segmentation masks at every denoising step. Despite architectural differences, all of these methods depend on manually annotated segmentation masks to guide the image generation process. In contrast, our approach does not need ground-truth annotations: it extracts statistical structure directly from MT images and produces synthetic images with segmentation masks through a simulation-based pipeline, removing the dependency on manual segmentation entirely.

DRIFT [28] recently introduced a recurrent approach for instance segmentation of filamentous objects, motivated by the way humans "pick and trace" filaments. To enable training and evaluation, they generate synthetic datasets of curved lines with varying widths and lengths. Although this constitutes a creative strategy to alleviate data scarcity, the resulting images are highly abstract — essentially binary line drawings — and deviate strongly from the visual statistics of real microscopy data (see Fig. A.5b for examples). Their evaluation is centered on a custom recurrent model and comparisons to prior filament-tracing methods (SOAX [29], SIFNE [30], [31], and [5]), but the data and benchmark are not designed to capture the variability in appearance of actual MT microscopy images, nor to evaluate the performance of foundation models for segmentation. In contrast, our image pipeline explicitly aligns with real images in terms of model features (by design) and the biological plausibility of the resulting `SynthMT` dataset is judged by domain experts. Additionally, we make this data openly accessible via Hugging Face for easy benchmarking or usage for training.

Generalist segmentation models. Recent work on foundation models for microscopy segmentation includes CellSAM [10], μSAM [11], and Cellpose-SAM [12], all of which adapt SAM [32] to microscopy tasks. SAM has since been updated to SAM2 [33] and SAM3 [9], adding video segmentation and concept-specific text prompting. CellSAM employs a vision transformer to localize bounding boxes that are then passed to an adapted SAM module, removing the need for manual prompts. μSAM adds a new decoder trained on microscopy data. Cellpose-SAM combines a modified SAM encoder with the flow field prediction and gradient tracking of Cellpose [34]. Furthermore, StarDist [18, 35] models are explicitly designed for segmenting star-shaped objects; although MTs are not star-convex, we include StarDist for completeness.

All of these models were benchmarked on a broad range of datasets [34, 36–59], most of which target nuclei or whole-cell segmentation. A few include elongated or tubular structures (e.g., filamentous bacteria in Omnipose [38], DeepBacs [39], or mitochondria in EM datasets), but none involve cytoskeletal MTs, where geometric continuity and sub-pixel-width features are critical.

Foundation models for general cellular and nuclear segmentation have reached a level of performance where further improvements are increasingly limited by annotation variability rather than model capacity. For example, Pachitariu et al. [12] report that Cellpose-SAM exceeds inter-annotator agreement on standard cell and nucleus benchmarks. In contrast, important challenges remain in settings like ours, where the objects of interest are elongated, filamentous structures, rather than compact objects like individual cells or nuclei.

Microtubule-specific methods. Several earlier works have proposed custom pipelines for MT segmentation, and in some cases also tracking, often involving their own (synthetic) datasets. These include FIESTA [8], SOAX [29], SIFNE [30], MTrack [60], DRIFT [28], the methods by Liu et al. [5, 61, 62] and Masoudi et al. [63], KnotResolver [64], the work by Laydi et al. [19], and TARDIS [7]. Classical approaches typically depend on high signal-to-noise ratio (SNR) data and require substantial manual input, keeping humans firmly in the loop. Machine-learning-based methods [5, 7, 19, 28, 61–63] attempt to address this limitation; however, their accessibility is often hindered by a reliance on proprietary MATLAB software rather than open-source Python environments. Furthermore, the specialized datasets used to develop these methods are not released to the public.

As a result, reproducibility and fair cross-method benchmarking remain limited. Crucially, none of these approaches have been evaluated against modern foundation models such as those built on SAM [32]. Within the MT community, a notable exception in terms of availability and generalization capabilities is TARDIS [7]. It combines CNN-based segmentation with graph-based instance segmentation and is released as open-source Python code that provides pretrained models, including one specialized for MTs in 2D total internal reflection fluorescence (TIRF) images. The authors of TARDIS report that compared to Amira [65], a commercial software suite for 3D visualization and analysis, annotation accuracy for MTs improves by 42%.

In this work, we choose FIESTA as a traditional baseline and add TARDIS as a domain-specific foundation model. Our pipeline and evaluation framework are implemented in Python with open-source dependencies, enabling reproducible, cross-lab benchmarking while avoiding proprietary tools such as MATLAB (except for FIESTA).

We document all evaluated methods in Appendix A.1, detailing implementation choices and practical requirements. Beyond benchmarking, `SynthMT` is designed as a practical resource for experimental biologists, providing accessible, validated tools for MT analysis.

3 Mathematical framework for synthetic image generation

We formalize our image generation pipeline as a two-step stochastic process that generates synthetic microscopy images $I \sim P_\theta(I)$ conditioned on a parameter set θ as illustrated in Fig. 2. These two steps are: (1) a purely geometric process imitating microtubule (MT) morphology, and (2) an imaging process that adapts these structures such that they resemble real images. This second step is largely inspired by the augmentation pipeline used in AnyStar [17].

(1) Microtubule geometry. Each MT is modeled as a polyline consisting of n segments $\{W_k\}_{k=0}^{n-1}$. Together, these segments define a curve $f : [0, L] \to \mathbb{R}^2$. Each W_k is specified by a segment length ℓ_k and a relative bend angle ϕ_k, where ℓ_k is sampled from a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ with $\mu, \sigma \in \theta$. Filament curvature is introduced through a stochastic evolution of the bend angles ϕ_k, which evolve as

$$\phi_k = \phi_{k-1} + B_k\, \eta_k, \qquad \eta_k \sim \text{Gamma}(\alpha, \beta), \tag{1}$$

where $\alpha, \beta \in \theta$ are shape and scale parameters, and B_k is a random variable with values in $\{\pm 1\}$ that determines whether the filament bends to the left or right at segment k. The sampling of B_k is determined through a flip probability $p_{\text{flip}} \in \theta$ and the maximum number of flips $n_{\text{flips}} \in \theta$,

$$P(B_k \neq B_{k-1}) = p_{\text{flip}}, \quad \sum_{k=0}^{n-1} |B_k - B_{k-1}| \leq 2 \cdot n_{\text{flips}}. \tag{2}$$

This yields a correlated random walk with persistence length governed by (α, β). The overall filament length L is drawn from a Gaussian distribution parametrized by θ, and the initial seed orientation ϕ_0 is sampled uniformly from $[0, 2\pi)$. Because segments are allowed to span only a few pixels, after step (2) they form smoothly curved filaments.

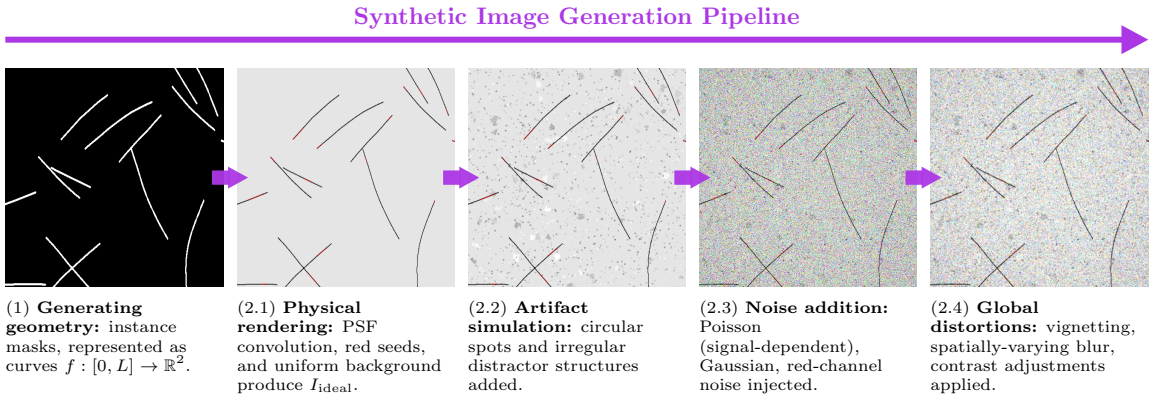


Figure 2. Our synthetic data generation pipeline produces realistic microtubule (MT) images with corresponding instance segmentation masks conditioned on a parameter set θ. (1) Generating geometry creates instance masks from geometric parameters (count, length, curvature) using polylines. (2.1) Physical rendering applies point-spread function (PSF) convolution to replicate optical properties, and adds red seeds and uniform background. (2.2) Artifact simulation introduces realistic distractor features (circular spots, irregular structures). (2.3) Noise addition models signal-dependent (Poisson) and signal-independent (Gaussian) noise sources. (2.4) Global distortions apply spatially-varying effects (vignetting, blur, contrast variations) to match real microscopy conditions. This approach enables the generation of labeled data that closely approximates experimental interference reflection microscopy (IRM) images, when its set of generation parameters θ is tuned accordingly (as explained in section 4).

(2) Image rendering. Given a curve f as constructed before (i.e., a piecewise-linear MT skeleton), we compute a binary mask M_S of shape $(h, w) \in \mathbb{N}^2$ that marks all pixels intersected by f. This mask is then convolved with the point-spread function (PSF) psf, and scaled by a contrast factor $A \in \theta$ and background intensity $B \in \theta$,

$$I_{\text{ideal}}(x) \coloneqq A\,(M_S * psf)(x) + B \tag{3}$$

for every pixel $x \in ([0, h) \times [0, w)) \cap \mathbb{N}^2$. Through convolution with the PSF, intensity spreads to neighboring pixels, which determines the filament's physical width in the rendered image.

Finally, noise and imaging artifacts are introduced via a stochastic operator ν_θ that combines multiple perturbation processes. These include signal-independent noise (e.g., additive Gaussian noise), signal-dependent effects (e.g., Poisson shot noise and multiplicative speckle), channel-specific distortions, spatially correlated background variations, and structured artifacts such as vignetting, blur, and random distractor spots. All of these random variables instantiated by the noise process are collected in θ as well. This yields the final image I through

$$I \coloneqq \nu_\theta(I_{\text{ideal}}). \tag{4}$$

Two-step stochastic process. Together, the complete pipeline defines a probability distribution $P_\theta(I)$ over synthetic images, where θ encodes the distribution of the number of MTs, their length distribution and curvature statistics, and further imaging parameters. In principle, this framework also extends to a temporal process by defining a Markov chain $\{I_t\}$ (e.g., including MT dynamics such as growing, shrinking, pause with Gaussian growth/shrinkage increments at each time step). In the present work, however, we restrict to the *single-frame setting*, that is, single-sampling $I \sim P_\theta$ without simulating multi-frame temporal dynamics.

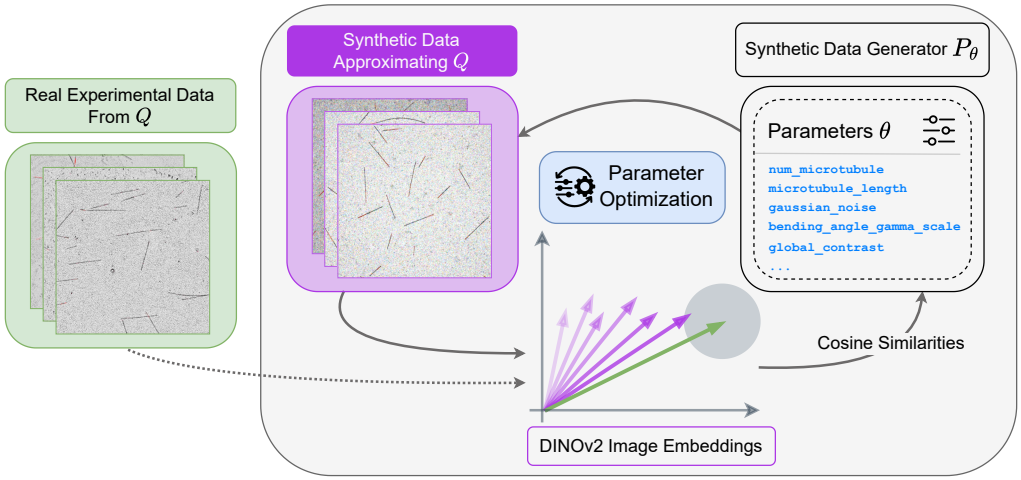


Figure 3. Optimizing θ aligns synthetic image distributions with real, annotation-free microscopy data. Real interference reflection microscopy (IRM) images (left) and synthetic images (center) are embedded using DINOv2. The parametric generator P_θ (right) creates images by sampling from distributions governing geometric properties (filament count, length, curvature) and imaging characteristics (PSF, noise, artifacts, contrast, distortions), all controlled by θ. An optimization loop iteratively refines θ by maximizing cosine similarity between real and synthetic embeddings, ensuring that synthetic images match the statistical properties and visual characteristics of experimental data.

4 Methods

We describe the creation of `SynthMT`, the first dataset of annotated, synthetic *in vitro* microtubule (MT) microscopy images that closely mimic the visual characteristics of real experimental data. Furthermore, we provide the protocol for assessing their perceptual realism using domain expert judgments and for evaluating fully automated segmentation methods on both `SynthMT` and a small labeled set of real images.

Creating `SynthMT` and annotating real data

With the mathematical formulation of the image generation process in place (see section 3), we summarize the full pipeline to construct `SynthMT`, illustrated in Fig. 3: For a given reference distribution Q consisting of real, unlabeled images, the generator P_θ iteratively creates images $I \sim P_\theta(I)$ that are compared to samples from Q. For this comparison, we embed the images using DINOv2 [66], a pre-trained state-of-the-art vision transformer, designed to generate rich visual features. The pipeline parameters θ are then tuned to minimize the resulting embedding distance, without requiring ground-truth annotations.

Real data. The real images for our empirical target distributions Q_i were provided by two wet labs that routinely investigate reconstituted MTs. In total, we obtained 44 real IRM videos, each containing between 30 and 360 frames, spanning multiple experimental conditions and microscope setups, and capturing substantial diversity in MT geometry and visual properties such as contrast, noise, pixel intensity distributions, and background textures. Four representative examples of this variability are shown in Figure 4. More information about this real data can be found in appendix A.3. These raw videos were histogram-normalized and background-subtracted[1], with no additional processing applied.

[1] As detailed in imagejdocu.list.lu/gui/process/subtract_background.

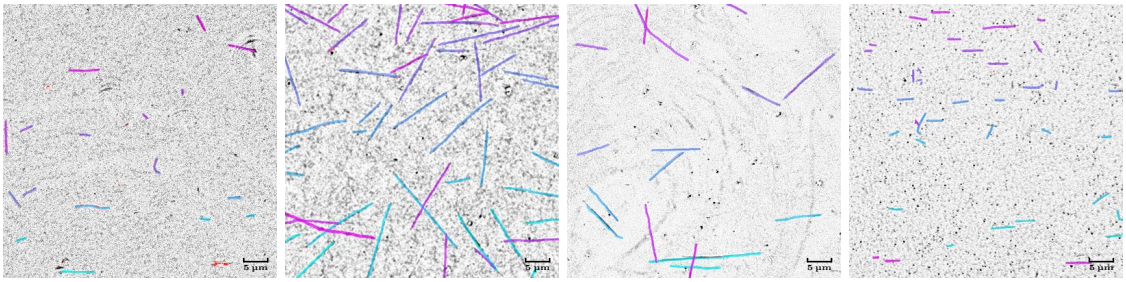


Figure 4. Real IRM images with human ground-truth labels (for evaluation only). Four exemplary frames from the 66 video crops of size 512×512 used to establish target distributions $\{Q_i\}_{i=1}^{66}$ (each containing 10 randomly sampled frames). Each image shows individual *in vitro* MTs growing from stabilized seeds under different experimental conditions, exhibiting natural variation in quantity, length, curvature, overlapping MTs, contrast, noise characteristics, filament density, and background properties. These target distributions act as references in the optimization process (Fig. 3), where DINOv2 embeddings of Q_i guide the synthetic data generation. Ground-truth labels from one annotator are overlaid to illustrate the filament structures present in the real data; they are used solely for later method evaluation and are not required for generating the synthetic dataset `SynthMT`.

From these videos, we take random 512×512 pixel crops. We manually discard crops containing significant imaging artifacts or distortions that would typically be excluded from a standard analysis, yielding a final set of 66 smaller videos. Each cropped video is treated as a distinct target domain Q_i, $i = 1, \ldots, 66$, from which we sample 10 random frames to serve as reference images for optimization. Importantly, for later method evaluations on unseen, real data, one of these frames from each crop is manually annotated by two independent human annotators using Label Studio [67]. In the analyses, we designate one annotator's labels as ground truth and treat the other annotator's labels as predictions, yielding an inter-annotator agreement score that quantifies the inherent variability of manual MT segmentation. This human baseline provides a reference point for assessing whether automated methods achieve human-level accuracy. Fig. 4 shows four exemplary frames together with the annotations used as ground truth.

Optimizing θ. To align statistical and visual properties of the generated synthetic images with those of the selected real microscopy data $\{Q_i\}_{i=1}^{66}$, we utilize the tree-structured Parzen estimator (TPE) [68] with 1000 iterations for each $i = 1, \ldots, 66$. TPE has repeatedly been shown to outperform alternative optimization algorithms [69, 70] and has become the de-facto standard in recent Hyperparameter Optimization (HPO) frameworks, where it is commonly used as the default sampling strategy [71, 72].

As a shared embedding space for comparing generated and real images, we choose the self-distillation method DINOv2 [66], which correlates well with human similarity judgments [73]. Moreover, as shown by Bolya et al. [74], the most informative embeddings for perceptual similarity are often obtained from intermediate rather than the final layers. This observation aligns with our findings: features from the fifth DINOv2 layer, denoted DINOv2_5, balance abstraction and sensitivity to the subtle, fine-grained image structures that define IRM MT data. Altogether, the optimization maximizes the cosine similarity

$$\text{sim}(I, J) \coloneqq \text{sim}_{\cos}\big(\text{DINOv2}_5(I), \text{DINOv2}_5(J)\big) \tag{5}$$

between generated images $I \sim P_\theta(I)$ and real samples J from Q_i in the feature space. We aggregate the similarity between synthetic and real frames using the *maximum similarity* across the 10 references in Q_i as the objective. This guarantees that the chosen parameters yield synthetic images that are plausible for at least one representative frame in Q_i. Finally, we retain the top $k = 10$ candidate solutions $\{\theta_i^j\}_{j=1}^{10}$ that yield the highest aggregated similarity. `SynthMT` is the product of

drawing 10 images for each $1 \leq j \leq 10$ and $1 \leq i \leq 66$, yielding 6600 images of size 512×512, examples of which are shown in Fig. A.4. The masks are stored as 3D TIFF images, where each slice is the mask assigned to a single instance. This preserves overlapping objects correctly and resembles the output of SAM (see appendix A.1).

`SynthMT` is publicly hosted on Hugging Face under a stable organization account at `huggingface.co/HTW-KI-Werkstatt/SynthMT`. The dataset card provides versioning, update logs, and contact information, and we will maintain `SynthMT` by releasing documented updates if errors are discovered or extensions are added. Users can contact the corresponding authors to report issues, request corrections, or raise takedown concerns.

Assessing the perceptual realism of `SynthMT`

We conduct a study with $n = 6$ domain experts to evaluate the quality and realism of our synthetic microscopy images. To enable a direct comparison, we include 10 real images, 10 synthetic images from `SynthMT`, and 10 images from the only openly available synthetic dataset for *in vitro* MTs, DRIFT [28]. All 30 images are presented in randomized order and each image is shown exactly once to each participant. The study is administered as a web-based interface (see appendix A.5 for details). Two of the six participants are co-authors of this paper; neither was involved in the data generation pipeline.

Experts rate each image on five aspects $\mathcal{D}$ using a 7-point Likert scale (1 = very poor, 7 = excellent): (a) Shape and general appearance of the MTs (structural fidelity), (b) Realism of the background, (c) Lighting, intensity, and blurring realism, (d) Noise pattern realism, and (e) Overall quality. The five aspects are motivated by quality evaluation criteria for microscopy imaging [75], while the use of aspect-specific Likert ratings follows recent generative evaluation studies emphasizing structured perceptual assessment of image realism [76, 77]. We use this setup rather than pairwise forced-choice comparisons [78–81] to obtain aspect-specific judgments that also inform future pipeline improvements.

Distributional comparison of ratings. Experts differ systematically in their use of Likert scales (e.g., lenient vs. strict raters). To remove these fixed rater effects, we z-normalize each expert's scores prior to aggregation. In particular, for expert $e = 1, \ldots, 6$ and aspect dimension $d \in \mathcal{D}$, we compute the within-expert mean $\mu_{e,d}$ and standard deviation $\sigma_{e,d}$ across all images, and transform raw ratings $r_{e,d}$ into normalized scores

$$r_{e,d}^{\text{norm}} := \frac{r_{e,d} - \mu_{e,d}}{\sigma_{e,d}}. \tag{6}$$

We visualize the normalized ratings for real and synthetic images using violin plots per dimension (Fig. 5), displaying the full rating density, median, and interquartile range. Because normalization is done per expert, these visualizations capture differences in perceived quality rather than rater-specific scale usage.

Evaluations on `SynthMT` and real data

As discussed in the related work (section 2), we focus on the classical algorithm FIESTA [8] as a baseline, and compare to the foundation models SAM [32], SAM2 [33], μSAM [11], CellSAM [10], Cellpose-SAM [12], and SAM3 [9] (listed chronologically by release date). StarDist [35] has an improper geometric bias for this task but is considered for completeness, and TARDIS [7] is added as the only MT-specific foundation model. The implementation and parameters of each method are detailed in appendix A.1. All methods are run in their "automated" mode without further input from our side. The only exceptions are SAM3Text, for which we use our manually specified default prompt "*thin line*", and FIESTA, where we set the *full width at half maximum* (FWHM) parameter to 3. We ignore other methods that cannot work in a fully automated fashion or are not publicly

available (e.g., Masoudi et al. [63] or MTrack [60]). Initial tests with a StarDist model trained on AnyStar [17] showed complete failure on our 2D data. The authors later verified[2] that the method is inherently ineffective in this setting.

We evaluate two categories of metrics: segmentation quality and downstream biological performance, as detailed below.

Preprocessing. We disable method-specific preprocessing steps and instead gather all of those in our own preprocessing function. For details, see appendices A.1 and A.2. This ensures that all methods share the same preprocessing capabilities during hyperparameter tuning (see HPO below).

Method outputs. The methods differ in output format, which becomes crucial in postprocessing and metric computation. In particular, SAM, SAM2, and SAM3 return lists of 512×512 boolean masks, one per instance, which resembles the format of our labels for `SynthMT` and the real images we annotated. StarDist, μSAM, CellSAM, and Cellpose-SAM produce 512×512 integer arrays, where 0 denotes background and each positive integer corresponds to a unique instance. We convert these into our format by splitting the integer-labeled arrays into separate boolean masks, one per nonzero label. FIESTA and TARDIS do not output segmentation masks as arrays. Instead, they return lists of anchor points $(p_k)_k$, one for each instance. Whenever we report instance lengths, for mask-based outputs M our proxy is the skeleton pixel count $L = |S(M)|$, where $S(M)$ is the thinned, one-pixel-wide binary skeleton. For anchor points $(p_k)_k$, we sum Euclidean distances, $L = \sum_k \|p_{k+1} - p_k\|_2$.

Postprocessing. To ensure comparability across methods, we also disable all method-specific postprocessing routines and apply a unified postprocessing step implemented by us as follows. Predicted instances are filtered by length and area (for anchor-point methods, only length is used), requiring them to fall within dataset-specific minimum and maximum values. In our benchmark, these thresholds are computed automatically from the ground-truth labels of `SynthMT` resp. the real data. In practical applications, however, domain experts typically know the plausible range of filament lengths and areas for their particular imaging setup, making such filtering a natural and interpretable step. By removing implausible tiny fragments and related artifacts, this unified postprocessing ensures more stable and meaningful comparisons across segmentation methods.

Segmentation metrics. Accurately evaluating segmentation of thin, elongated MTs is non-trivial: classical pixel-overlap metrics such as Intersection over Union (IoU) over-penalize small transversal misalignments while under-representing errors in biologically relevant properties like filament length, continuity, or curvature. For MTs that are only a few pixels wide, a slight lateral shift or minor width disagreement can disproportionately reduce IoU, even if the traced centerline (and thus the inferred length) is essentially correct. To emphasize geometry rather than raw area agreement, we adopt the Skeleton Intersection over Union (SKIoU) [61] metric.

For anchor-point outputs (i.e., outputs from FIESTA and TARDIS), we compute a corresponding instance mask by fitting a spline with smoothing condition `s=0`[3] through these points. We then represent this spline as a binary mask M to match the format of the other methods.

Given predicted and ground-truth instance masks M_{pred} and M_{gt}, and again denoting $S(M)$ as the skeleton of a mask M, their SKIoU is defined as

$$\text{SKIoU} \coloneqq \frac{2\,|S\,(M_{\text{pred}} \cap M_{\text{gt}})|}{|S\,(M_{\text{pred}})| + |S\,(M_{\text{gt}})|}. \tag{7}$$

This provides a symmetric, length-normalized overlap over skeleton pixels: it attains 1 only when centerlines coincide, and it drops sharply when filaments are missing (under-segmentation),

[2] `https://github.com/neel-dey/AnyStar/issues/4#issuecomment-3362693166`
[3] See SciPy's `splprep` function.

spuriously split or merged (topology errors), or substantially curved differently (shape distortion). Because skeleton extraction collapses width, SKIoU becomes robust to small diameter inconsistencies and lateral blurring artifacts that are common in IRM imaging.

For evaluation, we compute several common object detection and instance segmentation metrics based on the SKIoU values. We report the mean SKIoU of an image, calculated as follows. For each ground-truth instance, we find the best-matching prediction and record its SKIoU. If no matching prediction is found (a false negative), the score for that ground-truth instance is 0. The mean SKIoU value thus reflects segmentation quality while also penalizing for missed objects. While informative, this metric does not penalize false positives.

To provide a more complete picture, we also report Average Precision (AP) and F1 scores. AP is calculated as the mean of AP values at SKIoU thresholds from 0.5 to 0.95 in steps of 0.05, summarizing detection and segmentation accuracy. The F1 scores at specific SKIoU thresholds of 0.5 (F1@0.50) and 0.75 (F1@0.75) represent the harmonic mean of precision and recall. Comparing F1@0.50 and F1@0.75 allows us to gauge the consistency of a method's predictions; a small drop-off between these values indicates that the method maintains high localization accuracy even at stricter evaluation criteria.

Downstream biological metrics. Segmentation scores alone do not answer whether a method performs well at recovering biologically relevant filament statistics. To accommodate that, we report downstream biological metrics derived from instance geometries such as **counting** the number of predicted instances and gathering their computed **lengths** (as explained above). Furthermore, we collect the average **curvature** of each predicted instance as follows: Anchor-point outputs are converted into masks by connecting the points via straight lines. Then, given any instance mask (directly from the method or converted from anchor points), we order skeletonized pixels by greedy nearest-neighbor and fit a parametric spline $(x(u), y(u))$ with smoothing condition `s=0` to preserve geometry. Finally, we compute the average curvature of this instance as the mean of the curvatures $\kappa(u)$, where

$$\kappa(u) \coloneqq \frac{|x'(u)y''(u) - y'(u)x''(u)|}{\left(x'(u)^2 + y'(u)^2\right)^{3/2}}. \tag{8}$$

With these statistics at hand, we report mean and standard deviation across all images for counts, lengths, and curvatures. While this offers an intuitive understanding of systematic biases in the predictions relative to the ground-truth values, comparing distributions solely by their means and standard deviations is not sufficient. We therefore also compare the predicted and ground-truth distributions of length and curvature using normalized histograms with shared linear bins. Their dissimilarity is quantified through the Kullback–Leibler divergence (KL divergence) [82]

$$\mathrm{KL}(P \parallel Q) \coloneqq \sum_i P_i \log\left(\frac{P_i}{Q_i}\right), \tag{9}$$

where P and Q denote the predicted and ground-truth discrete distributions, respectively (lower is better).

Computational efficiency. We report computational throughput measured as processed images per second ("Img/s." in Table 1). All results are obtained under sequential, unbatched inference to reflect per-sample efficiency. Experiments are conducted on a single NVIDIA A100 GPU; FIESTA is evaluated separately on a MacBook M3 Pro.

Zero-shot setting. We evaluate all methods with their default parameters (as detailed in appendix A.1) on `SynthMT` and the unseen, real data.

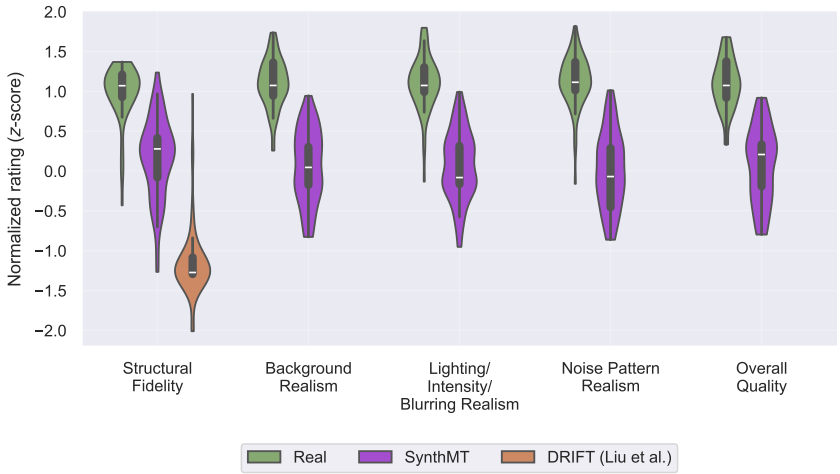


Figure 5. Domain experts confirm perceptual quality of `SynthMT` images. Violin plots show z-normalized ratings of $n = 6$ domain experts across five quality dimensions for real IRM images and synthetic images from `SynthMT` and DRIFT [28] (10 images each, 30 in total). Each violin displays the full distribution, median (white line) and interquartile range (thick bar). Ratings were collected on a 7-point Likert scale. DRIFT permits evaluation only of structural fidelity due to its black-and-white outputs (see exemplary images in Fig. A.5b). `SynthMT` images score higher than DRIFT on this dimension, indicating that parameter-optimized synthesis yields structures that more closely resemble real microscopy data. A measurable gap to real images persists across all dimensions. Nevertheless, experts rate the backgrounds, lighting, and noise patterns of `SynthMT` as internally coherent and plausibly aligned with real IRM, in contrast to DRIFT's limited realism.

HPO-based few-shot setting. Furthermore, we automatically adapt both method and preprocessing hyperparameters using only 10 random samples from `SynthMT`. We conduct this HPO via the tree-structured Parzen estimator [68] with 1000 trials to maximize mean SKIoU. We report the resulting optimal configurations in appendices A.1 and A.2, and discuss optimization trajectories and parameter importance in appendix A.7.

5 Results

The resulting `SynthMT` dataset, as described in section 4, contains 6600 synthetic images with a varying number and shape of microtubules (MTs), and diverse backgrounds and noise. They are accompanied with segmentation masks for every MT present in each image. Moreover, two independent human annotators created instance segmentations for 66 real interference reflection microscopy (IRM) images.

Perceptual realism of `SynthMT`

The images in `SynthMT` were constructed to match real IRM-like data in the fifth layer representation of the DINOv2 model (see Eq. (5)). For adoption of this dataset within the community, perceptual adequacy to humans matters as well. Here, we present the results of the expert study outlined in section 4.

While a gap relative to real images remains, the distributions in Fig. 5 nevertheless demonstrate that `SynthMT` occupies an intermediate but substantive region between DRIFT [28] and real data: `SynthMT` is geometrically closer to the real domain, perceptually coherent across background, lighting, and noise, and stable across experts. This establishes `SynthMT` as the most realistic synthetic alternative currently available.

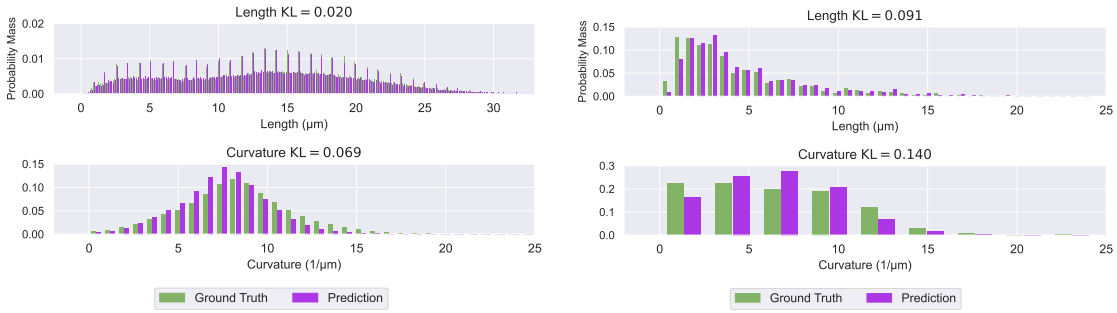
Table 1. Results on `SynthMT` signal unprecedented segmentation performance of the new SAM3 model. For each method, we report segmentation metrics including Skeleton Intersection over Union (SKIoU) [61], Average Precision (AP) (mean over SKIoU thresholds), and F1 of SKIoU at 0.50 and 0.75. In addition, we report biological downstream metrics derived from counts per image, and length and curvature measurements per MT. These values are reported as mean ± standard deviation across all images and can be directly compared to the ground-truth statistics (last row). To capture distributional differences in greater detail, we further report the Kullback–Leibler divergence (KL divergence) between the predicted and ground-truth distributions of length and curvature. All methods are evaluated zero-shot using their default parameters. Rows marked with "+ HPO" show performance after Hyperparameter Optimization (HPO) maximizing SKIoU on 10 randomly sampled synthetic `SynthMT` images, illustrating few-shot adaptation potential. SAM-family models are run in Automatic Instance Segmentation (AIS) mode; SAM3Text uses SAM3's text prompt mode (default prompt: "*thin line*"). For each column, the best-performing default and tuned methods are highlighted in **bold**, and ↑ (↓) signals higher (lower) is better.

		Segmentation Metrics				**Biological Downstream Metrics**				
	Img/s ↑	**SKIoU ↑**	**AP ↑**	**F1 ↑**		**Count**	**Length**		**Curvature**	
Model				@.50	@.75	n/img	μm/MT	KL ↓	μm^{-1}/MT	KL ↓
Baseline										
FIESTA [8]	0.21	0.12	0.12	0.22	0.18	99.63 ± 56.0	1.92 ± 4.3	5.03	6.74 ± 12.6	0.997
+ HPO	0.16	0.24	0.25	0.39	0.30	38.80 ± 17.1	3.64 ± 5.9	3.74	6.90 ± 10.2	0.706
Foundation Models for Microscopy										
StarDist [35]	4.93	0.01	0.04	0.00	0.00	1.85 ± 1.5	10.14 ± 2.9	0.69	7.67 ± 2.1	0.196
+ HPO	**8.64**	0.31	0.34	0.36	0.15	20.36 ± 11.0	2.65 ± 1.4	1.95	7.79 ± 7.8	0.183
TARDIS [7]	0.32	0.45	0.48	0.59	0.45	17.15 ± 4.9	7.71 ± 6.5	0.56	8.28 ± 3.8	**0.019**
+ HPO	0.34	0.48	0.55	0.57	0.41	12.33 ± 5.1	8.35 ± 6.3	0.41	8.50 ± 3.6	0.031
μSAM [11]	**7.44**	0.02	0.13	0.03	0.02	0.50 ± 1.0	6.03 ± 5.3	0.88	8.16 ± 8.1	0.130
+ HPO	6.56	0.66	0.73	0.74	0.64	14.58 ± 22.9	7.22 ± 6.3	1.24	8.68 ± 10.5	0.132
CellSAM [10]	2.63	0.56	0.79	0.66	0.58	7.95 ± 4.6	9.57 ± 5.5	0.19	8.12 ± 4.1	0.021
+ HPO	2.50	0.59	0.69	0.68	0.59	$\mathbf{11.60 \pm 6.7}$	9.33 ± 5.7	0.21	7.97 ± 4.5	0.031
Cellpose-SAM [12]	2.62	0.27	0.68	0.37	0.36	2.96 ± 3.2	$\mathbf{12.54 \pm 5.5}$	0.12	$\mathbf{8.45 \pm 4.2}$	**0.019**
+ HPO	2.59	0.65	0.75	0.76	0.65	10.66 ± 6.0	10.39 ± 6.1	0.12	$\mathbf{8.35 \pm 4.9}$	**0.012**
General Purpose Foundation Models										
SAM [32]	0.72	0.37	0.46	0.45	0.44	20.46 ± 17.3	3.91 ± 5.9	3.90	9.09 ± 22.5	0.700
+ HPO	0.52	0.16	0.16	0.25	0.24	74.90 ± 37.8	2.01 ± 4.7	5.45	7.52 ± 18.4	0.912
SAM2 [33]	0.96	0.01	0.05	0.01	0.01	0.06 ± 0.3	3.65 ± 2.8	1.62	6.90 ± 4.1	0.269
+ HPO	0.62	0.66	0.73	0.74	0.72	11.67 ± 5.3	11.89 ± 6.8	0.04	8.18 ± 4.1	0.014
SAM3 [9]	1.00	0.00	0.00	0.00	0.00	0.01 ± 0.1	3.02 ± 1.1	2.77	9.50 ± 3.9	0.489
+ HPO	0.78	0.54	0.58	0.64	0.64	23.69 ± 26.8	6.64 ± 8.1	2.90	8.29 ± 13.6	0.364
SAM3Text [9]	4.26	**0.85**	**0.86**	**0.91**	**0.90**	$\mathbf{13.03 \pm 6.5}$	11.74 ± 7.1	**0.07**	7.54 ± 3.3	0.063
+ HPO	3.97	**0.93**	**0.95**	**0.97**	**0.96**	11.73 ± 5.6	$\mathbf{13.19 \pm 6.7}$	**0.02**	7.56 ± 3.0	0.069
Ground Truth	-	1	1	1	1	11.28 ± 5.1	13.35 ± 6.7	0	8.42 ± 3.9	0

In the following sections, we show that this perceptual adequacy has functional consequences too: methods' strengths on `SynthMT` transfer to unseen, real IRM data, and tuning via Hyperparameter Optimization (HPO) on only 10 images from `SynthMT` further improves performance on both synthetic and real data for many methods.

Benchmarking `SynthMT` and real data

Beyond our generation pipeline used to produce `SynthMT`, Table 1 depicts another main contribution of this work: benchmark results for all nine introduced methods across the 6600 synthetic images contained in `SynthMT`. Additionally, we present results on unseen, real data in Table 2, and a qualitative analysis in Fig. 8. The benchmark is fully reproducible with the code available at `github.com/ml-lab-htw/SynthMT`.



(a) SAM3Text + HPO on SynthMT.

(b) SAM3Text + HPO on unseen real data.

Figure 6. SAM3Text + HPO closely matches ground-truth MT length and curvature distributions across scales and datasets. Normalized histograms compare predicted and ground-truth distributions for MT length (top row) and curvature (bottom row) on (a) `SynthMT` and (b) unseen, real data. SAM3Text + HPO preserves both low and high values across the full range of lengths and curvatures, as reflected by low KL divergence values computed from these histograms. Distributions for the other methods on `SynthMT` are shown in Figs. A.8 and A.9.

From FIESTA to SAM3Text. From Table 1 we find that all foundation models perform substantially better than the traditional, deep learning-free baseline FIESTA [8] which is still state-of-the-art in many labs. As one of the fastest methods, SAM3Text with tuned hyperparameters emerges as the candidate of choice for almost all tasks. Notably, on `SynthMT` it is the only evaluated method that achieves sub-micrometer accuracy in MT length estimation. Fig. 6 visualizes the close alignment between its predicted and ground-truth distributions of MT lengths and curvatures across scales. The only task where SAM3Text slightly lags behind is curvature estimation.

Improvements through HPO. In the few-shot (HPO) setting, we optimized method hyperparameters by maximizing SKIoU using only 10 random images from `SynthMT`. We find that this often greatly improves segmentation, except for TARDIS and CellSAM, which seem rather agnostic to parameters, and SAM, which shows a substantial drop in performance, indicating overfitting.

Localization robustness. SAM3Text not only achieves the strongest segmentation scores overall, but also stands out for its internal consistency across complementary metrics. Its simultaneously high AP, SKIoU, F1@0.50, and F1@0.75 scores indicate that detected MTs are both reliably found and accurately localized, with little degradation when stricter overlap requirements are imposed. In particular, the stable performance from F1@0.50 to F1@0.75 suggests that SAM3Text predictions are geometrically precise. Other models from the SAM family exhibit a similar robustness pattern, albeit at a lower absolute performance level. In contrast, StarDist and TARDIS show a pronounced drop at F1@0.75, revealing inconsistencies between detection and precise alignment. A different failure mode is observed for CellSAM and Cellpose-SAM, which achieve comparatively high AP values while lagging behind in SKIoU. This discrepancy indicates that, although many MTs are detected, their spatial alignment is often imprecise, resulting in poorly localized segmentations. Such behavior is consistent with a higher incidence of missed or fragmented filaments in the later qualitative inspections.

Segmentation metrics as proxies for biological downstream performance. Overall, improvements in segmentation metrics translate into better performance on biological downstream tasks. However, this relationship breaks down for methods whose segmentation quality is

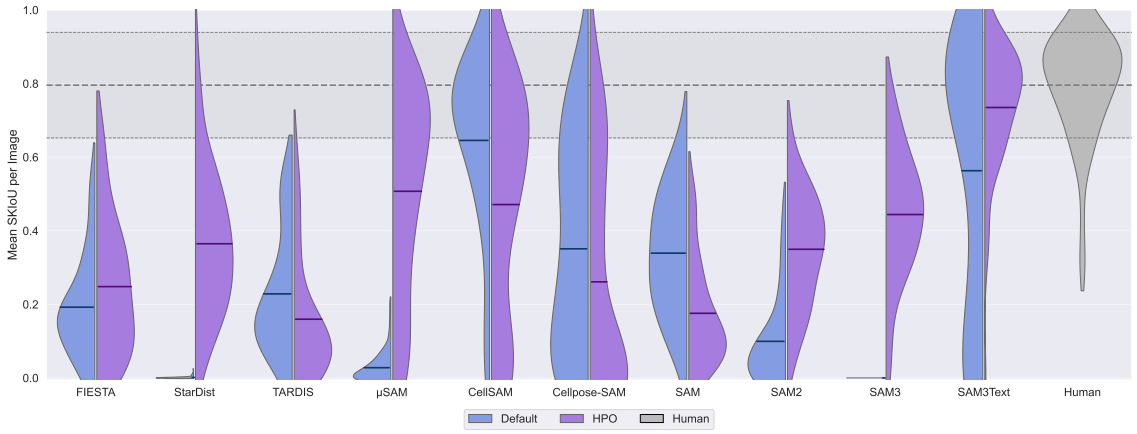


Figure 7. Only SAM3Text + HPO reaches human segmentation performance on unseen, real IRM data. Split violin plots show the distribution of per-image SKIoU scores ($n = 66$) for each method. The left side of each violin (blue) represents the default configurations, while the right side (purple) shows the performance after HPO on 10 random images from `SynthMT`. Each violin includes a horizontal line indicating the mean SKIoU across all images. The human performance is shown as a solid gray violin for reference. Its mean and standard deviation values are indicated by horizontal lines across the plot. The plot shows that the optimized SAM3Text matches the human inter-annotator baseline. While other methods also improve with HPO, none demonstrates the top-tier performance of SAM3Text. Notably, CellSAM already approaches human-level performance in its default configuration, but exhibits decreased performance after HPO.

fundamentally insufficient. For methods with near-zero segmentation scores, such as StarDist, μSAM, SAM2, and SAM3, even large gains in segmentation metrics achieved through HPO do not lead to meaningful improvements in biological performance and in some cases even degrade it. Apart from that, μSAM shows that poor performance on segmentation metrics is not immediately an indicator of bad downstream results. We hypothesize that this is because the default versions of SAM2, SAM3, Cellpose-SAM, and μSAM did not generate any predictions on up to 60% of the 6600 images, while their HPO versions did for almost all. Hence, μSAM identified only a small fraction of all MTs, yet statistically reproduced the correct distributions of MT geometries. We refer to Figs. A.8 and A.9 for these distributions for all methods, illustrating the curvature distribution match of μSAM, even with poor predictions for count and length. Still, relying on downstream biological measurements when segmentation is weak comes at a risk, as they may be computed on noisy or incorrectly identified structures.

Models made specifically for microscopy are strong on this task. Although SAM3Text consistently performs well, it is important to note that Cellpose-SAM, as a foundation model specifically released for microscopy tasks, also demonstrates strong performance on downstream metrics; even in its default setting (where it does not find any instances in 27% of the images). While TARDIS, μSAM, and CellSAM lag behind (in line with the findings of Pachitariu et al. [12]), they easily beat generalist models such as SAM or SAM2 on these biologically relevant tasks. As expected, StarDist, although one of the fastest with ≈ 5 images per second, performs the worst with its default settings, since it is architecturally made for star-convex shapes. However, after tuning its hyperparameters it still outperforms FIESTA.

Tests on unseen, real IRM data with human baseline further confirm SAM3Text. Analyzing performances on unseen, real data and comparing them to a human baseline allows us to

Table 2. SAM3 signals capabilities of fully automated MT analysis on unseen, real IRM data. Analogous to Table 1, we evaluate methods on biological downstream metrics. Additionally, we report an inter-annotator baseline by treating the annotations of a second, independent human annotator as predictions and comparing them against the ground-truth annotations of the first annotator, thereby quantifying inherent variability in manual segmentation. The strategy of adapting hyperparameters of SAM3Text through an HPO on 10 randomly sampled `SynthMT` images leads to human-grade accuracy. Models that are on par with or surpass the human inter-annotator baseline are listed in **bold**.

	Biological Downstream Metrics				
	Count	**Length ↓**		**Curvature**	
Model	n/img	μm/MT	KL ↓	μm^{-1}/MT	KL ↓
Baseline					
FIESTA [8]	93.05 ± 77.4	1.15 ± 1.4	1.39	$\mathbf{6.80 \pm 11.8}$	0.27
+ HPO	27.09 ± 28.0	1.87 ± 2.2	0.97	$\mathbf{6.70 \pm 11.1}$	0.28
Foundation Models for Microscopy					
StarDist [35]	0.68 ± 0.9	8.56 ± 2.4	1.46	7.66 ± 2.1	0.54
+ HPO	33.76 ± 13.1	1.68 ± 1.1	0.61	8.19 ± 12.9	0.35
TARDIS [7]	10.20 ± 8.2	4.78 ± 3.5	0.32	$\mathbf{7.42 \pm 4.9}$	**0.10**
+ HPO	8.85 ± 8.2	5.85 ± 4.4	0.41	7.71 ± 5.5	0.12
μSAM [11]	1.27 ± 1.6	4.63 ± 3.4	0.29	6.64 ± 4.0	0.26
+ HPO	67.26 ± 117.9	1.72 ± 2.3	1.10	7.71 ± 14.0	0.29
CellSAM [10]	21.29 ± 14.3	3.88 ± 2.7	**0.07**	$\mathbf{7.08 \pm 5.9}$	**0.09**
+ HPO	16.73 ± 16.3	$\mathbf{4.34 \pm 3.0}$	**0.08**	8.25 ± 10.1	0.14
Cellpose-SAM [12]	12.27 ± 16.4	3.57 ± 2.7	**0.05**	$\mathbf{7.03 \pm 6.5}$	0.18
+ HPO	13.00 ± 18.8	4.05 ± 3.3	**0.07**	7.92 ± 9.4	0.09
General Purpose Foundation Models					
SAM [32]	28.55 ± 12.6	2.24 ± 2.6	0.79	$\mathbf{6.80 \pm 10.0}$	0.16
+ HPO	108.26 ± 29.6	1.43 ± 2.1	1.39	$\mathbf{6.62 \pm 12.2}$	0.15
SAM2 [33]	2.73 ± 2.6	2.39 ± 1.6	0.37	$\mathbf{7.43 \pm 7.8}$	0.23
+ HPO	48.79 ± 16.9	3.07 ± 3.3	0.44	7.74 ± 11.5	0.15
SAM3 [9]	0.03 ± 0.2	1.83 ± 0.4	1.45	8.29 ± 2.1	0.94
+ HPO	16.59 ± 11.0	$\mathbf{4.28 \pm 3.5}$	0.10	7.71 ± 8.6	0.07
SAM3Text [9]	14.98 ± 15.4	5.41 ± 3.9	0.21	$\mathbf{7.17 \pm 5.3}$	0.25
+ HPO	$\mathbf{25.08 \pm 16.9}$	4.69 ± 3.5	**0.09**	$\mathbf{6.62 \pm 4.9}$	0.14
Inter-annotator	25.29 ± 18.6	4.56 ± 3.4	0.09	7.43 ± 7.3	0.11
Ground Truth	23.91 ± 18.1	4.39 ± 3.6	0	6.91 ± 6.6	0

put the previously presented numbers on the synthetic `SynthMT` dataset into context.

Fig. 7 demonstrates that SAM3, provided with a simple text prompt ("*thin line*") and with optimized hyperparameters using 10 images from our synthetic dataset `SynthMT`, maintains strong performance on unseen, real images and achieves segmentation quality comparable to human performance. It reaches a mean SKIoU of 0.74 ± 0.18, compared to a human of 0.8 ± 0.14, placing the model's performance within the variability observed between human annotators. Although other methods also benefit from HPO, none reaches this level of performance. The methods that do not improve through HPO are the same as those found above in the analysis on `SynthMT` (TARDIS, CellSAM, SAM). Cellpose-SAM is the only method that improves on synthetic data, but worsens on unseen, real data.

Table 2 confirms these findings through the quantitative biological metrics, where SAM3Text consistently improves with HPO. It matches and slightly surpasses human inter-annotator

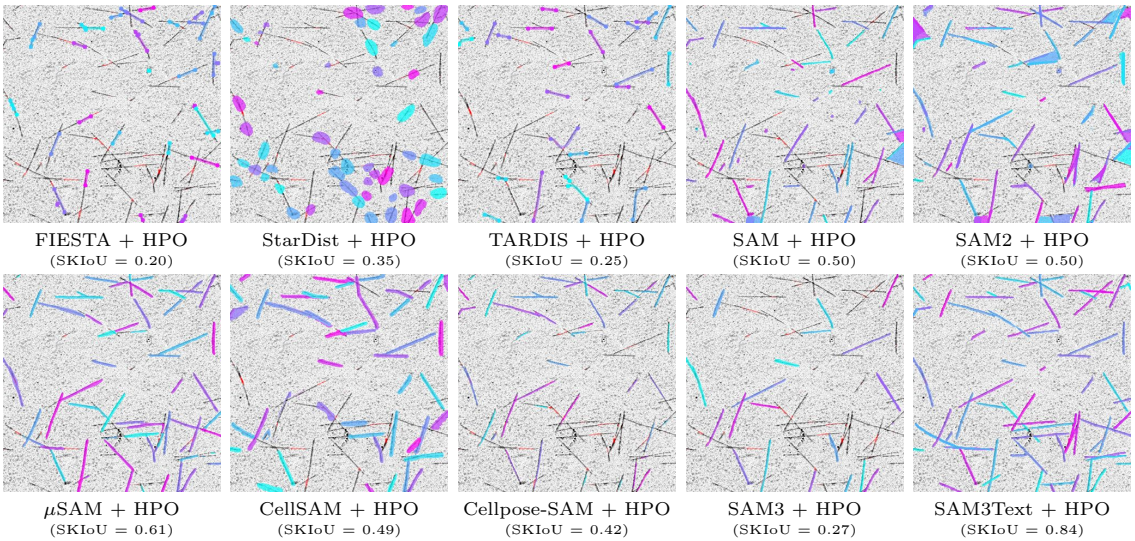


Figure 8. Qualitative comparison on an unseen, real-world *in vitro* reconstituted MT assay. For each method, we show predictions after HPO on 10 synthetic images from `SynthMT` (few-shot setting). The selected real image is particularly challenging, as it contains many intersecting MTs and exhibits a low signal-to-noise ratio (SNR), exposing a wide range of failure modes. For anchor-point methods such as FIESTA and TARDIS, only the first and last predicted points per instance are shown for visual clarity. Underneath each image we report the mean SKIoU value for this specific image, in order to correlate it with a visual impression. SAM3Text clearly performs best in this setting, while all other methods show limitations that may hinder their suitability for large-scale fully automated analysis. For more comparisons and dynamic exploration of this kind, we refer to our project page at `DATEXIS.github.io/SynthMT-project-page`.

agreement in counting accuracy, with a deviation of only 1.17 instances per image relative to the Ground Truth (GT), and achieves sub-micrometer accuracy in length estimation (deviation of $0.3\,\mu m$). For curvature, SAM3Text remains within the inter-annotator range, achieving a better average ($0.29\,\mu m^{-1}$ deviation) and only a marginally higher KL divergence.

However, all of the foundation models for microscopy — except StarDist — achieve reasonable downstream performance. In particular, CellSAM already performs strongly in its default configuration, ranking second in SKIoU. It also shows good agreement with the ground truth for length and curvature, but underperforms in counting accuracy. In contrast, only SAM3Text consistently attains top or near-top performance across all biological metrics without such compromises.

Qualitative analysis reinforces the strength of SAM3Text. We examine a particularly challenging real example with many intersecting MTs and low SNR in Fig. 8 to qualitatively compare methods, and find that these observations are in line with the quantitative results reported before.

Among all methods tuned with HPO, SAM3Text clearly stands out. It consistently captures all visible MTs (see bottom right in Fig. 8), including most of the complex intersections, producing continuous and well-separated instances with only very few false positive or false negative detections.

Its predecessors exhibit diverse failure modes: SAM generates numerous small, spurious detections despite postprocessing, while SAM2 merges intersecting filaments and occasionally segments enclosed areas, producing incorrect, non-elongated shapes.

The anchor-point-based methods FIESTA and TARDIS, which were explicitly designed for MT analysis, show mixed performance. Both methods can resolve intersections, but they miss a

substantial number of filaments. Moreover, when instances are detected, endpoints are often not identified correctly, leading to inaccurate length estimates.

Among the other foundation models specialized for microscopy, μSAM, CellSAM, and Cellpose-SAM recover many filament structures but exhibit notable weaknesses. μSAM and CellSAM frequently merge intersecting MTs, resulting in under-segmentation that hinders downstream measurements. CellSAM additionally produces overly thick segmentations. It is worth recalling that our SKIoU metric is robust to such width inaccuracies by design, as it operates on skeletonized versions of the predictions. Cellpose-SAM performs best within this group, correctly identifying a large fraction of instances, but still misses a large number of filaments, which limits its suitability for fully automated analysis pipelines that require high instance recall. Finally, StarDist fails to capture the elongated morphology of MTs as expected, since its star-convex object prior biases predictions toward circular shapes.

6 Discussion

Suitability of automated microtubule (MT) analysis. A central question in MT image analysis is whether fully automated, expert-level segmentation is currently achievable. Our results provide a nuanced, previously unknown answer. Most segmentation approaches still fall short of the accuracy required for reliable downstream biological measurements on interference reflection microscopy (IRM) data of *in vitro* reconstituted MTs. They frequently fragment filaments, miss faint or short instances completely, or introduce spurious detections, even when Hyperparameter Optimization (HPO) is used (few-shot setting).

SAM3 altered the picture. However, our study highlights a key positive outcome: guiding the recently introduced SAM3 model by a simple text prompt ("thin line") and tuning it on only 10 synthetic `SynthMT` images reaches performance within the human inter-annotator range on unseen, real data. This is, to our knowledge, the first demonstration that a general-purpose vision foundation model can be adapted to deliver fully automated, biologically meaningful MT segmentation.

Tuning with `SynthMT` enables human-grade performance. A key enabler of this achievement is the synthetic dataset `SynthMT`, which can be generated entirely from real MT images without the need for manual annotations. It automatically captures the visual statistics of real IRM frames and uses these to simulate filamentous structures. This enables the adaptation of segmentation methods to new microscopes, imaging conditions, or laboratories without any annotation effort. The success of SAM3Text after HPO on only 10 of those images demonstrates that method configuration can be effectively guided toward human-grade performance, without requiring any manual annotation of real data.

Within this context, `SynthMT` serves as both a benchmark and a diagnostic tool. It exposes the typical failure modes of existing methods, including fragmentation, missed instances, and sensitivity to domain shifts, and it provides a controlled setting for evaluating and tuning segmentation methods.

Summary. `SynthMT` provides a reproducible and annotation-free framework for evaluating segmentation methods, identifying their limitations, and enabling powerful foundation models like SAM3 to be configured for filamentous microscopy data. This establishes a practical and fully automated route toward scalable MT segmentation suitable for high-throughput biological experiments, such as (non-dynamic) end-point experiments.

Limitations and future work

One limitation is that our evaluation focuses exclusively on fully automated segmentation methods. Semi-automated approaches that require human input or corrections are not considered.

Moreover, our synthetic image generation relies on a parametric approach derived from real MT images. While it avoids the need for manual annotation, alternative approaches such as diffusion-based models have shown potential for generating realistic microscopy images [27], but they typically require large annotated datasets for training. Investigating these methods as a complementary or alternative generation strategy is an avenue for future work.

Furthermore, we did not use `SynthMT` as a large-scale training dataset for finetuning pretrained foundation models beyond the few-shot HPO experiments. `SynthMT` could potentially support fully supervised or semi-supervised training to further improve model performance.

Lastly, `SynthMT` currently focuses on 2D image generation from single static frames, which does not capture the temporal dynamics of MTs such as their alternating phases of growth and shrinkage. Extending the pipeline to videos would allow instance tracking over time, which is crucial for studying MT kinematics. Several approaches could be explored to achieve robust MT tracking. Time could be treated as a third dimension in volumetric models such as AnyStar [17] or Cellpose-SAM [12], or fully automated segmentation outputs could be combined with existing tracking algorithms [6, 60, 64]. Notably, SAM2 and SAM3 already support video data, which opens the possibility of extending their promptable, fully automated segmentation capabilities to temporal MT datasets.

Acknowledgments

Our work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) Project-ID 528483508 - FIP 12. We would like to thank Dominik Fachet and Gil Henkin from the Reber lab for providing data, and also thank the further study participants Moritz Becker, Nathaniel Boateng, and Miguel Aguilar. The Reber lab thanks staff at the Advanced Medical Bioimaging Core Facility (Charité, Berlin) for imaging support and the Max Planck Society for funding. Furthermore, we thank Kristian Hildebrand and Chaitanya A. Athale (IISER Pune, India) and his lab for helpful discussions.

Author contributions

MK and **JW**: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Project Administration, Resources, Software, Validation, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing.
DF: Data Curation, Resources, Writing – Original Draft Preparation.
SR, **FG** and **ER**: Conceptualization, Funding Acquisition, Supervision, Writing – Review & Editing.

References

1. Garcin C, Straube A. Microtubules in cell migration. Essays in biochemistry. 2019;63(5):509-20.

2. Prota AE, Lucena-Agell D, Ma Y, Estevez-Gallego J, Li S, Bargsten K, et al. Structural insight into the stabilization of microtubules by taxanes. Elife. 2023;12:e84791.

3. Kavallaris M, Tait AS, Walsh BJ, He L, Horwitz SB, Norris MD, et al. Multiple microtubule alterations are associated with Vinca alkaloid resistance in human leukemia cells. Cancer research. 2001;61(15):5803-9.

4. Soliman A, Bakota L, Brandt R. Microtubule-modulating agents in the fight against neurodegeneration: will it ever work? Current neuropharmacology. 2022;20(4):782-98.

5. Liu Y, Kolagunda A, Treible W, Nedo A, Caplan J, Kambhamettu C. Intersection to overpass: instance segmentation on filamentous structures with an orientation-aware neural network and terminus pairing algorithm. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops; 2019. p. 0-0.

6. Li Z, Jian Y, Hu J, Zhang C, Meng X, Liu J. SpindlesTracker: An automatic and low-cost labeled workflow for spindle analysis. IEEE Journal of Biomedical and Health Informatics. 2023;27(8):4098-109.

7. Kiewisz R, Fabig G, Conway W, Johnston J, Kostyuchenko VA, Bařinka C, et al. Accurate and fast segmentation of filaments and membranes in micrographs and tomograms with TARDIS. bioRxiv. 2024.

8. Ruhnow F, Zwicker D, Diez S. Tracking single particles and elongated filaments with nanometer precision. Biophysical journal. 2011;100(11):2820-8.

9. Carion N, Gustafson L, Hu YT, Debnath S, Hu R, Suris D, et al. Sam 3: Segment anything with concepts. arXiv preprint arXiv:251116719. 2025.

10. Israel U, Marks M, Dilip R, Li Q, Yu C, Laubscher E, et al. CellSAM: a foundation model for cell segmentation. BioRxiv. 2025.

11. Archit A, Freckmann L, Nair S, Khalid N, Hilt P, Rajashekar V, et al. Segment anything for microscopy. Nature Methods. 2025;22(3):579-91.

12. Pachitariu M, Rariden M, Stringer C. Cellpose-SAM: superhuman generalization for cellular segmentation. bioRxiv. 2025:2025-04.

13. Mazurowski MA, Dong H, Gu H, Yang J, Konz N, Zhang Y. Segment anything model for medical image analysis: an experimental study. Medical Image Analysis. 2023;89:102918.

14. Ma J, He Y, Li F, Han L, You C, Wang B. Segment anything in medical images. Nature Communications. 2024;15(1):654.

15. Huang Y, Yang X, Liu L, Zhou H, Chang A, Zhou X, et al. Segment anything model for medical images? Medical Image Analysis. 2024;92:103061.

16. Fan K, Liang L, Li H, Situ W, Zhao W, Li G. Research on Medical Image Segmentation Based on SAM and Its Future Prospects. Bioengineering. 2025;12(6):608.

17. Dey N, Abulnaga M, Billot B, Turk EA, Grant E, Dalca AV, et al. AnyStar: Domain randomized universal star-convex 3D instance segmentation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision; 2024. p. 7593-603.

18. Weigert M, Schmidt U, Haase R, Sugawara K, Myers G. Star-convex Polyhedra for 3D Object Detection and Segmentation in Microscopy. In: The IEEE Winter Conference on Applications of Computer Vision (WACV); 2020. doi:10.1109/WACV45572.2020.9093435.

19. Laydi AA, Cueff L, Crespo M, Mourabit YE, Bouvrais H. Adaptive Attention Residual U-Net for curvilinear structure segmentation in fluorescence microscopy and biomedical images. arXiv preprint arXiv:250707800. 2025.

20. Lugo Carlos A NF Saikia Eashan. A Typical Workflow to Simulate Cytoskeletal Systems. Journal of Visualized Experiments (JoVE). 2023;(194):e64125. doi:doi:10.3791/64125.

21. Dmitrieff S, Nédélec F. ConfocalGN: a minimalistic confocal image simulator. bioRxiv. 2016:088906.

22. Saguy A, Nahimov T, Lehrman M, Gómez-de Mariscal E, Hidalgo-Cenalmor I, Alalouf O, et al. This Microtubule Does Not Exist: Super-Resolution Microscopy Image Generation by a Diffusion Model. Small Methods. 2025;9(3):2400672.

23. Weigert M, Schmidt U, Boothe T, Müller A, Dibrov A, Jain A, et al. Content-aware image restoration: pushing the limits of fluorescence microscopy. Nature methods. 2018;15(12):1090-7.

24. Park T, Liu MY, Wang TC, Zhu JY. Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2019. p. 2337-46.

25. Fernandez V, Pinaya WHL, Borges P, Tudosiu PD, Graham MS, Vercauteren T, et al. Can segmentation models be trained with fully synthetically generated data? In: International workshop on simulation and synthesis in medical imaging. Springer; 2022. p. 79-90.

26. Zhang L, Rao A, Agrawala M. Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision; 2023. p. 3836-47.

27. Konz N, Chen Y, Dong H, Mazurowski MA. Anatomically-controllable medical image generation with segmentation-guided diffusion models. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2024. p. 88-98.

28. Liu Y, Peng S, Caplan J, Kambhamettu C. Pick and Trace: Instance Segmentation for Filamentous Objects with a Recurrent Neural Network. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2023. p. 635-45.

29. Xu T, Vavylonis D, Tsai FC, Koenderink GH, Nie W, Yusuf E, et al. SOAX: a software for quantification of 3D biopolymer networks. Scientific reports. 2015;5(1):9081.

30. Zhang Z, Nishimura Y, Kanchanawong P. Extracting microtubule networks from superresolution single-molecule localization microscopy data. Molecular biology of the cell. 2017;28(2):333-45.

31. He K, Gkioxari G, Dollár P, Girshick R. Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision; 2017. p. 2961-9.

32. Kirillov A, Mintun E, Ravi N, Mao H, Rolland C, Gustafson L, et al. Segment anything. In: Proceedings of the IEEE/CVF international conference on computer vision; 2023. p. 4015-26.

33. Ravi N, Gabeur V, Hu YT, Hu R, Ryali C, Ma T, et al. Sam 2: Segment anything in images and videos. arXiv preprint arXiv:240800714. 2024.

34. Stringer C, Wang T, Michaelos M, Pachitariu M. Cellpose: a generalist algorithm for cellular segmentation. Nature methods. 2021;18(1):100-6.

35. Schmidt U, Weigert M, Broaddus C, Myers G. Cell Detection with Star-Convex Polygons. In: Medical Image Computing and Computer Assisted Intervention - MICCAI 2018 - 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II; 2018. p. 265-73. doi:10.1007/978-3-030-00934-2_30.

36. Greenwald NF, Miller G, Moen E, Kong A, Kagel A, Dougherty T, et al. Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. Nature biotechnology. 2022;40(4):555-65.

37. Edlund C, Jackson TR, Khalid N, Bevan N, Dale T, Dengel A, et al. LIVECell—A large-scale dataset for label-free live cell segmentation. Nature methods. 2021;18(9):1038-45.

38. Cutler KJ, Stringer C, Lo TW, Rappez L, Stroustrup N, Brook Peterson S, et al. Omnipose: a high-precision morphology-independent solution for bacterial cell segmentation. Nature methods. 2022;19(11):1438-48.

39. Spahn C, de Mariscal EG, Laine R, Pereira P, von Chamier L, Conduit M, et al. DeepBacs for multi-task bacterial image analysis using open-source deep learning approaches. Communications Biology. 2022 Dec;5(1). doi:10.1038/s42003-022-03634-z.

40. Kumar N, Verma R, Anand D, Zhou Y, Onder OF, Tsougenis E, et al. A multi-organ nucleus segmentation challenge. IEEE transactions on medical imaging. 2019;39(5):1380-91.

41. Nunley H, Shao B, Denberg D, Grover P, Singh J, Avdeeva M, et al. Nuclear instance segmentation and tracking for preimplantation mouse embryos. Development. 2024;151(21):dev202817.

42. Wolny A, Cerrone L, Vijayan A, Tofanelli R, Barro AV, Louveaux M, et al. Accurate and versatile 3D segmentation of plant tissues at cellular resolution. Elife. 2020;9:e57613.

43. Verma R, Kumar N, Patil A, Kurian NC, Rane S, Graham S, et al. MoNuSAC2020: A multi-organ nuclei segmentation and classification challenge. IEEE Transactions on Medical Imaging. 2021;40(12):3413-23.

44. Ma J, Xie R, Ayyadhury S, Ge C, Gupta A, Gupta R, et al. The multimodality cell segmentation challenge: toward universal solutions. Nature methods. 2024;21(6):1103-13.

45. Pape C, Remme R, Wolny A, Olberg S, Wolf S, Cerrone L, et al. Microscopy-based assay for semi-quantitative detection of SARS-CoV-2 specific antibodies in human sera: A semi-quantitative, high throughput, microscopy-based assay expands existing approaches to measure SARS-CoV-2 specific antibody levels in human sera. Bioessays. 2021;43(3):2000257.

46. Graham S, Jahanifar M, Azam A, Nimir M, Tsang YW, Dodd K, et al. Lizard: a large-scale dataset for colonic nuclear instance segmentation and classification. In: Proceedings of the IEEE/CVF international conference on computer vision; 2021. p. 684-93.

47. Bondarenko V, Nikolaev M, Kromm D, Belousov R, Wolny A, Blotenburg M, et al. Embryo-uterine interaction coordinates mouse embryogenesis during implantation. The EMBO Journal. 2023;42(17):e113280.

48. Wei D, Lin Z, Franco-Barranco D, Wendt N, Liu X, Yin W, et al. MitoEM dataset: large-scale 3D mitochondria instance segmentation from EM images. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2020. p. 66-76.

49. Vergara HM, Pape C, Meechan KI, Zinchenko V, Genoud C, Wanner AA, et al. Whole-body integration of gene expression and single-cell morphology. Cell. 2021;184(18):4819-37.

50. Lucchi A, Li Y, Smith K, Fua P. Structured image segmentation using kernelized features. In: European conference on computer vision. Springer; 2012. p. 400-13.

51. Conrad R, Narayan K. Instance segmentation of mitochondria in electron microscopy images with a generalist deep learning model trained on a diverse dataset. Cell Systems. 2023;14(1):58-71.

52. Mekuč MŽ, Bohak C, Boneš E, Hudoklin S, Marolt M, et al. Automatic segmentation and reconstruction of intracellular compartments in volumetric electron microscopy data. Computer methods and programs in biomedicine. 2022;223:106959.

53. Lin Z, Wei D, Petkova MD, Wu Y, Ahmed Z, K KS, et al. Nucmm dataset: 3d neuronal nuclei instance segmentation at sub-cubic millimeter scale. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2021. p. 164-74.

54. Gallusser B, Maltese G, Di Caprio G, Vadakkan TJ, Sanyal A, Somerville E, et al. Deep neural network automated segmentation of cellular structures in volume electron microscopy. Journal of Cell Biology. 2022;222(2):e202208005.

55. Schwartz MS, Moen E, Miller G, Dougherty T, Borba E, Ding R, et al. Caliban: Accurate cell tracking and lineage construction in live-cell imaging experiments with deep learning. bioRxiv. 2024. Available from: `https://www.biorxiv.org/content/early/2024/08/07/803205`. arXiv:https://www.biorxiv.org/content/early/2024/08/07/803205.full.pdf. doi:10.1101/803205.

56. Caicedo JC, Goodman A, Karhohs KW, Cimini BA, Ackerman J, Haghighi M, et al. Nucleus segmentation across imaging experiments: the 2018 Data Science Bowl. Nature methods. 2019;16(12):1247-53.

57. Mathieu G, Bachir ED, et al. Brifiseg: a deep learning-based method for semantic and instance segmentation of nuclei in brightfield images. arXiv preprint arXiv:221103072. 2022.

58. Dietler N, Minder M, Gligorovski V, Economou AM, Lucien Joly DAH, Sadeghi A, et al. YeaZ: A convolutional neural network for highly accurate, label-free segmentation of yeast microscopy images. bioRxiv. 2020:2020-05.

59. Kim H, Shin J, Kim E, Kim H, Hwang S, Shim JE, et al. YeastNet v3: a public database of data-specific and integrated functional gene networks for Saccharomyces cerevisiae. Nucleic acids research. 2014;42(D1):D731-6.

60. Kapoor V, Hirst WG, Hentschel C, Preibisch S, Reber S. MTrack: automated detection, tracking, and analysis of dynamic microtubules. Scientific reports. 2019;9(1):3794.

61. Liu Y, Treible W, Kolagunda A, Nedo A, Saponaro P, Caplan J, et al. Densely connected stacked u-network for filament segmentation in microscopy images. In: Proceedings of the European Conference on Computer Vision (ECCV) Workshops; 2018. p. 0-0.

62. Liu Y, Caplan J, Kambhamettu C. Extraction and Quantification of Actin Cytoskeleton in Microscopic Images Using a Deep Learning Based Framework and a Curve Clustering Model. In: 2022 26th International Conference on Pattern Recognition (ICPR). IEEE; 2022. p. 4270-6.

63. Masoudi S, Razi A, Wright CH, Gatlin JC, Bagci U. Instance-level microtubule tracking. IEEE transactions on medical imaging. 2020;39(6):2061-75.

64. Khatri D, Yadav SA, Athale CA. KnotResolver: tracking self-intersecting filaments in microscopy using directed graphs. Bioinformatics. 2024;40(9):btae538.

65. Stalling D, Westerhoff M, Hege HC, et al. Amira: A highly interactive system for visual data analysis. The visualization handbook. 2005;38:749-67.

66. Oquab M, Darcet T, Moutakanni T, Vo H, Szafraniec M, Khalidov V, et al. Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:230407193. 2023.

67. Studio L;. `https://labelstud.io`.

68. Bergstra J, Bardenet R, Bengio Y, Kégl B. Algorithms for hyper-parameter optimization. Advances in neural information processing systems. 2011;24.

69. Li S, Bai Y, Zhang T, Wang H. Evaluating Hyper-Parameter Optimization On Remote Sensing Object Detection. In: 2022 2nd International Conference on Algorithms, High Performance Computing and Artificial Intelligence (AHPCAI). IEEE; 2022. p. 1-7.

70. Liang M, An B, Li K, Du L, Deng T, Cao S, et al. Improving genomic prediction with machine learning incorporating TPE for hyperparameters optimization. Biology. 2022;11(11):1647.

71. Bergstra J, Yamins D, Cox D. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In: International conference on machine learning. PMLR; 2013. p. 115-23.

72. Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: A Next-Generation Hyperparameter Optimization Framework. In: The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining; 2019. p. 2623-31.

73. Stein G, Cresswell J, Hosseinzadeh R, Sui Y, Ross B, Villecroze V, et al. Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. Advances in Neural Information Processing Systems. 2023;36:3732-84.

74. Bolya D, Huang PY, Sun P, Cho JH, Madotto A, Wei C, et al. Perception encoder: The best visual embeddings are not at the output of the network. arXiv preprint arXiv:250413181. 2025.

75. Feng Y, Chai X, Ba Q, Yang G. Quality assessment of synthetic fluorescence microscopy images for image segmentation. In: 2019 IEEE International Conference on Image Processing (ICIP). IEEE; 2019. p. 814-8.

76. Otani M, Togashi R, Sawai Y, Ishigami R, Nakashima Y, Rahtu E, et al. Toward verifiable and reproducible human evaluation for text-to-image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2023. p. 14277-86.

77. Nguyen TV, Hoster J, Glaser A, Hildebrand K, Biessmann F. Generating Synthetic Satellite Imagery for Rare Objects: An Empirical Comparison of Models and Metrics. arXiv preprint arXiv:240901138. 2024.

78. Koh JY, Baldridge J, Lee H, Yang Y. Text-to-image generation grounded by fine-grained user attention. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision; 2021. p. 237-46.

79. Ramesh A, Pavlov M, Goh G, Gray S, Voss C, Radford A, et al. Zero-shot text-to-image generation. In: International conference on machine learning. Pmlr; 2021. p. 8821-31.

80. Saharia C, Chan W, Saxena S, Li L, Whang J, Denton EL, et al. Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems. 2022;35:36479-94.

81. Yates M, Hart G, Houghton R, Torres MT, Pound M. Evaluation of synthetic aerial imagery using unconditional generative adversarial networks. ISPRS Journal of Photogrammetry and Remote Sensing. 2022;190:231-51.

82. Kullback S, Leibler RA. On information and sufficiency. The annals of mathematical statistics. 1951;22(1):79-86.

83. Bieling P, Telley IA, Hentrich C, Piehler J, Surrey T. Fluorescence microscopy assays on chemically functionalized surfaces for quantitative imaging of microtubule, motor, and+ TIP dynamics. Methods in cell biology. 2010;95:555-80.

84. Hirst WG, Kiefer C, Abdosamadi MK, Schäffer E, Reber S. In vitro reconstitution and imaging of microtubule dynamics by fluorescence and label-free microscopy. STAR protocols. 2020;1(3):100177.

85. Hirst WG, Fachet D, Kuropka B, Weise C, Saliba KJ, Reber S. Purification of functional Plasmodium falciparum tubulin allows for the identification of parasite-specific microtubule inhibitors. Current Biology. 2022;32(4):919-26.

86. Hutter F, Hoos H, Leyton-Brown K. An efficient approach for assessing hyperparameter importance. In: International conference on machine learning. PMLR; 2014. p. 754-62.

A Appendix

A.1 Details about evaluated method

We summarize implementation and parameter details for all evaluated methods (ordered chronologically in terms of publication date): FIESTA [8], StarDist [35], TARDIS [7], SAM [32], SAM2 [33], μSAM [11], CellSAM [10], Cellpose-SAM [12], SAM3 [9] (with Automatic Instance Segmentation (AIS)) and SAM3Text (SAM3 guided by a text prompt). StarDist, μSAM, CellSAM, and Cellpose-SAM produce 512×512 integer arrays, where 0 denotes background and each positive integer corresponds to a unique instance. As each pixel can belong to only exactly one instance or background, overlaps cannot be represented. On the other hand, SAM, SAM2, SAM3 and SAM3Text return lists of 512×512 boolean masks, one per instance. This representation allows for overlapping instances, since multiple masks can contain the same pixel as foreground (instance). FIESTA and TARDIS do not output segmentation masks as arrays. Instead, they return lists of anchor points $(p_k)_k = (x_k, y_k)_k$, one for each instance. For more information about the Hyperparameter Optimization (HPO) that yields the below stated "tuned" hyperparameter values, we refer to appendix A.7.

The full code, including fixed versions for full reproducibility, is available at `github.com/ml-lab-htw/SynthMT`.

FIESTA. Fluorescence Image Evaluation Software for Tracking and Analysis (FIESTA [8]) is a classical, non–machine-learning software tool for tracking fluorescently labeled filaments in 2D or 3D time-lapse microscopy data. We use FIESTA as a baseline, because it represents the traditional, widely adopted approach for microtubule (MT) analysis.

The tracking algorithm evaluates every image in a sequence independently before linking detected objects into trajectories. This involves several steps: thresholding, feature detection, image segmentation, a fitting process using Gaussian models to achieve sub-pixel localization, and interpolation. Finally, a graph-theoretic approach is used to link the detected objects into trajectories. The primary input is a time-series of images, and the output consists of data files with filament coordinates (anchor points).

In this work, we use FIESTA via a MATLAB implementation[4] invoked from Python using the MATLAB Engine API for Python[5]. We utilize only the single-frame detection capabilities of FIESTA to identify filaments, without using its temporal tracking functionality. The `fwhm_estimate` parameter has no default value, so we fixed an estimate of 3.0 based on our data. Furthermore, we made several adaptations to the original FIESTA source code to ensure smooth execution, which we attribute to MATLAB version incompatibilities or our use of FIESTA in a scripted, non-GUI environment. The default and tuned hyperparameters are summarized in Table A.1.

Table A.1. Optimized FIESTA hyperparameters obtained through HPO.

Parameter	Default	Tuned
`background_filter`	`False`	`False`
`binary_image_processing`	`none`	`average`
`dynamicfil`	`False`	`False`
`focus_correction`	`False`	`False`
`fwhm_estimate`	(3.0)	4.13
`height_threshold`	2.0	1.37
`min_cod`	0.5	0.09
`reduce_fit_box`	1.0	2.61

[4] `https://github.com/fiesta-tud/FIESTA`
[5] `https://pypi.org/project/matlabengine`

StarDist. StarDist is a deep learning-based object detection and segmentation method, particularly effective for identifying cell nuclei in microscopy images in densely packed scenarios. Instead of traditional bounding boxes, it represents objects as star-convex polygons, which are well-suited for the roundish shape of nuclei. The core of StarDist is a convolutional neural network (CNN) based on the U-Net architecture, that, for every pixel, predicts an object probability and the radial distances to the object's boundary representing a star-convex polygon. The latter are finally refined using non-maximum suppression (NMS).

StarDist is used via `model = stardist.models.StarDist2D.from_pretrained(pretrained)`[6], where `pretrained` can be one of `2D_versatile_fluo`, `2D_versatile_he`, or `2D_paper_dsb2018`. Predictions on an `image` are called as

```
model.predict_instances(image, axes="YX",
    prob_thresh=prob_thresh, nms_thresh=nms_thresh)
```

The default and tuned hyperparameters are summarized in Table A.2.

Table A.2. Default StarDist hyperparameters and those obtained through HPO.

Parameter	Default	Tuned
`pretrained`	`2D_versatile_fluo`	`2D_paper_dsb2018`
`prob_thresh`	0.479	0.61
`nms_thresh`	0.3	0.03

TARDIS. Transformer and Rapid Dimensionless Instance Segmentation (TARDIS) is a fully automated segmentation workflow designed for cytoskeletal filaments and organelles. The TARDIS workflow consists of three main steps: semantic segmentation which produces a semantic mask, post-processing of the semantic mask into a point cloud representation of the objects, and instance segmentation of the point cloud. It is used via `tardis_em`[7]. Predictions on an `image` are generated by first saving it as a grayscale 2D TIFF in a temporary working directory `work_dir` and then calling

```
GeneralPredictor(predict="Microtubule_tirf", dir_s=work_dir,
    binary_mask=False, correct_px=None, normalize_px=1.0,
    convolution_nn="fnet_attn", checkpoint=[None, None],
    model_version=None, output_format="tif_csv", patch_size=128,
    cnn_threshold=str(cnn_threshold),
    dist_threshold=dist_threshold,
    points_in_patch=900, predict_with_rotation=True,
    filter_by_length=None, connect_splines=0,
    connect_cylinder=None, instances=True, continue_b=False)()
```

The default and tuned hyperparameters are summarized in Table A.3. Note that the predictions are saved as CSV files in `work_dir` from which the anchor points can be read.

Table A.3. Default TARDIS hyperparameters and those obtained through HPO.

Parameter	Default	Tuned
`cnn_threshold`	0.1	0.18
`dist_threshold`	0.5	0.69

SAM. Segment Anything Model (SAM) is a foundation model for image segmentation. Trained on over a billion masks, SAM enables zero-shot transfer to new image distributions and

[6] https://pypi.org/project/stardist
[7] https://pypi.org/project/tardis-em/

segmentation tasks. We use SAM in Automatic Instance Segmentation (AIS) mode, which generates masks without manual prompts by sampling points across the image.

Its architecture features an image encoder, a prompt encoder for point or bounding box inputs, and a mask decoder. While the original paper also mentions text prompts, this feature was never made publicly available. We use SAM via the Hugging Face `transformers.pipeline`[8] wrapper. Instantiation happens via `model=pipeline("mask-generation", model="facebook/sam-vit-huge")` and prediction on an `image` are called via

```
img = (image-image.min())/(image.max()-image.min())*255
model(Image.fromarray(img.astype(np.uint8)),
    points_per_batch=64, pred_iou_thresh=pred_iou_thresh,
    stability_score_thresh=stability_score_thresh,
    min_mask_region_area=0)
```

The default and tuned hyperparameters are summarized in Table A.4.

Table A.4. Default SAM hyperparameters and those obtained through HPO.

Parameter	Default	Tuned
`pred_iou_thresh`	0.88	0.56
`stability_score_thresh`	0.95	0.93

SAM2. Segment Anything Model 2 (SAM2) is a unified model for both video and image segmentation, extending the capabilities of its predecessor to the temporal domain. It introduces the Promptable Visual Segmentation (PVS) task, which generalizes segmentation to videos using prompts like points, boxes, or masks on any frame. SAM2 employs a streaming architecture with a memory module to track objects across frames, effectively handling appearance changes, occlusions, and motion. For single images, it functions similar to SAM, but it was trained on a newer Segment Anything Video (SA-V) dataset. Usage is identical to SAM above; the only differences are the model used (`facebook/sam2.1-hiera-large`) and the default and tuned hyperparameters (see Table A.5).

Table A.5. Default SAM2 hyperparameters and those obtained through HPO.

Parameter	Default	Tuned
`pred_iou_thresh`	0.8	0.26
`stability_score_thresh`	0.95	0.70

μSAM. μSAM improves and extends SAM for microscopy data. It addresses the challenge of identifying objects across various microscopy modalities like light microscopy (LM) and electron microscopy (EM) by fine-tuning SAM with a new decoder, resulting in improved instance segmentation. This approach provides specialized models for LM and EM that outperform the default SAM and includes an interactive tool for annotation. μSAM is the only package[9] that needs to be installed via *conda* and is then used via

```
predictor, segmenter =
    micro_sam.automatic_segmentation.get_predictor_and_segmenter(
        model_type="vit_l_lm", checkpoint=None,
        amg=False, is_tiled=False)
```

Prediction on an `image` are called by first computing embeddings,

[8] `https://pypi.org/project/transformers`
[9] `https://github.com/computational-cell-analytics/micro-sam`

```
embeddings = micro_sam.util.precompute_image_embeddings(
     predictor=predictor, input_=image, ndim=2, batch_size=1)
```

initializing the segmenter with `segmenter.initialize(img, embeddings)`, and producing the masks using

```
segmenter.generate(
    min_size=0,
    center_distance_threshold=center_distance_threshold,
    boundary_distance_threshold=boundary_distance_threshold,
    foreground_threshold=foreground_threshold,
    foreground_smoothing=foreground_smoothing,
    distance_smoothing=distance_smoothing, output_mode=None,
    n_threads=1)
```

The default and tuned hyperparameters are summarized in Table A.6. Model-internal postprocessing is turned off by setting `min_size=0` above (see section 4).

Table A.6. Default μSAM hyperparameters and those obtained through HPO.

Parameter	Default	Tuned
`center_distance_threshold`	0.5	0.89
`boundary_distance_threshold`	0.5	0.52
`foreground_threshold`	0.5	1.95
`foreground_smoothing`	1.0	1.11
`distance_smoothing`	1.6	0.73

CellSAM. CellSAM is a foundation model for cell segmentation that extends SAM to perform automated cellular instance segmentation. To overcome SAM's limitations with dense cellular images, CellSAM employs a prompt engineering approach, using a transformer-based object detector called CellFinder to generate bounding box prompts. CellFinder and the segmentation model share the same Vision Transformer (ViT) backbone. Trained on a large, diverse corpus of cellular imaging data, CellSAM achieves state-of-the-art performance on numerous segmentation datasets. It is used via `model = cellSAM.get_model(version="1.2")`[10] (requires setting a DeepCell access token under `DEEPCELL_ACCESS_TOKEN`). Prediction on an `image` are called via

```
cellSAM.segment_cellular_image(
    cellSAM.cellsam_pipeline.normalize_image(image)
    model=model, normalize=False, postprocess=False,
    remove_boundaries=False, bounding_boxes=None,
    bbox_threshold=self.bbox_threshold, fast=False)
```

The default and tuned hyperparameters are summarized in Table A.7. We turned off all model-internal pre- and postprocessing steps by setting `normalize=False, postprocess=False, remove_boundaries=False` above. However, note that we do use the `cellSAM.cellsam_pipeline.normalize_image` as its a main part of the architecture (similar to how with SAM, we need to rescale to `uint8` (see above)).

Cellpose-SAM. Cellpose-SAM integrates the pretrained transformer backbone of SAM into the Cellpose framework. This combination aims to achieve better generalization and performance, outperforming inter-human agreement and approaching a hypothetical "human-consensus" bound for segmentation quality. By leveraging the Cellpose framework's effective loss function and post-processing with SAM's strong inductive biases, Cellpose-SAM establishes itself as a foundation

[10] `https://github.com/mario-koddenbrock/cellSAM`

Table A.7. Default CellSAM hyperparameter and that obtained through HPO.

Parameter	Default	Tuned
`cellprob_threshold`	0	0.37
`diameter`	30	37
`flow_threshold`	0.4	0.83

model for biological segmentation. It is used via `model = cellpose.models.CellposeModel(pretrained_model="cpsam")`[11]. Prediction on an `image` are called via

```
model.eval(x=image, anisotropy=None, augment=False, batch_size=1,
    bsize=256, cellprob_threshold=cellprob_threshold,
    channel_axis=None, compute_masks=True,
    diameter=self.diameter, do_3D=False,
    flow_threshold=self.flow_threshold,
    invert=False, max_size_fraction=1,
    min_size=0, niter=None, normalize=False,
    resample=True, rescale=None, stitch_threshold=0.0,
    tile_overlap=0.1, z_axis=None)
```

The default and tuned hyperparameters are summarized in Table A.8. We turned off all model-internal pre- and postprocessing steps by setting `invert=False, max_size_fraction=1, min_size=0, normalize=False` above.

Table A.8. Default Cellpose-SAM hyperparameters and those obtained through HPO.

Parameter	Default	Tuned
`cellprob_threshold`	0	0.37
`diameter`	30	37
`flow_threshold`	0.4	0.83

SAM3. Segment Anything Model 3 (SAM3) is a unified model that detects, segments, and tracks objects in images and videos based on "concept prompts" such as short noun phrases or image exemplars. This task, termed Promptable Concept Segmentation (PCS), returns segmentation masks and unique identities for all matching object instances. The model architecture features an image-level detector and a memory-based video tracker sharing a single backbone. SAM3 is the first in the series to offer fully functional text-prompted segmentation. Identical in usage to SAM and SAM2 above; only difference is the model used (`facebook/sam3`) and the default and tuned hyperparameters (see Table A.9). Note that access to SAM3 on Hugging Face[12] has to be granted.

Table A.9. Default SAM3 hyperparameters and those obtained through HPO.

Parameter	Default	Tuned
`pred_iou_thresh`	0.88	0.72
`stability_score_thresh`	0.95	0.74

SAM3Text. SAM3Text uses SAM3's text prompting capabilities instead of AIS. For that, we come up with ten different text prompts which the HPO can choose from. As a plausible default, we set `thin line`. The other options are `elongated structure`, `straight black line`,

[11] `https://pypi.org/project/cellpose`
[12] `https://huggingface.co/facebook/sam3`

`linear biological filament`, `narrow elongated line`, `thin red-highlighted line`, `thin bright-on-dark microstructure`, `thin structure in noisy background`, `small linear object`, and `thin linear structure among noise`. The model is used via the Hugging Face `transformers` module,

```
model = transformers.Sam3Model.from_pretrained("facebook/sam3")
processor = transformers.Sam3Processor.from_pretrained("facebook/sam3")
```

Predictions on an `image` are called via

```
img = (image-image.min())/(image.max()-image.min())*255
inputs = processor(
    Image.fromarray(img.astype(np.uint8)),
    text=text_prompt, return_tensors="pt")
outputs = model(**inputs)
```

and results are given by

```
processor.post_process_instance_segmentation(
    outputs, threshold=threshold, mask_threshold=mask_threshold,
    target_sizes=inputs.get("original_sizes").tolist())[0]
```

The default and tuned hyperparameters are summarized in Table A.10.

Table A.10. Default SAM3Text hyperparameters and those obtained through HPO.

Parameter	Default	Tuned
`threshold`	0.5	0.40
`mask_threshold`	0.5	0.44
`text_prompt`	`thin line`	`thin line`

A.2 Common preprocessing

All method wrappers inherit from a common base class that centralizes the preprocessing parameters `grayscale`, `sharpen_radius`, `smooth_radius`, `percentile_min`, `percentile_max`, `clip_to_percentiles`, `rescale_using_percentiles`, `invert`, and `histogram_normalization`. We chose the default parameters for each method such that our preprocessing pipeline reflects their internal one that we turned off (see selection of method-specific default parameters in appendix A.1). The idea behind this is that in the HPO, all methods have the same amount of preprocessing capabilities available. In Table A.11, we list both the default and tuned preprocessing parameters for each method. For more information, we refer to our implementation of `preprocessing.process_image()` in our codebase.

A.3 Microtubule in vitro reconstitution and microscopy

MTs were reconstituted and visualized using interference reflection microscopy (IRM) as previously described. In short, reaction chambers were constructed on PLL-PEG passivated glass slides and coverglassses functionalized with Hydroxy-ω-Amino polyethylene glycol (PEG) and α-Biotinamido-ω-Amino PEG [83]. The chambers were further functionalized and passivated by subsequent washes with 1% (w/v) Pluronic F-127, 1 mg/ml kappa-casein and 0.2 mg/ml Neutravidin and MTs seeds, containing fluorescently labeled, biotinylated tubulin, stabilised by GMPCPP. Dynamic polymerization reactions varied depending on experimental conditions but were carried out with 6–20 μM tubulin in the presence of 1 mg/ml k-casein, 0.1%–1% (v/v) β-mercaptoethanol, 2.5 mM Protocatechuic Acid, 25 nM Protocatechuate-3,4-dioxygenase, 0.15% (w/v) methylcellulose in a BRB80-based buffer [84]. Functional tubulin was derived from various

Table A.11. Default (Def.) and tuned preprocessing parameters obtained through HPO. Abbreviations: `gryscl` = `grayscale`, `shrpn_rad` = `sharpen_radius`, `smth_rad` = `smooth_radius`, `perc_min` = `percentile_min`, `perc_max` = `percentile_max`, `clip_perc` = `clip_to_percentiles`, `rescl_perc` = `rescale_using_percentiles`, `hist_norm` = `histogram_normalization`, `f` = `False`, `t` = `True`.

	gryscl		shrpn_rad		smth_rad		perc_min		perc_max		clip_perc		rescl_perc		invert		hist_norm	
Model	Def.	Tuned	Def.	Tuned	Def.	Tuned	Def.	Tuned	Def.	Tuned	Def.	Tuned	Def.	Tuned	Def.	Tuned	Def.	Tuned
FIESTA	t	t	0	0.18	0	2.25	0	4.25	100	96.11	f	f	f	f	f	f	f	f
StarDist	t	t	0	8.12	0	18.10	0	0.18	100	96.71	f	f	f	t	f	t	f	t
TARDIS	t	t	0	7.17	0	0.36	0	2.89	100	99.99	f	f	f	f	f	f	f	f
μSAM	f	f	0	3.07	0	1.80	0	0.63	100	96.27	f	t	f	f	f	t	f	f
CellSAM	f	t	0	1.15	0	5.11	0	2.66	99	95.33	t	f	f	f	f	t	t	f
Cellpose-SAM	f	t	0	1.45	0	19.79	1	2.01	99	99.42	f	f	t	t	f	t	f	f
SAM	f	t	0	1.02	0	17.62	0	4.53	100	98.60	f	t	f	f	f	t	f	f
SAM2	f	f	0	1.48	0	0.50	0	2.58	100	97.90	f	f	f	t	f	f	f	f
SAM3	f	f	0	0.35	0	4.28	0	2.59	100	97.03	f	f	f	f	f	t	f	t
SAMText	f	t	0	0.39	0	3.41	0	3.51	100	99.46	f	t	f	t	f	f	f	f

sources, ranging from commercially available mammalian tubulin to protozoan tubulin as described in [85]. The samples were imaged using either a Nikon Ti2 inverted widefield microscope or a Nikon Eclipse Ti-E microscope. The Ti2 system was equipped with a 50/50 beam splitter, a Nikon Plan Apochromat 60×/0.95 NA oil objective, Lumencore SpectraX LED illumination, and a pco.edge 4.2 LT HQ camera. The Ti-E system featured a 50/50 beam splitter, a Nikon Plan Apochromat 100×/1.49 NA oil immersion objective, and a Photometrics Prime 95B sCMOS camera. MTs were visualized using IRM and the seeds using a excitation with a 635 nm LED or 647 nm laser, respectively. The resulting images have a resolution of 9.0909 to 9.2308 pixels per micrometer.

A.4 Example images from `SynthMT` and other available datasets

Fig. A.4 depicts examples from our `SynthMT` dataset (see section 4 for details about the process of creation it). Even though there is no other publicly available dataset for our task (single *in vitro* microtubules (MTs) nucleated from fixed seeds), we show samples of other datasets that are close to it. However, as noted in section 2, none of these papers validate the fidelity of their datasets and do not evaluate them on foundation models (that is, treat it as a benchmark). In Fig. A.5a, we show samples from MicSim_FluoMT [19], and Fig. A.5b depicts samples from DRIFT [28].

A.5 User interface of the study

The user interface that was used to assess the perceptual realism of `SynthMT` is visualized in Fig. A.6, and some sample images that appear in it are shown in Fig. A.7. We also make this code publicly available at `https://github.com/DATEXIS/SynthMT-study`.

A.6 Length and curvature distributions

Besides the Kullback–Leibler divergence (KL divergence), which we report throughout the paper (e.g., in Tables 1 and 2), we visualize the actual predicted and ground-truth distributions of `SynthMT` for all methods in Figs. A.8 and A.9.

A.7 Model HPO and parameter importance

For completeness, we report the progression of the Hyperparameter Optimization (HPO) (as outlined in section 4) for all evaluated methods in Fig. A.10. Recall that we use 10 random images from `SynthMT` and Skeleton Intersection over Union (SKIoU) as a metric to be optimized in each run. Each subplot corresponds to a distinct method and shows the best observed value for SKIoU as a function of the trial index (1000 in total). The underlying search was performed over each

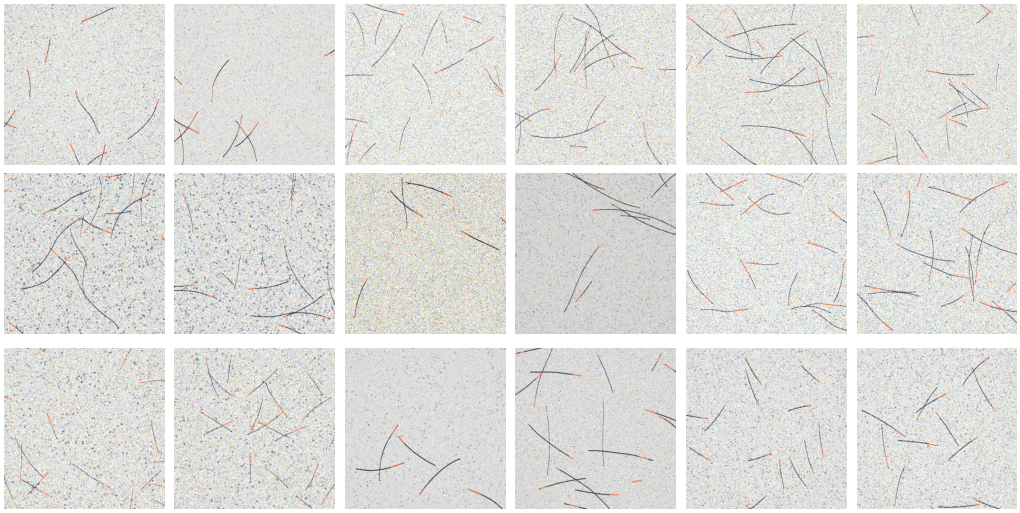
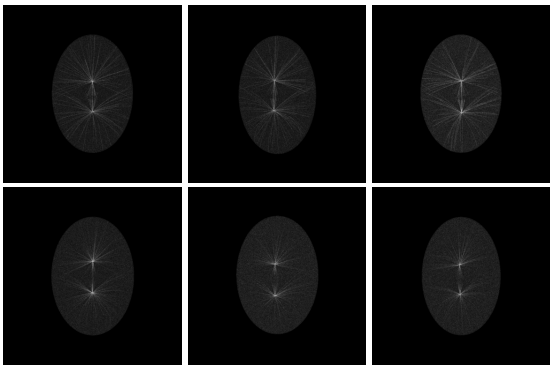


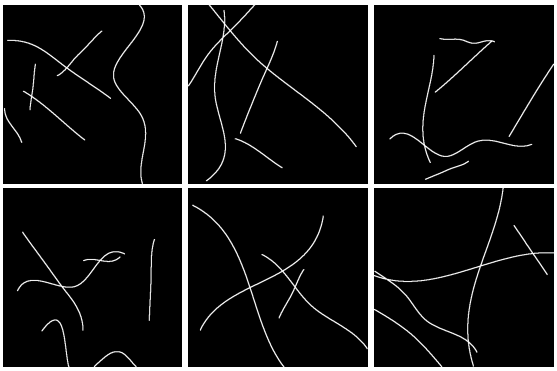
Figure A.4. Examples from the `SynthMT` dataset. 18 images sampled from the synthetic `SynthMT` dataset. Each image depicts individual MTs growing from stabilized seeds (shown in red) under simulated interference reflection microscopy (IRM) conditions. The dataset captures natural variation in MT quantity, length, and curvature across different experimental conditions represented by 660 optimized parameter sets. The generative pipeline (Fig. 2) produces realistic imaging characteristics including background texture, vignetting effects, blur, and multiple noise sources (Poisson, Gaussian, and spatially correlated noise), and is guided by real images as depicted in Fig. 3. Every image is accompanied by pixel-accurate segmentation masks for each MTs, providing ground-truth annotations for quantitative benchmarking of segmentation methods.

method's specific (see appendix A.1) and the preprocessing hyperparameters (see appendix A.2), and their search areas are detailed in our codebase. It can be inferred that the HPO is effective for all methods, where most of them exhibit a rapid increase already within the first tens of trials. Occasional jumps in the curves correspond to the discovery of qualitatively better hyperparameter settings, for example combinations that improve image contrast handling or regularization for faint microtubules. We use the highest validation SKIoU reached by each method as the "tuned" HPO parameters (as reported in appendices A.1 and A.2).

The importance of each of the parameters can be inferred from the bar plots in Fig. A.11 and is calculated using f-ANOVA [86]. Except for StarDist and SAM3Text, preprocessing parameters are always most important. The parameter importance analysis further reveals that for several methods, a small subset of hyperparameters accounts for most of the performance variance. For instance, while SAM and μSAM have multiple hyperparameters, their performance is predominantly influenced by a single key parameter. Similarly, the performance of CellSAM and Cellpose-SAM is largely determined by two critical parameters, although these differ between the two models. This highlights that even for methods with a larger hyperparameter space, only a few parameters are crucial for optimization in this context.



(a) MicSim_FluoMT [19]: Six 666 × 666 examples from fluorescence microscopy simulations of *in vivo* astral MTs during *C. elegans* mitosis. Top row shows samples from the "simple" variant; bottom row from the "complex" variant with dense, overlapping filament networks.



(b) DRIFT [28]: Six 512 × 512 synthetic images (Dataset F configuration: kernel size 3, step size 15) generated for generic curved filament segmentation. These abstract geometric structures lack biological context and imaging artifacts characteristic of real microscopy data.

Figure A.5. Representative samples from related synthetic MT datasets. While no publicly available dataset directly addresses our specific task of segmenting single *in vitro* MTs growing from fixed seeds under IRM microscopy, we compare against the closest available alternatives. (a) MicSim_FluoMT [19] simulates *in vivo* fluorescence microscopy of astral MTs during mitosis in *C. elegans*, featuring dense, overlapping filament networks. (b) DRIFT [28] generates synthetic curved filamentous structures for general segmentation tasks. Both datasets differ fundamentally from ours in imaging modality (fluorescence vs. IRM), biological context (*in vivo* vs. *in vitro*), structural complexity (dense networks vs. individual filaments), and scope: neither provides human assessment of synthetic quality nor systematic benchmarking across foundation models, which are central contributions of `SynthMT`.

Figure A.6. Web-based interface for expert assessment of perceptual realism of synthetic microscopy images. Participants viewed a single image at a time and rated it along five predefined dimensions using a 7-point Likert scale: structural fidelity of MTs, background realism, lighting and blurring realism, noise pattern realism, and overall quality. Images were presented in randomized order, and each image was rated exactly once per participant.

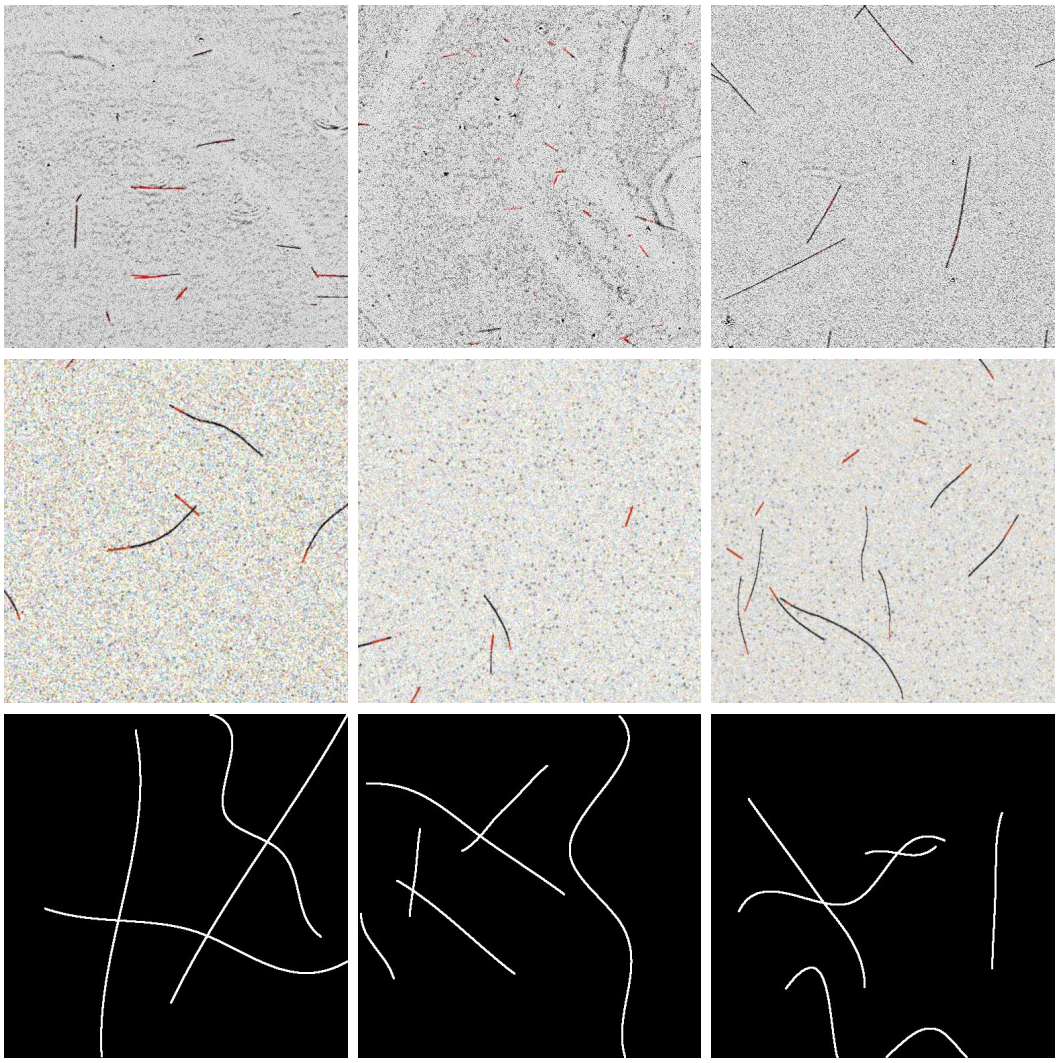


Figure A.7. Representative images from the human validation study. Nine examples from the evaluation set shown to domain experts: three real IRM images (top row), three synthetic images from `SynthMT` (middle row), and three synthetic images from DRIFT [28] (bottom row). Experts rated each image on multiple quality dimensions without knowing its source, allowing us to quantitatively assess how closely `SynthMT` matches the appearance and characteristics of authentic microscopy data compared to existing synthetic alternatives.

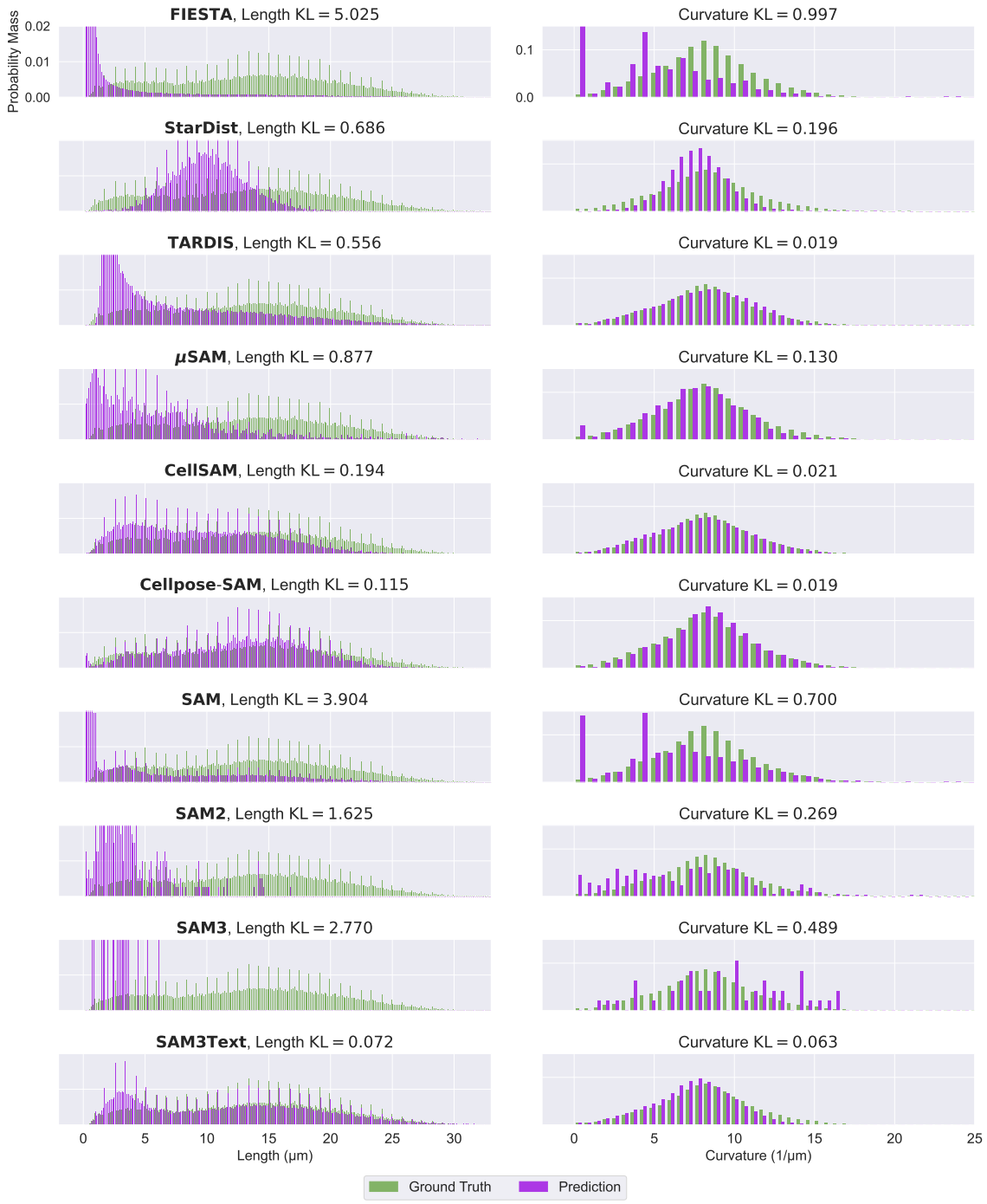


Figure A.8. Predicted and ground-truth distributions of length and curvature for all methods with their default parameters.

Figure A.9. Predicted and ground-truth distributions of length and curvature for all methods with tuned parameters obtained through Hyperparameter Optimization (HPO).

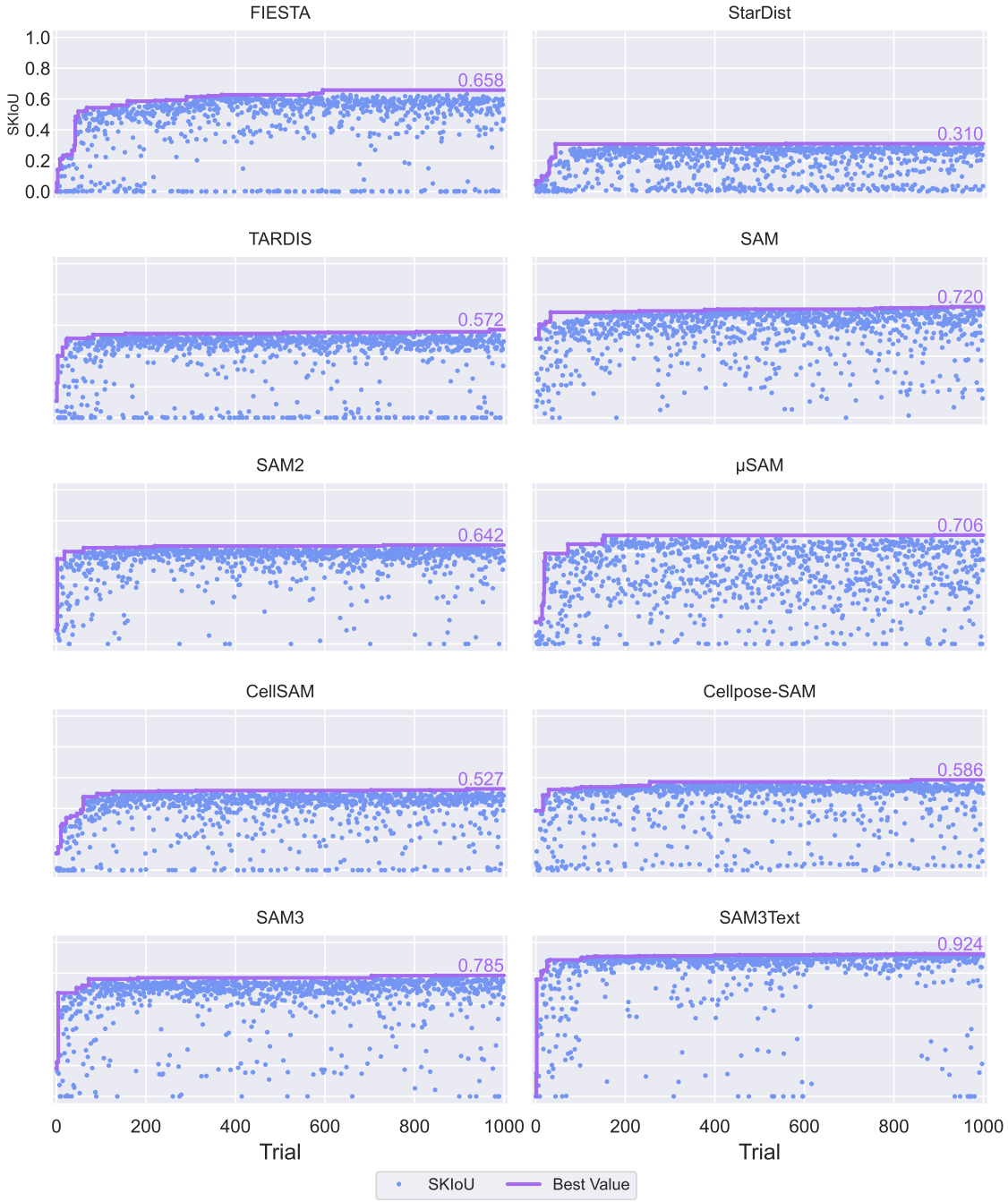


Figure A.10. HPO trajectories highlight rapid optimization. We plot the best SKIoU value observed up to each of the 1000 HPO trials for each method, using 10 images from `SynthMT` for optimization. Most methods approach their optimal performance within the first 200 trials, demonstrating efficient convergence. FIESTA, which has the largest number of hyperparameters (see Table A.1), requires approximately 600 trials to converge. This illustrates both the speed of improvement and the eventual plateau in performance.

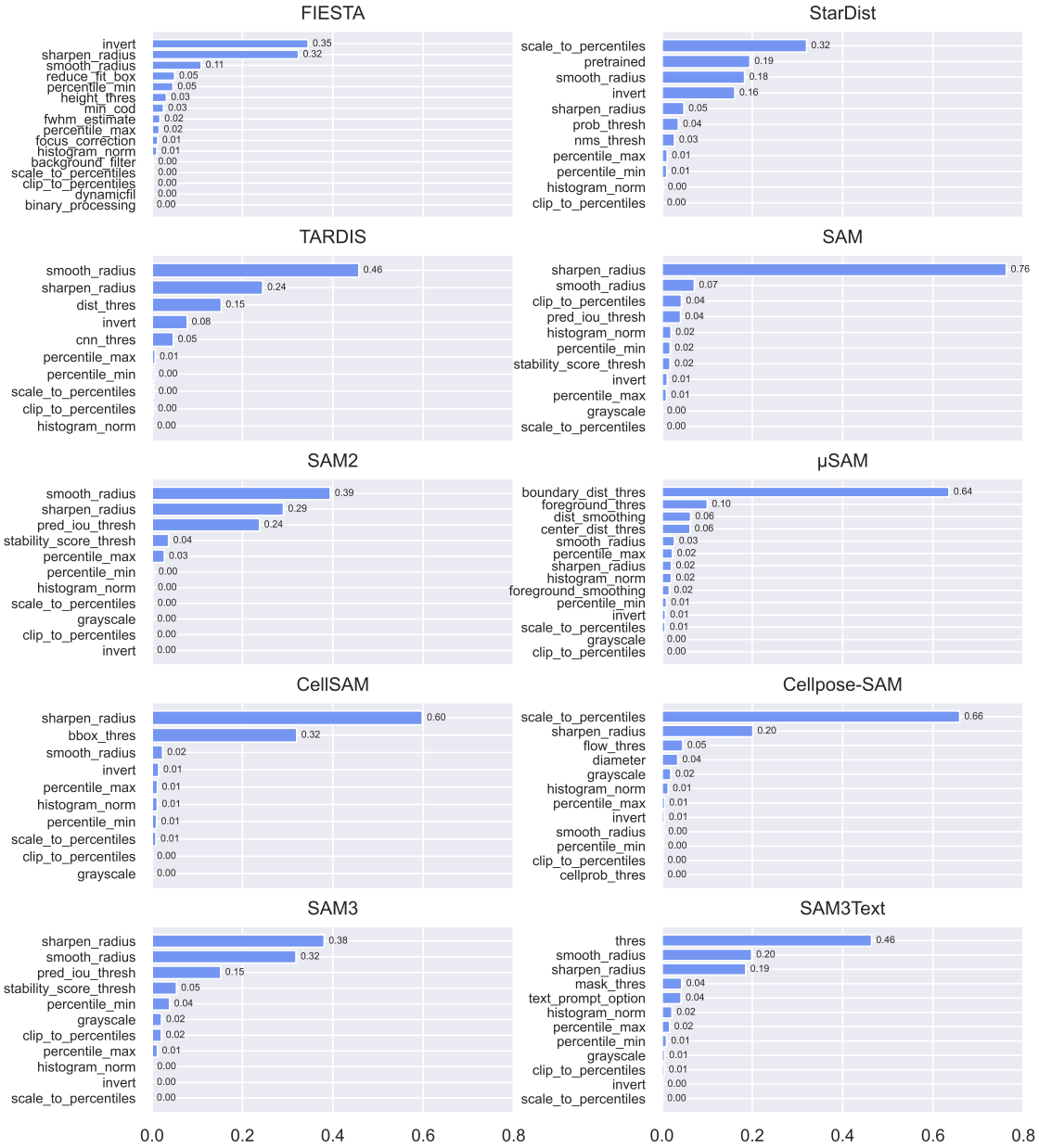


Figure A.11. A few key hyperparameters drive method performance. Parameter importance for all models, calculated via f-ANOVA on the 10 `SynthMT` images used for optimization. The y-axis shows the percentage of variance explained by each hyperparameter, with higher values indicating greater impact. For most models, performance is dominated by a small number of influential parameters, which are often related to preprocessing steps.