FACET: FACTORIZED ASYMMETRIC CONDITIONING FOR EFFICIENT TRANSPORT IN HIGH-FIDELITY FLUORESCENCE MICROSCOPY SYNTHESIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Fluorescence microscopy reveals where proteins localize, but only a limited number of proteins can be imaged in the same cell; generating these images from amino-acid sequence and the cell's morphological context enables in silico localization of unimaged proteins. The two conditions, however, play asymmetric roles: morphological context is spatially aligned with the target, whereas sequence is non-spatial and must specify protein-dependent localization within it, with recurring coarse patterns shared across proteins and finer protein-specific variation. Existing generators condition on both jointly, without separating what each explains. We introduce FACET (*Factorized Asymmetric Conditioning for Efficient Transport*), a probabilistic generative framework that encodes this structure as an explicit inductive bias: sequence semantics are learned from what context leaves unexplained, coarse localization regularities are shared across proteins through a semantic memory, and protein-specific variation is a bounded residual around them. A variance-preserving state projection further lets FACET perform continuous stochastic transport through a pretrained diffusion predictor with minimal parameter overhead. On held-out proteins, FACET improves spatial overlap by 34.3% on the Human Protein Atlas and 14.0% on OpenCell over a backbone-matched baseline, and reduces FID by 27.2% and 46.5%, respectively, with 75% fewer network evaluations. It also substantially improves protein-association structure recovery and yields better-calibrated predictions, while detailed ablations show complementary contributions from its design choices. These results identify factorized asymmetric conditioning, rather than generator capacity alone, as a key lever for high-fidelity, efficient, and biologically meaningful cellular image synthesis.

1 INTRODUCTION

Protein subcellular localization is a spatial phenotype that links molecular identity to cellular function and is directly observed through fluorescence microscopy (Thul et al., 2017). Recent sequence-to-image models generate these phenotypes from protein sequence and cellular context (Khwaja et al., 2023; Zheng & Huang, 2025; Zhang et al., 2025). Yet these conditions play fundamentally different roles: context is spatially aligned with the target, whereas sequence is non-spatial and must specify protein-dependent localization semantics within that context. The conditions are therefore asymmetric in representation and explanatory role, rather than in biological importance.

Our key insight is that protein localization is inherently context-relative and hierarchically organized, spanning from recurring regularities shared across proteins to finer protein-specific variation. This motivates three coupled modeling requirements. First, sequence representations should capture localization information beyond what context already explains: even a linear map from context features explains over a third of held-out target-feature variance (Appendix B.2), a lower bound on the redundancy that sequence semantics need not re-encode. Second, recurring coarse localization regularities should be shared across proteins rather than relearned independently. Third, finer protein-specific variation should remain distinct from both the shared regularities and image stochasticity.

We introduce FACET, a probabilistic generative framework that translates this hierarchy into asymmetric conditioning: sequence semantics are factorized into coarse semantics, retrieved from a memory built on context-residualized features, and finer protein-specific variation, which jointly condition stochastic transport through a pretrained diffusion model. This formulation encodes an

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

explicit inductive bias: sequence semantics are learned from what context leaves unexplained, coarse localization regularities are shared across proteins, and protein-specific variation is bounded relative to them, a structure that favors generalization to unseen proteins.

FACET improves spatial overlap by 34.3% on the Human Protein Atlas (HPA; Thul et al., 2017) and 14.0% on OpenCell (Cho et al., 2022) benchmarks over a backbone-matched baseline, and reduces FID by 27.2% and 46.5%, respectively, with 75% fewer network evaluations. It also strengthens protein-association separation by 39–87% across association-confidence thresholds and reduces calibration error by 74.7%.

Our major contributions are:

- *Factorized asymmetric conditioning.* We introduce FACET, a generative framework that assigns cellular context, shared localization semantics, protein-specific variation, and image stochasticity to separate conditioning roles, using a context-residualized semantic memory while preserving their hierarchical dependencies.
- *Principled latent-variable formulation with efficient transport.* We derive a conditional latent-variable model whose evidence lower bound decomposes into coarse-semantic, protein-specific, and transport terms, each optimized blockwise by our staged objective, and establish an exact variance-preserving state projection that enables continuous stochastic transport through a pretrained finite-schedule latent diffusion predictor with under 0.6% parameter overhead.
- *Broad empirical gains and controlled analyses.* On HPA and OpenCell, FACET substantially improves spatial agreement and distributional fidelity with 75% fewer network evaluations. Controlled ablations isolate the contributions of its semantic, hierarchical, and transport components, while further analyses show cross-dataset transfer, consistent gains across multi-channel context compositions, stronger recovery of protein-association patterns, and improved predictive calibration.

Related work. Sequence-conditioned localization imaging has progressed from bidirectional sequence-image transformers (Khwaja et al., 2023) and sequence- and landmark-conditioned prediction (Zhang et al., 2025) to diffusion conditioned on sequence and morphology (Zheng & Huang, 2025); concurrent preprints scale identity-conditioned generation to 12,800 human proteins (Sun et al., 2026) or adopt autoregressive generation (Kambhampati et al., 2026). These works mainly advance the image model; FACET instead factorizes sequence semantics and cellular context within a conditional generator, complementing any pretrained variance-preserving backbone, so we compare in a backbone-matched setting. Extended discussion on related work is provided in Appendix A.

2 FACET: Factorized Asymmetric Conditioning for Efficient Transport

2.1 Factorized probabilistic model

Modeling objective. We study conditional protein spatial-phenotype generation, where a target image Y is generated from protein sequence S and cellular morphology context C. While C specifies the spatial environment, sequence-dependent localization spans coarse regularities shared across proteins and finer protein-specific variation. We represent this hierarchy with a reusable semantic memory $\mathcal{M} = \{V_r\}_{r=1}^{K}$ of coarse localization prototypes, a prototype-valued latent R for sequence-dependent prototype selection, and a continuous residual δz_S for variation beyond the prototypes. This factorization organizes coarse and fine sequence variation while retaining direct conditioning on both S and C in the image model.

With target image encoding $z_1 = E_Y(Y)$, we model

$$p_\Theta(Y \mid S, C, \mathcal{M}) = \sum_R \int p(Y \mid z_1)\, p_\Theta(z_1, z_0, R, \delta z_S \mid S, C, \mathcal{M})\, dz_1\, dz_0\, d\delta z_S, \tag{1}$$

Latent-variable factorization. With a fixed, deterministic decoder D, we take $p(Y \mid z_1) = \delta(Y - D(z_1))$, so learning the conditional image distribution reduces to modeling the latent distribu-

tion. We factorize the latter as

$$p_\Theta(z_1, z_0, R, \delta z_S \mid S, C, \mathcal{M}) = \underbrace{p_0(z_0)}_{\text{base distribution}} \underbrace{p_\eta(R \mid S, \mathcal{M})}_{\text{prototype prior}} \underbrace{p_\psi(\delta z_S \mid S, R)}_{\text{residual prior}} \underbrace{p_\theta(z_1 \mid z_0, S, C, R, \delta z_S)}_{\text{conditional transport}}. \tag{2}$$

Here, η and ψ parameterize the prototype and residual priors, respectively. After integrating out the initial latent z_0, this becomes

$$p_\Theta(z_1, R, \delta z_S \mid S, C, \mathcal{M}) = p_\eta(R \mid S, \mathcal{M})\, p_\psi(\delta z_S \mid S, R)\, p_\theta(z_1 \mid S, C, R, \delta z_S). \tag{3}$$

Since R is prototype-valued, its selected memory state directly conditions the residual and transport factors.

These roles are deliberately asymmetric: C explains cell-specific spatial organization, R supplies coarse sequence-dependent semantics from $\mathcal{M}$, δz_S captures fine-grained variations beyond the prototypes, and p_θ models realization-level image variation.

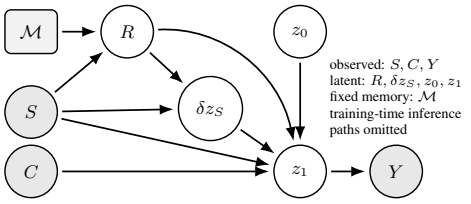

Figure 1: Probabilistic graphical model for FACET.

Proposition 1 (Conditional evidence lower bound)**.** *For any variational posterior* $q(R, \delta z_S \mid z_1, S, C, \mathcal{M}) = q_\omega(R \mid z_1, S, C, \mathcal{M})\, q_\phi(\delta z_S \mid z_1, S, C, R, \mathcal{M})$*, we have*

$$\log p_\Theta(z_1 \mid S, C, \mathcal{M}) \geq \mathbb{E}_q\left[\log p_\theta(z_1 \mid S, C, R, \delta z_S)\right] - D_{\mathrm{KL}}(q_\omega(R \mid z_1, S, C, \mathcal{M}) \,\|\, p_\eta(R \mid S, \mathcal{M})) - \mathbb{E}_{q_\omega} D_{\mathrm{KL}}(q_\phi(\delta z_S \mid z_1, S, C, R, \mathcal{M}) \,\|\, p_\psi(\delta z_S \mid S, R)). \tag{4}$$

Here D_{KL} is the Kullback-Leibler divergence, and ω, ϕ parameterize the amortized posteriors. See Appendix D.1 for proof.

2.2 Context-residualized semantic memory

Context residualization. To avoid relearning spatial information already available from C, we construct memory from context-residualized target features. With fixed features $\Phi_Y(Y_i)$ and $\Phi_C(C_i)$, an affine ridge predictor b fit on training data defines $U_i = \Phi_Y(Y_i) - b(\Phi_C(C_i))$, retaining target features unexplained by context. A fitted T_C compresses the context prediction as $z_C = f_C(C) = T_C(b(\Phi_C(C)))$. Further details and held-out explanatory analysis are in Appendix B.2.

Protein-level prototype construction. For each training protein p, we average residuals over its images $\mathcal{I}_p$ and apply a fixed standardized PCA projection T as $e_p = T\left(\frac{1}{|\mathcal{I}_p|}\sum_{i\in\mathcal{I}_p} U_i\right)$, giving proteins equal weight and a compact shared representation. A fixed annotation-derived rule assigns each protein to one of K coarse semantic states; for proteins $\mathcal{P}_k$ assigned to state k, we define the centroid $V_k = \frac{1}{|\mathcal{P}_k|}\sum_{p\in\mathcal{P}_k} e_p$ and memory $\mathcal{M} = \{V_k\}_{k=1}^K$. The projection, assignments, and centroids use training proteins only and are then frozen; construction details and dimensions are in Appendix B.2.

Sequence-conditioned soft memory readout. At inference, target images and annotation-derived states are unavailable; the sequence prior instead yields the soft readout $m_S = \mathbb{E}[R \mid S, \mathcal{M}] = \sum_{r=1}^K p_\eta(R = V_r \mid S, \mathcal{M}) V_r$. This inference-time approximation preserves graded semantics without discrete sampling noise.

2.3 Bounded variational sequence residual

The residual δz_S captures fine-grained protein-specific variation beyond the coarse readout m_S.

Target-informed posterior and sequence-only prior. We define the refined semantic state as $z_S = m_S + \delta z_S$ and protein language model encoding as $h_S = E_S(S)$. For each training image, the posterior input is the projected residual $U_Y = T(\Phi_Y(Y) - b(\Phi_C(C)))$. The target-informed amortized posterior and sequence-only prior are $q_\phi(\delta z_S \mid U_Y, m_S) = \mathcal{N}\big(\mu_q(U_Y, m_S), \sigma_z^2 I\big)$ and $p_\psi(\delta z_S \mid h_S, m_S) = \mathcal{N}\big(\mu_p(h_S, m_S), \sigma_z^2 I\big)$, respectively.

Fixed-scale divergence and bounded parameterization. Both distributions use the same fixed isotropic covariance. Consequently,

$$D_{\mathrm{KL}}(q_\phi \| p_\psi) = \frac{\|\mu_q - \mu_p\|_2^2}{2\sigma_z^2}. \tag{5}$$

The fixed scale prevents the optimizer from reducing the divergence by inflating a learned variance (Appendix D.5). We parameterize the residual scale relative to the empirical coarse-memory magnitude s_m as $\sigma_z = c_\sigma s_m$, and bound the posterior and prior means using $\mu_a = c_\mu s_m \tanh f_a(\cdot),\ c_\sigma, c_\mu > 0, a \in \{q, p\}$. Thus, the fine residual has an explicit scale relative to the coarse semantic state. The implementation details are reported in Appendix E.

2.4 Asymmetric stochastic transport

2.4.1 Variance-preserving stochastic interpolant

Stochastic interpolant. We learn continuous stochastic transport from $z_0 \sim \mathcal{N}(0, I)$ to the target latent z_1. For $\tau \in [0, 1]$, define the stochastic interpolant (Albergo et al., 2025)

$$x_\tau = A_\tau z_1 + B_\tau z_0 + G_\tau \xi, \qquad A_\tau = \tau, \quad B_\tau = 1 - \tau, \quad G_\tau = \kappa\tau(1 - \tau), \tag{6}$$

where $\xi \sim \mathcal{N}(0, I)$ and $\kappa > 0$. Defining $Q_\tau = \sqrt{B_\tau^2 + G_\tau^2}$ gives

$$x_\tau = A_\tau z_1 + Q_\tau \epsilon_\tau, \qquad \epsilon_\tau \sim \mathcal{N}(0, I).$$

Variance-preserving normalization. To leverage the learned image diffusion prior of a pretrained ε-predictor trained on a finite variance-preserving schedule, we normalize each continuous interpolant state x_τ onto the same variance-preserving family.

Proposition 2 (Variance-preserving state compatibility)**.** *Let $N_\tau = \sqrt{A_\tau^2 + Q_\tau^2}$ and $y_\tau = x_\tau / N_\tau$. Then*

$$y_\tau = a_\tau z_1 + \sigma_\tau \epsilon_\tau, \qquad a_\tau = \frac{A_\tau}{N_\tau}, \quad \sigma_\tau = \frac{Q_\tau}{N_\tau}, \tag{7}$$

with $a_\tau^2 + \sigma_\tau^2 = 1$ exactly.

Proposition 2 folds both noise sources into a single effective noise, so a frozen pretrained ε-predictor can drive continuous stochastic transport with minimal parameter overhead. Unlike Diff2Flow (Schusterbauer et al., 2025), which targets deterministic flow matching, it also covers the latent noise $G_\tau \xi$. The projection is exact; only the finite-schedule lookup is approximate:

$$k(\tau) = \arg\min_k |\tau_k - \tau|. \tag{8}$$

The schedule-specific representability details are provided in Appendix H.2.

Base velocity. Given the adapted noise estimate $\hat{\epsilon}_L$, i.e., the LoRA-adapted pretrained prediction followed by an ε-adapter, a lightweight module that adds an m_S-conditioned residual correction, we form a base clean estimate and velocity:

$$\hat{z}_1^L = \frac{x_\tau - Q_\tau \hat{\epsilon}_L}{\max(A_\tau, \epsilon_{\mathrm{num}})}, \qquad v_L = \hat{z}_1^L + \dot{Q}_\tau \hat{\epsilon}_L. \tag{9}$$

See Appendix D.2 for the full derivation and further details.

2.4.2 Context-first hierarchical correction

Ordered modulation. Let H_k be an intermediate image feature at native noise index k and e_τ a continuous-time embedding. FACET applies feature-wise linear modulation (FiLM) (Perez et al., 2018) in an ordered correction:

$$H_C = \mathrm{FiLM}_C(H_k, [z_C; e_\tau]), \qquad H_{CS} = \mathrm{FiLM}_S(H_C, [z_S; e_\tau]). \tag{10}$$

The sequence representation therefore refines a feature representation that has already absorbed cellular context rather than competing with context at the same injection point.

Gated sequence short-cut. A sequence short-cut and a lightweight residual correction head produce the velocity correction Δv:

$$H_{CS} \leftarrow H_{CS} + g(\tau)P_z(z_S), \qquad g(\tau) = 4\tau(1-\tau). \tag{11}$$

Here P_z projects the protein state to a channel-wise bias broadcast over the feature map. The correction head is zero-initialized so that the added path is inactive at initialization. The gate vanishes at both transport endpoints and has unit magnitude at the midpoint.

Role-specific conditioning paths. The resulting computation preserves the pretrained predictor's direct sequence input, uses m_S in the ε-adapter, uses $E_C(C)$ in the backbone and the compact context state $z_C = f_C(C)$ in the first FiLM block, and uses the refined state z_S only in the hierarchical sequence correction. This allocation allows the pretrained predictor to retain its original conditional pathway while the learned modules provide structured sequence-dependent refinement.

Velocity composition and parameter-efficient adaptation. The final velocity field is

$$v_\Theta(x_\tau, \tau, S, C) = v_L(x_\tau, \tau, S, C) + g(\tau)\Delta v(H_k, z_C, z_S, \tau). \tag{12}$$

The original image-predictor weights, VAE, and protein encoder remain frozen. Low-Rank Adaptation (LoRA) (Hu et al., 2022) supplies parameter-efficient updates to the ε-predictor, while the ε-adapter, semantic prior, continuous prior, and correction head are trained. The exact module configuration and parameter counts are provided in Appendix E.

2.5 Training and inference

Modular training. Rather than optimizing the ELBO jointly, FACET follows a staged procedure aligned with its factorization: we establish the context-residualized memory and context map, pretrain and freeze the prototype prior and memory-conditioned ε-adapter, and then optimize the continuous residual and conditional transport components. The transport-stage objective combines conditional flow matching (Lipman et al., 2023) with reconstruction, sequence-ranking, variational, and residual regularization:

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{\mathrm{FM}} + \lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{pix}}\mathcal{L}_{\mathrm{rec,pix}} + \lambda_{\mathrm{rank}}\mathcal{L}_{\mathrm{rank}} + \lambda_{\mathrm{VI}}D_{\mathrm{KL}}(q_\phi \| p_\psi) \\ + \lambda_{\mathrm{shrink}}\mathcal{L}_{\mathrm{shrink}} + \lambda_{\mathrm{same}}\mathcal{L}_{\mathrm{same}}.\end{aligned} \tag{13}$$

This staged procedure optimizes the terms of Eq. 4 blockwise: $\mathcal{L}_{\mathrm{FM}}$ serves as a surrogate for $\mathbb{E}_q[\log p_\theta(z_1 \mid \cdot)]$, $\lambda_{\mathrm{VI}}D_{\mathrm{KL}}(q_\phi \| p_\psi)$ is the residual KL term, and $\beta D_{\mathrm{KL}}(\mathrm{sg}[q_\omega] \| p_\eta)$ in $\mathcal{L}_{\mathrm{proto}}$ (Eq. 14) is the prototype KL term, whose cross-entropy terms anchor q_ω and p_η to annotation-derived states; the remaining losses are auxiliary regularizers.

Appendix B.1 details the stage-wise objectives, loss definitions, negative sampling, curriculum, and optimization; Algorithm 1 summarizes training.

Inference procedure. Figure 4 and Algorithm 2 summarize inference. Given sequence S and context C, FACET computes h_S, z_C, the soft memory readout m_S, and the derived semantic state $z_S = m_S + \delta z_S$. Starting from a noise latent, the pretrained, LoRA-adapted ε-predictor receives the projected transport state with $E_C(C)$, retains the direct h_S pathway, and uses a separate m_S-conditioned ε-adapter to produce the base velocity v_L. The context-first hierarchical head predicts a gated correction, yielding $v_\Theta = v_L + g(\tau)\Delta v$. The resulting SDE is integrated with Euler-Maruyama (Albergo et al., 2025; Song et al., 2021); the final step yields the base clean-latent estimate $\hat{z}_1^L$, decoded by the frozen VAE decoder. Training-only posterior and target-dependent reconstruction paths are omitted. See Appendices E and H.2 for details.

3 Experiments and Results

3.1 Experimental setting

Datasets and evaluation regimes. We evaluate FACET on protein-held-out HPA and OpenCell benchmarks. HPA contains 696 test images from 97 held-out proteins and provides nucleus, endoplasmic-reticulum, and microtubule context channels; we evaluate four compositions of these channels. OpenCell uses nucleus-only context and evaluates both HPA-adapted FACET, which transfers the HPA-trained FACET modules, and OpenCell-adapted FACET, which fits these modules on OpenCell data. Dataset construction and preprocessing are detailed in Appendix E.

Metrics and statistics. We report MI, MS-SSIM, Haralick texture agreement, PSNR, continuous spatial overlap ($\mathrm{IoU}_{\mathrm{cont}}$), FID, and network function evaluations (NFE). $\mathrm{IoU}_{\mathrm{cont}}$ is the generalized Jaccard (Ruzicka) similarity. For protein-association pattern recovery, we report Pearson-similarity separation and Mann-Whitney U tests; for predictive calibration, we report ECE, MCE, and Brier score. Metric definitions, aggregation, statistical tests, and sampling and calibration protocols are provided in Appendices F and G.

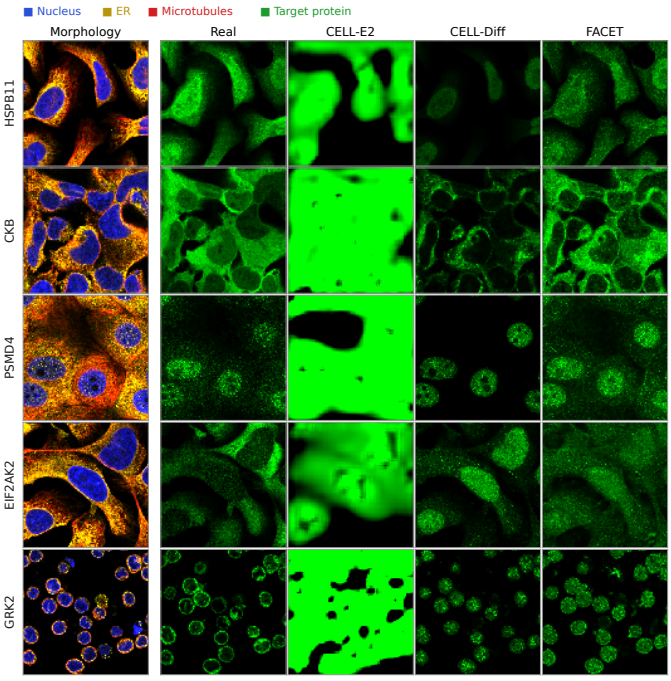

Figure 2: Representative held-out HPA examples: morphology context, target, and model outputs. CELL-E2 uses nucleus-only context; CELL-Diff and FACET use all three context channels.

Baselines and controls. We compare against two published generators, CELL-E2 (Khwaja et al., 2023) and CELL-Diff (Zheng & Huang, 2025), as well as several backbone-matched internal baselines: ε-adaptation-only control, pathway and inference-time memory ablations, adaptation and initialization variants, and sampler variants across NFE budgets. To ensure that performance differences reflect FACET's conditioning design rather than differences in backbone capacity or pretraining, all backbone-matched systems (CELL-Diff, FACET, and its internal variants) share the same pretrained ε-predictor, VAE, and ESM2 protein language model from the corresponding CELL-Diff checkpoint. CELL-Diff is, to our knowledge, the only public sequence-conditioned latent diffusion model with checkpoints for both benchmarks (Appendix E.3). In all FACET variants, the VAE and ESM2 remain frozen; the ε-predictor is either kept frozen or, in variants with backbone LoRA, adapted only through the introduced LoRA updates.

3.2 Conditional phenotype generation on HPA

3.2.1 Results on HPA

We first evaluate protein-conditioned spatial-phenotype generation on the held-out HPA test set. Tab. 1 summarizes the comparison across

Table 1: Comparison on the HPA test set. Higher is better for all metrics except NFE and FID. Best results are shown in bold.

Method	NFE ↓	MI ↑	MS-SSIM ↑	Haralick ↑	PSNR ↑	$\mathrm{IoU}_{\mathrm{cont}}$ ↑	FID ↓
CELL-E2	–	0.4041	0.1370	0.5644	6.36	0.1755	166.4
CELL-Diff	100	0.5355	0.3471	0.5182	15.77	0.2478	45.60
FACET	**25**	**0.5481**	**0.3651**	**0.5854**	**16.15**	**0.3328**	**33.20**

Table 2: Performance across cellular-context compositions on HPA. All systems use the same proteins, sequence inputs, and sampling protocol per composition. Bold marks the better value within each composition.

Context	System	MI ↑	MS-SSIM ↑	Haralick ↑	PSNR ↑	$\mathrm{IoU}_{\mathrm{cont}}$ ↑	FID ↓
Nucleus	CELL-Diff	**0.3091**	**0.2777**	0.4932	**15.32**	0.2047	**51.10**
	FACET	0.2519	0.1712	**0.5010**	14.66	**0.2768**	52.12
Nucleus + ER	CELL-Diff	0.4562	0.3154	0.4898	15.52	0.2182	60.00
	FACET	**0.5120**	**0.3413**	**0.5726**	**15.96**	**0.3268**	**33.83**
Nucleus + MT	CELL-Diff	0.5097	0.3404	0.5196	15.60	0.2435	47.60
	FACET	**0.5226**	**0.3524**	**0.5796**	**16.03**	**0.3266**	**33.24**
Nucleus + ER + MT	CELL-Diff	0.5355	0.3471	0.5182	15.77	0.2478	45.60
	FACET	**0.5481**	**0.3651**	**0.5854**	**16.15**	**0.3328**	**33.20**

image-quality metrics and sampling cost. FACET achieves the best value on all six reported image-quality metrics while requiring only 25 network function evaluations per image, compared with 100 for CELL-Diff. Relative to the baseline, it improves continuous intensity overlap from 0.2478 to 0.3328 and lowers FID from 45.6 to 33.20. Despite its added modules, FACET also reduces generation time for the 696-image test set from 78 to 15 minutes on the same GPU (Appendix G.2). Fig. 2 and Appendix Figs. 6, 7, and 9 show representative qualitative results.

3.2.2 Performance across cellular-context compositions

Tab. 2 evaluates the same checkpoint across four cellular-context compositions. FACET improves every reported metric in the two partial multi-channel regimes and the full-context regime. A withheld morphology channel is replaced by the pretrained predictor's learned placeholder embedding, in both the context canvas $E_C(C)$ and the features from which z_C is computed, so every system sees the same ablated context. Under nucleus-only context, it retains gains in continuous intensity overlap and Haralick agreement, whereas CELL-Diff is higher on the remaining reported metrics. Representative qualitative results are depicted in Appendix Fig. 5.

3.3 Cross-dataset evaluation

We next evaluate HPA-module transfer on OpenCell, using the nucleus image as the cellular context. For both CELL-Diff and FACET, the pretrained ε-predictor backbone, VAE encoder/decoder, and ESM2 protein language model are taken from the OpenCell CELL-Diff checkpoint. The HPA-adapted FACET variant transfers the FACET-specific modules from HPA, whereas the OpenCell-adapted variant fit these modules directly on OpenCell training data. As shown in Tab. 3, HPA-module transfer improves MI slightly, Haralick agreement, PSNR, continuous intensity overlap, and FID relative to CELL-Diff, although MS-SSIM is lower. OpenCell-adaptation improves MI, MS-SSIM, Haralick agreement, continuous intensity overlap, and FID relative to the HPA-adapted variant, with only a small decrease in PSNR. See qualitative results in Appendix Fig. 8.

Table 3: Comparison on the OpenCell test set (nucleus-only context). All systems except CELL-E2 share the OpenCell-trained CELL-Diff backbone; HPA-adapted transfers the FACET modules from HPA without refitting, whereas OpenCell-adapted fits them on OpenCell. Best results are in bold.

Method	NFE ↓	MI ↑	MS-SSIM ↑	Haralick ↑	PSNR ↑	IoU_{cont} ↑	FID ↓
CELL-E2	–	0.1649	0.1020	0.4542	11.31	0.1779	245.8
CELL-Diff	100	0.2732	**0.4078**	0.3663	13.20	0.3722	68.90
FACET (HPA-adapted)	**25**	0.2753	0.3653	0.5177	**13.85**	0.4222	43.33
FACET (OpenCell-adapted)	**25**	**0.2998**	0.3893	**0.5199**	13.75	**0.4244**	**36.83**

3.4 Spatially resolved protein-association pattern recovery

To assess whether generated images preserve protein-association patterns, we adapt the localization-matched evaluation of Sun et al. (2026). We use STRING (Szklarczyk et al., 2023), a database of known and predicted functional protein associations, solely to obtain external evaluation labels. We generate the 97 proteins in the HPA test split under eight fixed cellular contexts, using matched context and noise seeds across systems. For each protein pair, we compute within-cell Pearson similarity after removing the per-cell mean image across proteins. This suppresses similarity caused by shared cellular morphology. We then compare STRING-associated pairs with non-associated pairs that share at least one annotated compartment. STRING labels are used only for evaluation and do not enter model training or conditioning.

Table 4: Spatially resolved protein-association pattern recovery under a localization-matched evaluation adapted from (Sun et al., 2026). The baseline denotes CELL-Diff. Pearson similarity is computed after removing the per-cell mean image. Δ measures the separation between STRING-associated and matched non-associated pairs; positive values indicate stronger recovery.

System	STRING score threshold	Pairs (assoc./null)	Pearson Δ ↑	$p_{\text{assoc. vs. null}}$
Baseline	0.85	47/2659	+0.1005	0.298
FACET	0.85	47/2659	**+0.1875**	1.3×10^{-3}
Baseline	0.70	65/2641	+0.0954	0.170
FACET	0.70	65/2641	**+0.1718**	4.5×10^{-4}
Baseline	0.50	112/2594	+0.0922	0.042
FACET	0.50	112/2594	**+0.1369**	2.5×10^{-4}
Baseline	0.40	164/2542	+0.0925	0.0093
FACET	0.40	164/2542	**+0.1289**	1.7×10^{-5}

We report the difference

$$\Delta = \mathbb{E}[r(a,b) \mid \text{STRING-associated}] - \mathbb{E}[r(a,b) \mid \text{matched null}],$$

Table 5: Controlled FACET configuration ablation on the HPA benchmark. Each configuration is evaluated on the same held-out proteins using identical cellular context, sampler, and preprocessing. Checkmarks indicate active pathways. "hier." represents "hierarchical".

FACET configuration	ε-adapter	Memory-conditioned semantic pathway	Backbone LoRA (base velocity v_L)	Hierarchical head (velocity correction Δv)	$\mathrm{IoU}_{\mathrm{cont}}\uparrow$
FACET (full)	✓	✓	✓	✓	**0.3328**
Memory-conditioned + LoRA	✓	✓	✓	–	0.3236
Memory-conditioned + hier. head	✓	✓	–	✓	0.3052
Memory-conditioned	✓	✓	–	–	0.2826
ε-adaptation-only	✓	–	–	–	0.2581

Table 6: HPA-module transfer and OpenCell adaptation ablation. A checkmark indicates that the corresponding component is active. Note that, for all the configurations, the ε-predictor backbone, VAE encoder/decoder, and protein LM are adopted from the OpenCell checkpoint by Zheng & Huang (2025).

Configuration	OpenCell adaptation	HPA module initialization	OpenCell-fit memory	$\mathrm{IoU}_{\mathrm{cont}}\uparrow$
FACET, HPA-adapted	–	✓	–	0.4222
FACET, OpenCell-adapted, cold start	✓	–	✓	**0.4244**
Warm start, inherited HPA memory	✓	✓	–	0.4155
Warm start, refitted OpenCell memory	✓	✓	✓	0.4233

where positive values indicate that STRING-associated proteins are rendered more similarly. As shown in Tab. 4, FACET produces a larger positive separation than CELL-Diff at all four STRING thresholds. At STRING score ≥ 0.85, FACET increases Δ from $+0.1005$ to $+0.1875$, while at STRING score ≥ 0.40, it increases Δ from $+0.0925$ to $+0.1289$. All four FACET contrasts are significant under the two-sided Mann-Whitney U test (Mann & Whitney, 1947), with p values ranging from 1.7×10^{-5} to 1.3×10^{-3}. These results indicate that FACET recovers association-consistent spatial patterns beyond shared cellular context.

3.5 Component ablations and additional analyses

3.5.1 Pathway ablation

We compare controlled FACET configurations under a common evaluation protocol. Tab. 5 shows that the complete configuration performs best. Adding the sequence-conditioned semantic pathway raises $\mathrm{IoU}_{\mathrm{cont}}$ from 0.2581 to 0.2826, while adding either the backbone LoRA or the hierarchical velocity-correction head provides further improvement. The full configuration, which includes both components, achieves the highest observed value of 0.3328.

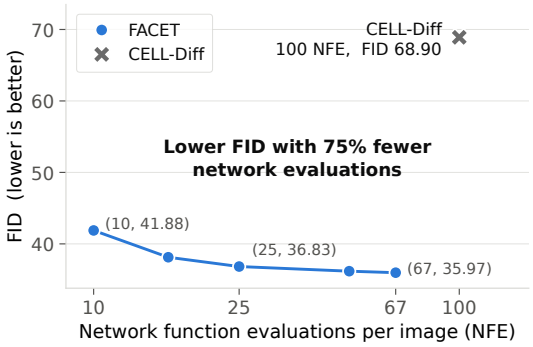

Figure 3: Quality–cost comparison on OpenCell.

3.5.2 Quality-cost trade-off

The quality-cost comparison is summarized in Fig. 3. The OpenCell-adapted FACET model reaches lower FID with substantially fewer network function evaluations than CELL-Diff. At 25 NFE, it achieves FID $= 36.83$, compared with 68.90 for CELL-Diff at 100 NFE, corresponding to a 75% reduction in network evaluations per image.

3.5.3 Ablation on adaptation configurations

To examine the roles of initialization and OpenCell adaptation, we compare the corresponding configurations in Tab. 6. The comparison shows that performance is sensitive to both the initialization of the FACET-specific modules and whether the semantic memory is refitted on OpenCell data. The cold-start configuration with OpenCell-fitted memory gives the highest overlap (0.4244), while refitting the memory in the warm-start setting nearly recovers the cold-start performance.

3.5.4 Ablation of inference-time memory

To isolate the contribution of the learned semantic memory at inference, we keep all other FACET components fixed and replace the learned memory with either the uniform mean of the prototypes or a zero memory. As shown in Tab. 7, the unmodified learned memory achieves the highest $\mathrm{IoU}_{\mathrm{cont}}$

Table 7: Ablation of inference-time memory on HPA. A checkmark indicates that the corresponding memory configuration is active.

Configuration	Learned memory	Memory replaced at inference	$\mathrm{IoU}_{\mathrm{cont}} \uparrow$
FACET, unmodified memory	✓	–	**0.3328**
FACET, uniform prototype mean	–	✓	0.2820
FACET, zero memory	–	✓	0.3014

of 0.3328. Replacing it with the uniform prototype mean or zero memory reduces performance to 0.2820 and 0.3014, respectively. Zero memory degrades less because the standardized PCA space is centered, so zero approximates the frequency-weighted prototype average, whereas the uniform mean overweights rare localization categories and injects a protein-agnostic bias. This result indicates that FACET benefits from the protein-conditioned organization of the learned memory rather than from memory injection alone.

3.5.5 Sampler ablation

We compare the Euler-Maruyama SDE sampler with a deterministic probability-flow ODE integrated using Heun's method (Karras et al., 2022) on OpenCell (Tab. 8). At matched NFE, both samplers reach nearly identical FID, while the SDE sampler yields slightly higher $\mathrm{IoU}_{\mathrm{cont}}$. Both remain well below CELL-Diff at 100 NFE (FID 68.90), indicating that FACET's efficiency gains are not specific to the choice of solver.

Table 8: OpenCell sampling results across solvers and network function evaluations (NFE).

Sampler	NFE	FID ↓	$\mathrm{IoU}_{\mathrm{cont}} \uparrow$
SDE	10	41.88	0.4301
SDE	25	36.83	0.4244
SDE	50	36.18	0.4214
ODE (Heun's)	25	36.60	0.4205
ODE (Heun's)	50	36.17	0.4187

3.5.6 Predictive calibration

To evaluate calibration and overall probabilistic forecast quality, we report ECE, MCE, and the Brier score using 16 samples per input. We follow the post-hoc temperature-scaling procedure of Guo et al. (2017), fitting the temperature on a protein-disjoint validation partition and keeping it fixed for test evaluation. As shown in Tab. 9, FACET substantially improves all three calibration measures: ECE decreases from 0.21501 to 0.05442, MCE from 0.24886 to 0.14850, and Brier score from 0.23372 to 0.19915. These results indicate that FACET produces more reliable predictive foreground probabilities. This provides additional evidence that the factorized stochastic predictor is not only accurate but also better calibrated under repeated sampling.

Table 9: Predictive calibration on HPA. One temperature per model, fit on a protein-disjoint validation split and held fixed on test. Foreground probabilities come from 16 predictive samples per input.

Method	ECE ↓	MCE ↓	Brier ↓
CELL-Diff	0.21501	0.24886	0.23372
FACET	**0.05442**	**0.14850**	**0.19915**

4 Conclusion

We introduced FACET, a factorized probabilistic framework that turns the asymmetric, hierarchical structure of protein localization into an explicit inductive bias atop a pretrained diffusion prior. On unseen proteins across two benchmarks, it substantially improves spatial and distributional fidelity at a fraction of the sampling cost, while improving protein-association recovery and calibration. With the backbone held fixed, these gains are attributable to FACET's factorized probabilistic formulation rather than generator capacity, making it complementary to stronger backbones and a template for conditional generation whenever inputs differ in spatial alignment and explanatory power.

Scope and future directions. FACET inherits its backbone's finite noise schedule, which bounds the usable sampling steps; continuous-time variance-preserving backbones lift this directly. Extending FACET to 3D and time-lapse imaging, multi-protein generation, and perturbations are natural next steps toward in silico spatial proteomics.

REFERENCES

Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

Lawrence D Brown, T Tony Cai, and Anirban DasGupta. Interval estimation for a binomial proportion. *Statistical science*, pp. 101–117, 2001.

Nathan H. Cho, Keith C. Cheveralls, Andreas-David Brunner, Kibeom Kim, André C. Michaelis, Preethi Raghavan, Hirofumi Kobayashi, Laura Savy, Jason Y. Li, Hera Canaj, James Y. S. Kim, Edna M. Stewart, Christian Gnann, Frank McCarthy, Joana P. Cabrera, Rachel M. Brunetti, Bryant B. Chhun, Greg Dingle, Marco Y. Hein, Bo Huang, Shalin B. Mehta, Jonathan S. Weissman, Rafael Gómez-Sjöberg, Daniel N. Itzhak, Loïc A. Royer, Matthias Mann, and Manuel D. Leonetti. OpenCell: Endogenous tagging for the cartography of human cellular organization. *Science*, 375 (6585):eabi6983, 2022. doi: 10.1126/science.abi6983.

Yunfeng Fan, Wenchao Xu, Haozhao Wang, Junxiao Wang, and Song Guo. PMR: Prototypical modal rebalance for multimodal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20029–20038, 2023.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020. doi: 10.1038/s42256-020-00257-z.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330, 2017.

Robert M. Haralick, K. Shanmugam, and Its'Hak Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3(6):610–621, 1973. doi: 10.1109/TSMC.1973.4309314.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Desmond J Higham. An algorithmic introduction to numerical simulation of stochastic differential equations. *SIAM review*, 43(3):525–546, 2001.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Yu Huang, Junyang Lin, Chang Zhou, Hongxia Yang, and Longbo Huang. Modality competition: What makes joint training of multi-modal network fail in deep learning? (Provably). In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9226–9259, 2022.

Michael F Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 18(3):1059–1076, 1989.

Harold Jeffreys. An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society of London. Series A, Mathematical and physical sciences*, pp. 453–461, 1946.

Sandeep Kambhampati, Eric Zimmermann, Emre Hayir, Kevin K. Yang, Fei Chen, and Alex X. Lu. Vermeer: Autoregressive generative modeling of microscopy predicts protein localization. *bioRxiv*, 2026. doi: 10.64898/2026.06.01.729395.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Emaad Khwaja, Yun S. Song, Aaron Agarunov, and Bo Huang. CELL-E 2: Translating proteins to pictures and back with a bidirectional text-to-image transformer. In *Advances in Neural Information Processing Systems*, volume 36, pp. 4899–4914, 2023.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

Jeffrey T. Leek, Robert B. Scharpf, Héctor Corrada Bravo, David Simcha, Benjamin Langmead, W. Evan Johnson, Donald Geman, Keith Baggerly, and Rafael A. Irizarry. Tackling the widespread and critical impact of batch effects in high-throughput data. *Nature Reviews Genetics*, 11(10): 733–739, 2010. doi: 10.1038/nrg2825.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

Henry B Mann and Donald R Whitney. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, pp. 50–60, 1947.

Yunhong Min, Juil Koo, Seungwoo Yoo, and Minhyuk Sung. BézierFlow: Learning Bézier stochastic interpolant schedulers for few-step generation. In *International Conference on Learning Representations*, 2026.

Xiaokang Peng, Yake Wei, Andong Deng, Dong Wang, and Di Hu. Balanced multimodal learning via on-the-fly gradient modulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8238–8247, 2022.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pp. 3942–3951, 2018. doi: 10.1609/aaai.v32i1.11671.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Johannes Schusterbauer, Ming Gui, Frank Fundel, and Björn Ommer. Diff2Flow: Training flow matching models via diffusion model alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28347–28357, 2025.

Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. In *Advances in Neural Information Processing Systems*, volume 29, pp. 3738–3746, 2016.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Huangqingbo Sun, Konstantin Kahnert, Jan N. Hansen, William D. Leineweber, Mingyang Li, Wanyue Feng, Frederic Ballllosera Navarro, Ulrika Axelsson, Wei Ouyang, and Emma Lundberg. Generative machine learning unlocks the first proteome-wide image of human cells. *bioRxiv*, 2026. doi: 10.64898/2026.03.31.715748.

Damian Szklarczyk, Rebecca Kirsch, Mikaela Koutrouli, Katerina Nastou, Farrokh Mehryary, Radja Hachilif, Annika L. Gable, Tao Fang, Nadezhda T. Doncheva, Sampo Pyysalo, Peer Bork, Lars J. Jensen, and Christian von Mering. The STRING database in 2023: Protein–protein association networks and functional enrichment analyses for any sequenced genome of interest. *Nucleic Acids Research*, 51(D1):D638–D646, 2023. doi: 10.1093/nar/gkac1000.

Peter J. Thul, Lovisa Åkesson, Mikaela Wiking, Diana Mahdessian, Aikaterini Geladaki, Hammou Ait Blal, Tove Alm, Anna Asplund, Lars Björk, Lisa M. Breckels, Anna Bäckström, Frida Danielsson, Linn Fagerberg, Jenny Fall, Laurent Gatto, Christian Gnann, Sophia Hober, Martin Hjelmare, Fredric Johansson, Sunjae Lee, Cecilia Lindskog, Jan Mulder, Claire M. Mulvey, Peter Nilsson, Per Oksvold, Johan Rockberg, Rutger Schutten, Jochen M. Schwenk, Åsa Sivertsson, Evelina Sjöstedt, Marie Skogs, Charlotte Stadler, Devin P. Sullivan, Hanna Tegel, Casper Winsnes, Cheng Zhang, Martin Zwahlen, Adil Mardinoglu, Fredrik Pontén, Kalle von Feilitzen, Kathryn S. Lilley, Mathias Uhlén, and Emma Lundberg. A subcellular map of the human proteome. *Science*, 356 (6340):eaal3321, 2017. doi: 10.1126/science.aal3321.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024.

Weiyao Wang, Du Tran, and Matt Feiszli. What makes training multi-modal classification networks hard? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12695–12705, 2020.

Xuesong Wang, Yimin Fan, Yucheng Guo, Chenghao Fu, Kinhei Lee, Khachatur Dallakyan, Yaxuan Li, Qijin Yin, Yu Li, and Le Song. Prediction of cellular morphology changes under perturbations with a transcriptome-guided diffusion model. *Nature Communications*, 16(1):8210, 2025. doi: 10.1038/s41467-025-63478-z.

Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The thrity-seventh asilomar conference on signals, systems & computers, 2003*, volume 2, pp. 1398–1402. Ieee, 2003.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3836–3847, 2023.

Xinyi Zhang, Yitong Tseo, Yunhao Bai, Fei Chen, and Caroline Uhler. Prediction of protein subcellular localization in single cells. *Nature methods*, 22(6):1265–1275, 2025.

Dihan Zheng and Bo Huang. Bridging protein sequences and microscopy images with unified diffusion models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 78139–78155, 2025.

A Related Work

Asymmetric multimodal conditioning. Joint multimodal training can under-utilize one modality even when all inputs are informative. Gradient Blending identified different optimization and generalization rates across modalities (Wang et al., 2020), gradient modulation and prototype rebalancing mitigate modality imbalance (Peng et al., 2022; Fan et al., 2023), and theoretical work shows that joint gradient descent can produce modality competition in simplified settings (Huang et al., 2022). These works primarily study discriminative fusion. FACET addresses a distinct generative setting in which conditions differ in representation, spatial alignment, and explanatory role, assigning them distinct but coupled generative roles rather than rebalancing their contributions to a shared discriminative objective.

Protein and cellular image generation. Protein-localization image generation has been approached with bidirectional sequence–image transformers conditioned on nucleus context (Khwaja et al., 2023), sequence- and landmark-conditioned image prediction (Zhang et al., 2025), diffusion conditioned on protein sequence and cellular morphology (Zheng & Huang, 2025), conditional diffusion using cellular landmarks and learned protein identities (Sun et al., 2026), and recent channel-adaptive autoregressive generation from sequence and cellular landmarks (Kambhampati et al., 2026). Perturbation-conditioned diffusion models instead predict morphology from transcriptomic or perturbation information (Wang et al., 2025). HPA (Thul et al., 2017) and OpenCell (Cho et al., 2022) provide complementary protein-localization microscopy resources. Our focus is not a new image backbone, but how sequence-dependent localization and cellular morphology are structured within a probabilistic conditional generator, including an architecture-controlled comparison to CELL-Diff.

Diffusion, stochastic flows, and few-step transport. Diffusion models construct generative processes through progressive corruption and denoising (Ho et al., 2020; Song et al., 2020; 2021), latent diffusion moves this process to a compressed image representation (Rombach et al., 2022). Flow matching learns continuous vector fields from tractable conditional paths (Lipman et al., 2023), rectified flow emphasizes direct transport paths (Liu et al., 2023), stochastic interpolants unify deterministic and stochastic transports (Albergo et al., 2025), and optimal-transport couplings can improve flow training (Tong et al., 2024). Recent methods bridge pretrained diffusion and flow models through interpolant and timestep alignment (Schusterbauer et al., 2025), or accelerate few-step generation by learning stochastic-interpolant schedules (Min et al., 2026). FACET uses stochastic flow matching as its transport component and introduces an explicit variance-preserving state projection that makes continuous transport states compatible with a pretrained finite-schedule diffusion predictor.

Hierarchical latent variables and parameter-efficient conditioning. Variational autoencoders provide amortized latent-variable inference (Kingma & Welling, 2014), while hierarchical variants show how multiple stochastic levels can represent different factors of variation (Sønderby et al., 2016). Feature-wise Linear Modulation (FiLM) provides a general mechanism for conditional affine modulation (Perez et al., 2018). Low-Rank Adaptation (LoRA) trains additive low-rank weight updates while retaining pretrained parameters (Hu et al., 2022), and ControlNet demonstrates parameter-efficient conditioning around a strong pretrained image prior (Zhang et al., 2023). FACET uses variational residual inference, FiLM, and LoRA within an asymmetric factorization that assigns cellular context, coarse sequence semantics, fine protein-specific variation, and image stochasticity distinct but coupled generative roles.

Context residualization, shortcuts, and invariance. Shortcut learning concerns reliance on easy predictive cues (Geirhos et al., 2020), batch effects arise from acquisition or processing variation (Leek et al., 2010), and domain-adversarial learning suppresses domain-identifying information to obtain invariance (Ganin et al., 2016). FACET differs from these objectives: cellular morphology is a scientifically meaningful conditioning variable and remains directly available to the generator rather than being treated as a nuisance signal to remove. Context residualization is applied only when constructing the semantic memory, encouraging sequence semantics to capture localization information beyond what context already explains rather than enforcing invariance to context. The objective remains high-fidelity conditional generation from both sequence and cellular context, not removal of either condition.

B Additional Method Details

B.1 Training objective and curriculum

The probabilistic formulation in the main text specifies the roles of the prototype-valued state R, continuous residual δz_S, and image latent z_1. Rather than optimizing the ELBO jointly, we adopt a modular training procedure: we first construct the context-residualized memory and pretrain the semantic prototype prior and ε-adapter, then train the continuous residual and conditional transport model.

Discrete semantic objective. Let r denote the annotation-derived prototype state for a training protein. The discrete semantic components are fit using

$$\mathcal{L}_{\text{proto}} = \text{CE}(q_\omega, r) + \beta D_{\text{KL}}(\text{sg}[q_\omega] \,\|\, p_\eta) + \gamma \,\text{CE}(p_\eta, r), \tag{14}$$

where $\text{sg}[\cdot]$ denotes stop-gradient. The cross-entropy term trains q_ω against the annotation-derived state; only in the KL term is q_ω treated as a stop-gradient target for p_η. Thus, the divergence updates the prior without moving the posterior in the reverse direction.

Memory-conditioned ε-adapter pretraining. We pretrain the ε-adapter using noise prediction, sequence ranking, a prior KL term $D_{\text{KL}}(\text{sg}[q_\omega]\|p_\eta)$, and an adapter-output penalty, while continuing to update (ω, η). The ε-adapter and prototype prior are then frozen for transport training.

Flow-matching and reconstruction losses. As derived in Appendix D.2, for a sampled τ, the conditional flow target is

$$u_\tau^\star = z_1 + \dot{Q}_\tau \epsilon_\tau, \qquad \mathcal{L}_{\text{FM}} = \mathbb{E}\left[\|v_\Theta - u_\tau^\star\|_2^2\right]. \tag{15}$$

For reconstruction supervision, we derive a *training-time* auxiliary latent estimate by retaining the adapted predictor's noise estimate $\hat{\epsilon}_L$:

$$\hat{z}_1^{\,\theta} := \hat{z}_1^{\,L} + g(\tau)\Delta v, \tag{16}$$

which follows from the velocity decomposition in Eqs. 9 and 12; its derivation is provided in Appendix D.3. This fixed-noise assignment is a training-time auxiliary construction, not the inference emission. Let D denote the frozen decoder and d the residual dimension. The auxiliary terms are

$$\mathcal{L}_{\text{rec}} = \mathbb{E}\left[\left\|\hat{z}_1^{\,\theta} - z_1\right\|_2^2\right], \qquad \mathcal{L}_{\text{rec,pix}} = \mathbb{E}\left[\left\|D(\hat{z}_1^{\,\theta}) - D(z_1)\right\|_2^2\right], \tag{17}$$

$$\mathcal{L}_{\text{shrink}} = \mathbb{E}\left[\|g(\tau)\Delta v\|_2^2\right], \qquad \mathcal{L}_{\text{same}} = \mathbb{E}\left[\frac{\left\|\mu_q(U_Y^{(i)}, m_S) - \mu_q(U_Y^{(j)}, m_S)\right\|_2^2}{d\, s_m^2}\right]. \tag{18}$$

Here i and j index two images of the same protein. The reconstruction terms supervise the auxiliary latent estimate, while flow matching supervises the composed velocity. The shrinkage term keeps the hierarchical branch a correction to the base estimate field. The same-protein term encourages the residual representation to capture protein-specific rather than cell-specific variation, consistent with the sequence-conditioned prior.

Sequence-ranking objective. For a target image, we evaluate the flow-matching loss using the matched sequence and n mismatched sequences while holding the interpolant state, timestep, and noise draws fixed. This produces a contrastive sequence-ranking loss

$$\mathcal{L}_{\text{rank}} = -\log \frac{\exp(-\mathcal{L}_{\text{FM}}^{+}/\tau_c)}{\exp(-\mathcal{L}_{\text{FM}}^{+}/\tau_c) + \sum_{j=1}^{n} \exp(-\mathcal{L}_{\text{FM}}^{(j)}/\tau_c)}, \tag{19}$$

Here $\mathcal{L}_{\text{FM}}^{+}$ and $\mathcal{L}_{\text{FM}}^{(j)}$ denote the per-candidate squared velocity errors $\|v_\Theta - u_\tau^\star\|_2^2$, not separately averaged dataset losses. The matched training candidate uses its target-informed residual posterior; each mismatched candidate uses its own sequence-conditioned prior. Negatives are sampled by first selecting a localization prototype other than the target's uniformly and then sampling a training protein uniformly within that prototype. Each negative is evaluated using its own sequence prior, matching the inference-time use of the model.

Implemented flow-stage objective. The final flow-stage objective is

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{\text{FM}} &+ \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{\text{pix}}\mathcal{L}_{\text{rec,pix}} + \lambda_{\text{rank}}\mathcal{L}_{\text{rank}} + \lambda_{\text{VI}} D_{\text{KL}}(q_\phi \| p_\psi) \\ &+ \lambda_{\text{shrink}}\mathcal{L}_{\text{shrink}} + \lambda_{\text{same}}\mathcal{L}_{\text{same}}.\end{aligned} \tag{20}$$

The continuous variational term uses the fixed-scale divergence from Eq. 5. Its numerical coefficient should therefore be interpreted together with the explicit factor $1/(2\sigma_z^2)$ rather than as an indication that the residual is negligible.

Staged curriculum. Stage I constructs the semantic memory and context map, fits the prototype prior with Eq. 14, and then trains the ε-adapter while continuing to update the prior; both are frozen before transport training. Stage II has three phases: (i) 2000 steps with $z_S = m_S$, without D_{KL} or $\mathcal{L}_{\mathrm{same}}$; (ii) 2000 steps in which δz_S is sampled from the posterior q_ϕ and λ_{VI} is ramped linearly over the first 1000 steps; and (iii) 3000 steps at constant λ_{VI}, in which δz_S is instead drawn from the prior p_ψ with a probability ramped linearly from 0 to 1 over the first 1500 steps, matching its use at inference. In phases (ii) and (iii), a quarter of posterior draws use another image of the same protein. Algorithm 1 summarizes the training process. At inference, the reported point-estimate experiments use the prior-mean state of Eq. 26 and the reported predictive-sampling experiments instead draw the protein-specific residual from its prior using the reparameterization as shown in Eq. 27.

B.2 Prototype construction details

Feature extraction and residualization. The same frozen VAE encoder E produces the spatial target latent $z_1 = E_Y(Y) = E(Y) \in \mathbb{R}^{4\times64\times64}$ and context encoding $E_C(C) = [E(C_{\mathrm{nucl}}); E(C_{\mathrm{ER}}); E(C_{\mathrm{MT}})] \in \mathbb{R}^{12\times64\times64}$. Pooling and flattening give

$$\Phi_Y(Y) = \mathrm{flatten}(\mathrm{AvgPool}_4(E_Y(Y))) \in \mathbb{R}^{1024},$$
$$\Phi_C(C) = \mathrm{flatten}(\mathrm{AvgPool}_4(E_C(C))) \in \mathbb{R}^{3072}.$$

The target latent $z_1 = E_Y(Y)$ summarize training images, whereas $E_C(C)$ remains available at inference. An affine ridge predictor b is fit using training proteins only, and the per-image residual is

$$U_i = \Phi_Y(Y_i) - b(\Phi_C(C_i)). \tag{21}$$

The held-out context-to-target explanatory analysis gives an R^2 of 0.377. This analysis motivates constructing sequence-dependent memory in the residual feature space rather than directly in the raw target-feature space. The same fitted ridge map also supplies the context branch: $z_C = f_C(C) = T_C(b(\Phi_C(C))) \in \mathbb{R}^{128}$. Here T_C standardizes the ridge predictions, applies a 128-component PCA, and rescales the components. The ridge map and T_C are fitted on training data and then frozen; b is not refitted for this branch.

Protein-level averaging and projection. For a training protein p with image set $\mathcal{I}_p$, we compute

$$e_p = T\left(\frac{1}{|\mathcal{I}_p|}\sum_{i\in\mathcal{I}_p} U_i\right), \tag{22}$$

where T is a fixed standardized principal-component projection. Protein-level averaging gives each training protein equal weight regardless of its number of images. The reported instantiation uses a 128-dimensional representation that retains 71.2% of the explained variance. The projection T is fitted on protein-averaged residuals, distinct from the context-prediction projection T_C. For each training image, the target-informed posterior receives the projected residual $U_Y = T(\Phi_Y(Y) - b(\Phi_C(C))) \in \mathbb{R}^{128}$; paired images of the same protein are projected identically.

Prototype vocabulary and assignments. The memory contains $K = 9$ prototype vectors based on UniProt-localization annotation. The annotation rule assigns each memory-construction protein to a coarse semantic state. If $\mathcal{P}_k$ is the set assigned to state k, the prototype is

$$V_k = \frac{1}{|\mathcal{P}_k|}\sum_{p\in\mathcal{P}_k} e_p, \qquad \mathcal{M} = \{V_k\}_{k=1}^K. \tag{23}$$

All feature transformations, assignments, and centroids are estimated from training proteins and frozen before evaluation. At inference, the prior $p_\eta(R \mid S, \mathcal{M})$ predicts a distribution over the prototype-valued states, and the deployed readout is

$$m_S = \sum_{r=1}^{K} p_\eta(R = V_r \mid S, \mathcal{M}) V_r. \tag{24}$$

Target images and annotation-derived states are not provided to this prior at inference.

Prototype posterior and prior parameterization. Both distributions are parameterized by three-layer MLPs with width 512 and SiLU activations (denoted by g_ω and g_η below), producing logits over the $K = 9$ prototypes:

$$q_\omega(R = V_r \mid z_1, C, S, \mathcal{M}) = \mathrm{softmax}\left(g_\omega([\widetilde{\Phi}_Y(Y); \widetilde{\Phi}_C(C); \widetilde{h}_S])\right)_r,$$
$$p_\eta(R = V_r \mid S, \mathcal{M}) = \mathrm{softmax}\left(g_\eta(\widetilde{h}_S)\right)_r, \tag{25}$$

with $\Phi_Y(Y) = \mathrm{flatten}(\mathrm{AvgPool}_4(z_1)) = \mathrm{flatten}(\mathrm{AvgPool}_4(E_Y(Y)))$. The tilde denotes standardization using training-set statistics; $\widetilde{h}_S$ is the standardized mean-pooled sequence embedding, excluding special tokens. The annotation label $r \in \{1, \ldots, K\}$ identifies the corresponding prototype $V_r \in \mathcal{M}$ and supervises both distributions; the prototype vectors are not directly input to either network.

B.3 Architecture and implementation details

Representations and initialization. The frozen VAE encoder E provides the target latent $z_1 = E_Y(Y) = E(Y)$ and, for HPA, the spatial context $E_C(C) = [E(C_{\mathrm{nucl}}); E(C_{\mathrm{ER}}); E(C_{\mathrm{MT}})]$. A fitted, frozen context map yields $z_C = f_C(C)$, while the frozen protein encoder produces $h_S = E_S(S)$. Target-image encoding supplies training targets and semantic features but is not required at inference. Generation starts from $z_0 \sim p_0 = \mathcal{N}(0, I)$ and decodes the generated latent using the frozen decoder D.

Frozen and trainable components. The pretrained error-predictor weights, VAE encoder and decoder, and protein sequence encoder remain frozen. The fitted ridge map b, projections T and T_C, and semantic memory are also frozen during transport training. The predictor receives the spatial context encoding $E_C(C)$ concatenated channel-wise with y_τ; the hierarchical context FiLM instead receives the compact $z_C = f_C(C)$. The prototype prior and ε-adapter are pretrained and subsequently frozen. Transport training updates the continuous residual components, context-first hierarchical head, and LoRA parameters in the ε-predictor. The reported instantiation adds 9.83M deployed trainable parameters (0.588% of the 1.67B-parameter instantiation); training-only posterior parameters are reported separately in Table 11.

Sequence semantic state. For the default point-estimate experiments, we use the prior mean

$$z_S = m_S + \mu_p(h_S, m_S). \tag{26}$$

Predictive-sampling experiments instead draw the protein-specific residual from its prior using the reparameterization

$$z_S = m_S + \mu_p(h_S, m_S) + \sigma_z\, \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, I). \tag{27}$$

Fixed settings. The following values are fixed reproducibility settings for the reported FACET instantiation and *are not claimed to be individually optimal*. For the reported experiments, we fix the residual-scale constants to $c_\sigma = 0.05$ and $c_\mu = 0.25$.

Table 10: Selected implementation settings for the FACET instantiation used in the reported experiments; component-level ablations are reported separately.

Setting	Value
Semantic memory vocabulary	$K = 9$ fixed UniProt localization categories
Residual representation	128 dimensions
Interpolant strength	$\kappa = 0.5$
Correction head	2 residual blocks
LoRA adaptation	rank 16
Inference budget	25 NFE

Relation to residual and additive parameterizations. FACET employs residual operations at four levels: context-residualized target features, a continuous semantic residual around the coarse memory, a residual correction to the pretrained noise estimate, and a correction to the base velocity field. These operations serve distinct purposes within the factorized probabilistic model. In particular, the deployed semantic state $z_S = m_S + \delta z_S$ is additive, but the conditional generator is not restricted to an additive decomposition of independent context and sequence functions. For example, sequential FiLM modulation introduces multiplicative interactions through the product $(1 + \gamma_S(z_S)) \odot (1 + \gamma_C(z_C))$. Moreover, the base velocity v_L is itself conditioned jointly on S and C, and the composed velocity $v_\Theta = v_L + g(\tau)\Delta v$ is integrated along an evolving stochastic trajectory, so additive velocity composition does not imply additive decomposition of the generated image.

B.4 Transport discretization and inference

Finite-schedule lookup. The continuous interpolant is mapped to the native variance-preserving schedule using the exact normalization in Proposition 2. For any pretrained predictor that is defined at finite native noise levels, the projected state is queried at

$$k(\tau) = \arg\min_k |\tau_k - \tau|. \tag{28}$$

The state projection itself preserves the variance-preserving identity exactly; the finite-index lookup is the remaining schedule-specific approximation. Here τ_k denotes the continuous coordinate associated with native VP noise index k; lookup selects a represented native level rather than a continuously adjustable predictor timestep.

Inference-time transport. At inference, the model computes m_S from the protein sequence, obtains z_S from the sequence-prior, and evaluates the composed velocity

$$v_\Theta = v_L + g(\tau)\Delta v. \tag{29}$$

The latent trajectory is integrated with the reported Euler-Maruyama sampler (Albergo et al., 2025; Song et al., 2021). The reported uniform sampling range is $\tau_{\min} = 0.05$ to $\tau_{\max} = 1$, with $\Delta\tau = (\tau_{\max} - \tau_{\min})/n$. The primary operating point uses $n = 25$ network function evaluations and $n - 1$ state updates; the final evaluation occurs at $\tau_{\max} - \Delta\tau$ and emits the base clean-latent estimate $\hat{z}_1^L$, which is decoded as $\hat{Y} = D(\hat{z}_1^L)$. Initialization at nonzero $\tau_{\min}$ uses a noise-derived latent and is an approximation to the target-dependent interpolant marginal. The deterministic ODE variant, integrated using Heun's method (Karras et al., 2022), is evaluated only as a sampling ablation (Tab. 8).

C Algorithms

Algorithms 1 and 2 summarize FACET training and inference, respectively. Here, $\mathrm{AdaptedPredictor}(\cdot)$ denotes the pretrained ε-predictor augmented with LoRA and the ε-adapter, reusing the sequence embedding h_S and memory readout m_S computed during conditioning. Fig. 4 depicts the inference pipeline.

D Proofs, Derivations, and other theoretical properties

D.1 Proof of Proposition 1

Since R is a prototype-valued discrete variable supported on $\mathcal{M}$,

$$p_\Theta(z_1 \mid S, C, \mathcal{M}) = \sum_{R \in \mathcal{M}} \int p_\Theta(z_1, R, \delta z_S \mid S, C, \mathcal{M})\, \mathrm{d}\delta z_S. \tag{30}$$

Multiplying and dividing the integrand by $q(R, \delta z_S \mid z_1, S, C, \mathcal{M})$ and applying Jensen's inequality gives

$$\log p_\Theta(z_1 \mid S, C, \mathcal{M}) \geq \mathbb{E}_q \left[\log \frac{p_\Theta(z_1, R, \delta z_S \mid S, C, \mathcal{M})}{q(R, \delta z_S \mid z_1, S, C, \mathcal{M})} \right]. \tag{31}$$

Algorithm 1 FACET training

Require: Training data $\mathcal{D} = \{(S_i, C_i, Y_i, r_i)\}$, frozen pretrained ε-predictor, encoders and decoder, loss weights and curriculum
Ensure: Trained FACET model and fixed memory $\mathcal{M}$

Stage I: Memory, context map, and prior pretraining

Extract $\{\Phi_Y(Y_i), \Phi_C(C_i)\}_i$ following Appendix B.2
Fit ridge map b; compute U_i using Eq. 21
Fit T; construct and freeze $\mathcal{M}$ using Eqs. 22–23
Fit T_C; freeze $f_C = T_C \circ b \circ \Phi_C$ following Appendix B.2
for minibatches (S_i, C_i, Y_i, r_i) **do**
    $z_{1,i} \leftarrow E_Y(Y_i)$; evaluate $q_\omega(\cdot \mid z_{1,i}, S_i, C_i; \mathcal{M})$ and $p_\eta(\cdot \mid S_i; \mathcal{M})$ using Eq. 25
    Update (ω, η) using $\mathcal{L}_{\text{proto}}$ in Eq. 14
end for
Pretrain the ε-adapter while updating (ω, η) following Appendix B.1
Freeze the ε-adapter and p_η

Stage II: Residual and transport training

Trainable: LoRA, correction head, and variational residual (ϕ, ψ). Curriculum: Appendix B.1.

for transport-training minibatches **do**
    Sample (S, C, Y) and a same-protein image $(C^{(j)}, Y^{(j)})$
    $z_1 \leftarrow E_Y(Y)$; $z_C \leftarrow f_C(C)$; $h_S \leftarrow E_S(S)$
    $m_S \leftarrow \sum_{r=1}^{K} p_\eta(R = V_r \mid S, \mathcal{M}) V_r$ ▷ Soft readout
    if phase (i) **then**
        $\delta z_S \leftarrow 0$
    else
        $U_Y \leftarrow T(\Phi_Y(Y) - b(\Phi_C(C)))$; $U_Y^{(j)} \leftarrow T(\Phi_Y(Y^{(j)}) - b(\Phi_C(C^{(j)})))$
        $\bar{\mu}_q \leftarrow \mu_q(U_Y, m_S)$; $\bar{\mu}_q^{(j)} \leftarrow \mu_q(U_Y^{(j)}, m_S)$; $\bar{\mu}_p \leftarrow \mu_p(h_S, m_S)$
        if phase (iii) and $\text{Bernoulli}(\rho)$ **then** ▷ ρ ramps from 0 to 1
            $\mu \leftarrow \bar{\mu}_p$
        else if $\text{Bernoulli}(0.25)$ **then**
            $\mu \leftarrow \bar{\mu}_q^{(j)}$ ▷ Same-protein swap
        else
            $\mu \leftarrow \bar{\mu}_q$
        end if
        $\delta z_S \leftarrow \mu + \sigma_z \varepsilon, \varepsilon \sim \mathcal{N}(0, I)$
    end if
    $z_S \leftarrow m_S + \delta z_S$

Stochastic interpolant

    Sample native index k with $\tau_k \geq \tau_{\min}$ and $Q_{\tau_k} > 0$; set $\tau \leftarrow \tau_k$
    Sample $z_0, \xi \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$
    $x_\tau \leftarrow A_\tau z_1 + B_\tau z_0 + G_\tau \xi$
    $\epsilon_\tau \leftarrow (B_\tau z_0 + G_\tau \xi)/Q_\tau$; $u_\tau^\star \leftarrow z_1 + \dot{Q}_\tau \epsilon_\tau$

Conditional transport

    $y_\tau \leftarrow x_\tau / N_\tau$
    $(\hat{\epsilon}_L, H_k) \leftarrow \text{AdaptedPredictor}(y_\tau, k, S, E_C(C), m_S)$ ▷ Adapted ε-predictor
    $\hat{z}_1^L \leftarrow (x_\tau - Q_\tau \hat{\epsilon}_L) / \max(A_\tau, \epsilon_{\text{num}})$
    $v_L \leftarrow \hat{z}_1^L + \dot{Q}_\tau \hat{\epsilon}_L$
    $v_\Theta \leftarrow v_L + g(\tau) \Delta v(H_k, z_C, z_S, \tau)$
    $\hat{z}_1^\theta \leftarrow \hat{z}_1^L + g(\tau) \Delta v(H_k, z_C, z_S, \tau)$ ▷ Training only

Losses and optimization

    Compute $\mathcal{L}_{\text{FM}}$, $\mathcal{L}_{\text{rec}}$, $\mathcal{L}_{\text{rec,pix}}$, and $\mathcal{L}_{\text{shrink}}$ using Eqs. 15–18 ($\mathcal{L}_{\text{rec,pix}}$ on every fourth step)
    For one example per minibatch, sample n negatives (Appendix B.1)
    Evaluate each at the same (x_τ, τ, C) with its own z_S (Eq. 26; $z_S = m_S$ in phase (i))
    Compute $\mathcal{L}_{\text{rank}}$ using Eq. 19
    if phase (ii) or (iii) **then**
        Compute $D_{\text{KL}}(q_\phi \| p_\psi)$ from $(\bar{\mu}_q, \bar{\mu}_p)$ using Eq. 5
        Compute $\mathcal{L}_{\text{same}}$ from $(\bar{\mu}_q, \bar{\mu}_q^{(j)})$ using Eq. 18
    end if
    Form $\mathcal{L}$ using Eq. 20
    Update LoRA, the correction head, and, from phase (ii), (ϕ, ψ) using $\nabla \mathcal{L}$
end for
return Trained model and $\mathcal{M}$

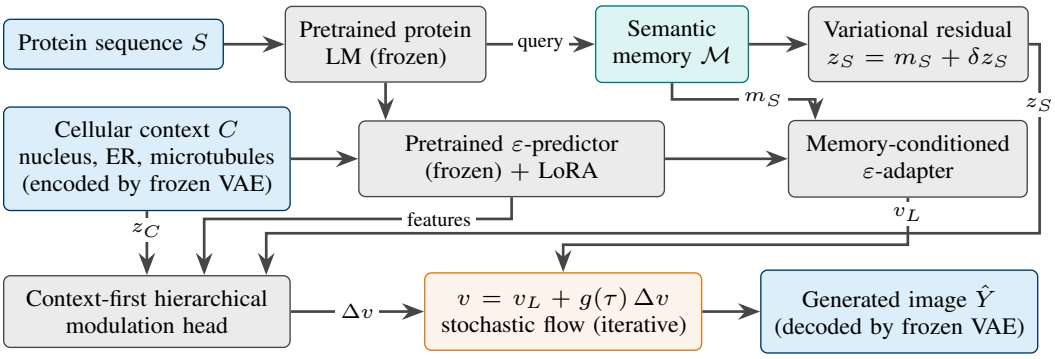

Figure 4: FACET inference overview. The protein sequence directly conditions the pretrained ε-predictor and queries the semantic memory to obtain m_S, while a sequence-conditioned residual refines it into z_S. The memory-conditioned ε-adapter forms the base velocity v_L, and context-first hierarchical modulation produces the correction Δv. Their composition $v_\Theta = v_L + g(\tau)\Delta v$ is iteratively integrated through stochastic latent transport, yielding the final latent estimate for decoding by the frozen VAE. Training-only paths are omitted; see Algorithm 2 for details.

Algorithm 2 FACET stochastic inference

Require: Protein sequence S, cellular context C, fixed memory $\mathcal{M} = \{V_r\}_{r=1}^{K}$, trained model, sampling parameters $(n, \tau_{\min}, \tau_{\max}, \eta_0)$

Ensure: Generated fluorescence image $\hat{Y}$

Conditioning

$z_C \leftarrow f_C(C)$; $h_S \leftarrow E_S(S)$

$m_S \leftarrow \sum_{r=1}^{K} p_\eta(R = V_r \mid S, \mathcal{M}) V_r$ ▷ Soft memory readout

$\delta z_S \leftarrow \mu_p(h_S, m_S)$ ▷ Default prior mean

Optional predictive sampling: $\delta z_S \leftarrow \mu_p(h_S, m_S) + \sigma_z \varepsilon$, $\varepsilon \sim \mathcal{N}(0, I)$

$z_S \leftarrow m_S + \delta z_S$

Initialization

Sample $z_0, \xi \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$

$x_{\tau_{\min}} \leftarrow B_{\tau_{\min}} z_0 + G_{\tau_{\min}} \xi$ ▷ $A_{\tau_{\min}} z_1$ omitted

$\Delta\tau \leftarrow (\tau_{\max} - \tau_{\min})/n$

Stochastic transport

for $j = 0, \ldots, n-1$ **do**

  $\tau \leftarrow \tau_{\min} + j\Delta\tau$

  $y_\tau \leftarrow x_\tau / N_\tau$ ▷ VP projection

  $k \leftarrow \arg\min_\ell |\tau_\ell - \tau|$ ▷ Eq. 8

  $(\hat{\epsilon}_L, H_k) \leftarrow \text{AdaptedPredictor}(y_\tau, k, S, E_C(C), m_S)$ ▷ Adapted ε-predictor

  $\hat{z}_1^L \leftarrow (x_\tau - Q_\tau \hat{\epsilon}_L) / \max(A_\tau, \epsilon_{\text{num}})$

  $v_L \leftarrow \hat{z}_1^L + \dot{Q}_\tau \hat{\epsilon}_L$

  $v_\Theta \leftarrow v_L + g(\tau)\Delta v(H_k, z_C, z_S, \tau)$ ▷ Apply hierarchical correction

  if $j < n-1$ **then**

    $s_\Theta \leftarrow -\hat{\epsilon}_L / Q_\tau$

    $\eta_\tau \leftarrow \eta_0 \tau(1-\tau)$

    Sample $\zeta_j \sim \mathcal{N}(0, I)$

    $x_{\tau+\Delta\tau} \leftarrow x_\tau + (v_\Theta + \eta_\tau s_\Theta)\Delta\tau + \sqrt{2\eta_\tau \Delta\tau}\,\zeta_j$ ▷ Euler-Maruyama

  end if

end for

Emission

$\hat{Y} \leftarrow D(\hat{z}_1^L)$ ▷ Last evaluation, $\tau = \tau_{\max} - \Delta\tau$

return $\hat{Y}$

Substituting the generative factorization in Eq. 3 and the posterior factorization from Proposition 1, the log-ratio decomposes as

$$\log p_\theta(z_1 \mid S, C, R, \delta z_S) + \log \frac{p_\eta(R \mid S, \mathcal{M})}{q_\omega(R \mid z_1, S, C, \mathcal{M})} + \log \frac{p_\psi(\delta z_S \mid S, R)}{q_\phi(\delta z_S \mid z_1, S, C, R, \mathcal{M})}. \tag{32}$$

Taking expectations therefore yields

$$\log p_\Theta(z_1 \mid S, C, \mathcal{M}) \geq \mathbb{E}_q[\log p_\theta(z_1 \mid S, C, R, \delta z_S)] - D_{\mathrm{KL}}(q_\omega \| p_\eta) - \mathbb{E}_{q_\omega} D_{\mathrm{KL}}(q_\phi \| p_\psi),$$

which is Eq. 4.

D.2 Derivation of the base latent estimate and velocity

We derive the base clean-latent estimate and velocity in Eq. 9 from the Gaussian-reparameterized stochastic interpolant. Recall that

$$x_\tau = A_\tau z_1 + B_\tau z_0 + G_\tau \xi, \qquad A_\tau = \tau, \tag{33}$$

where $z_0, \xi \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$ are independent of the target latent z_1. Define

$$Q_\tau = \sqrt{B_\tau^2 + G_\tau^2}, \qquad \epsilon_\tau = \frac{B_\tau z_0 + G_\tau \xi}{Q_\tau}. \tag{34}$$

For $Q_\tau > 0$, independence and Gaussianity imply

$$\epsilon_\tau \sim \mathcal{N}(0, I), \qquad x_\tau = A_\tau z_1 + Q_\tau \epsilon_\tau. \tag{35}$$

Clean-latent estimation. For $A_\tau > 0$, rearranging Eq. 35 gives

$$z_1 = \frac{x_\tau - Q_\tau \epsilon_\tau}{A_\tau}. \tag{36}$$

The pretrained ε-predictor is evaluated on the variance-preserving projection of x_τ at the selected native noise index, producing a noise estimate $\hat{\epsilon}_L$. Substituting this estimate for the unknown effective noise yields

$$\hat{z}_1^L = \frac{x_\tau - Q_\tau \hat{\epsilon}_L}{A_\tau}. \tag{37}$$

To avoid division by a vanishing coefficient near $\tau = 0$, the implemented estimator uses

$$\boxed{\hat{z}_1^L = \frac{x_\tau - Q_\tau \hat{\epsilon}_L}{\max(A_\tau, \epsilon_{\mathrm{num}})}.} \tag{38}$$

The unclipped expression follows exactly from the reparameterized interpolant when the effective noise is known; the clipped expression is its numerically stabilized estimator.

Base velocity. The original interpolant has pathwise derivative

$$\dot{x}_\tau = \dot{A}_\tau z_1 + \dot{B}_\tau z_0 + \dot{G}_\tau \xi. \tag{39}$$

The effective noise ϵ_τ generally depends on τ when constructed from fixed draws of z_0 and ξ. Thus, differentiating the reparameterized expression while holding ϵ_τ fixed is not, in general, the pathwise derivative of the original sampled trajectory. Instead, the relevant conditional velocity is obtained by averaging Eq. 39 over the unresolved Gaussian components.

Because (z_0, ϵ_τ) and (ξ, ϵ_τ) are jointly Gaussian and zero-mean, with $\mathrm{Cov}(\epsilon_\tau, \epsilon_\tau) = I$, their conditional expectations are

$$\mathbb{E}[z_0 \mid \epsilon_\tau] = \mathrm{Cov}(z_0, \epsilon_\tau)\, \mathrm{Cov}(\epsilon_\tau, \epsilon_\tau)^{-1} \epsilon_\tau = \frac{B_\tau}{Q_\tau} \epsilon_\tau, \tag{40}$$

$$\mathbb{E}[\xi \mid \epsilon_\tau] = \mathrm{Cov}(\xi, \epsilon_\tau)\, \mathrm{Cov}(\epsilon_\tau, \epsilon_\tau)^{-1} \epsilon_\tau = \frac{G_\tau}{Q_\tau} \epsilon_\tau. \tag{41}$$

Since z_1 is independent of the Gaussian noise variables, conditioning additionally on z_1 does not change these expectations. Consequently,

$$\begin{aligned}\mathbb{E}[\dot{x}_\tau \mid z_1, \epsilon_\tau] &= \dot{A}_\tau z_1 + \dot{B}_\tau \mathbb{E}[z_0 \mid \epsilon_\tau] + \dot{G}_\tau \mathbb{E}[\xi \mid \epsilon_\tau] \\ &= \dot{A}_\tau z_1 + \frac{B_\tau \dot{B}_\tau + G_\tau \dot{G}_\tau}{Q_\tau} \epsilon_\tau. \end{aligned} \tag{42}$$

Differentiating $Q_\tau = \sqrt{B_\tau^2 + G_\tau^2}$ gives

$$\dot{Q}_\tau = \frac{B_\tau \dot{B}_\tau + G_\tau \dot{G}_\tau}{Q_\tau}. \tag{43}$$

Therefore,

$$\mathbb{E}[\dot{x}_\tau \mid z_1, \epsilon_\tau] = \dot{A}_\tau z_1 + \dot{Q}_\tau \epsilon_\tau. \tag{44}$$

For the chosen interpolation coefficient $A_\tau = \tau$, we have $\dot{A}_\tau = 1$, so the conditional velocity becomes

$$u_\tau^\star = z_1 + \dot{Q}_\tau \epsilon_\tau. \tag{45}$$

This is the flow-matching target used in Appendix B.1.

Replacing the unknown clean latent and effective noise with their predictor-based estimates gives the base velocity

$$\boxed{v_L = \hat{z}_1^L + \dot{Q}_\tau \hat{\epsilon}_L.} \tag{46}$$

Together, Eqs. 38 and 46 recover the two expressions in Eq. 9.

Scope of the derivation. The Gaussian reparameterization and conditional-velocity identities above are exact under the stated noise assumptions. The resulting predictor-based field is an estimate: the variance-preserving projection establishes compatibility with the pretrained predictor's noise family, while nearest-index lookup approximates its finite schedule. In particular, $\hat{\varepsilon}_L$ estimates the conditional noise rather than the realized ε_τ. Flow matching therefore trains the hierarchical correction against the realized target $u_\tau^\star$ to capture the conditionally predictable residual between the base field and target velocity.

D.3 Derivation and interpretation of the auxiliary latent estimate

The Gaussian-reparameterized interpolant is

$$x_\tau = A_\tau z_1 + Q_\tau \epsilon_\tau, \qquad A_\tau = \tau. \tag{47}$$

As derived above, conditioning the interpolant's pathwise derivative on (z_1, ϵ_τ) yields the conditional velocity target

$$u_\tau^\star = \dot{A}_\tau z_1 + \dot{Q}_\tau \epsilon_\tau = z_1 + \dot{Q}_\tau \epsilon_\tau, \tag{48}$$

where $\dot{A}_\tau = 1$. Accordingly, the base velocity and hierarchically corrected velocity are

$$v_L = \hat{z}_1^L + \dot{Q}_\tau \hat{\epsilon}_L, \qquad v_\Theta = v_L + g(\tau)\Delta v. \tag{49}$$

Auxiliary reconstruction objective. Flow matching supervises the corrected velocity, while an auxiliary reconstruction objective provides direct supervision of clean-latent accuracy through the same correction head. To construct the *training-time* auxiliary latent estimate, we assign the velocity correction to the clean-latent component while retaining the adapted noise estimate $\hat{\epsilon}_L$. This fixed-noise choice uses the existing ε-predictor without introducing an additional one. Inverting the velocity decomposition then gives

$$\begin{aligned}\hat{z}_1^\theta &:= \frac{v_\Theta - \dot{Q}_\tau \hat{\epsilon}_L}{\dot{A}_\tau} \\ &= v_L + g(\tau)\Delta v - \dot{Q}_\tau \hat{\epsilon}_L \\ &= \hat{z}_1^L + g(\tau)\Delta v, \end{aligned} \tag{50}$$

where the second line uses $\dot{A}_\tau = 1$.

The fixed-noise decomposition is a modeling choice, not a unique inversion of the corrected velocity: it does not additionally enforce $x_\tau = A_\tau \hat{z}_1^\theta + Q_\tau \hat{\epsilon}_L$. The resulting auxiliary objectives in latent and pixel space are

$$\mathcal{L}_{\text{rec}} = \mathbb{E}\left[\left\|\hat{z}_1^\theta - z_1\right\|_2^2\right], \tag{51}$$

and

$$\mathcal{L}_{\text{rec,pix}} = \mathbb{E}\left[\left\|D(\hat{z}_1^\theta) - D(z_1)\right\|_2^2\right], \tag{52}$$

respectively, where D is the frozen decoder. During training, these auxiliary objectives complement flow matching by providing direct latent- and pixel-space supervision.

We use $D(z_1)$ rather than Y as the pixel-space target so that the auxiliary objective remains anchored to the same clean latent endpoint z_1, without requiring the learned correction to compensate for reconstruction error of the frozen decoder. Moreover, letting $r = \hat{z}_1^\theta - z_1$, a first-order expansion gives

$$D(\hat{z}_1^\theta) - D(z_1) \approx J_D(z_1)r, \qquad \mathcal{L}_{\text{rec,pix}} \approx \mathbb{E}\big[r^\top J_D(z_1)^\top J_D(z_1) r\big]. \tag{53}$$

Thus, unlike latent-space MSE, which penalizes latent directions uniformly, the pixel-space objective locally weights latent discrepancies according to their sensitivity under the frozen decoder.

The auxiliary estimate $\hat{z}_1^\theta$ is used only to construct these objectives; inference directly integrates v_Θ.

D.4 Derivation of the stochastic sampler

Score estimation. As derived above, the interpolant admits the Gaussian reparameterization $x_\tau = A_\tau z_1 + Q_\tau \epsilon_\tau$, where $\epsilon_\tau \sim \mathcal{N}(0, I)$. The Gaussian-denoising score identity (Albergo et al., 2025) gives

$$\nabla_x \log p_\tau(x \mid S, C) = -\frac{\mathbb{E}[\epsilon_\tau \mid x_\tau = x, S, C]}{Q_\tau}. \tag{54}$$

Motivated by this identity, we construct an approximate score from the adapted noise prediction:

$$s_\Theta = -\frac{\hat{\epsilon}_L}{Q_\tau}. \tag{55}$$

Stochastic dynamics. The stochastic-interpolant framework (Albergo et al., 2025) shows that an exact probability-flow velocity and its corresponding score can define an SDE with the same time-dependent marginals, provided the initialization and regularity conditions hold. For FACET, we use the learned corrected velocity v_Θ and approximate score s_Θ in the following SDE:

$$\mathrm{d}x_\tau = (v_\Theta + \eta_\tau s_\Theta)\,\mathrm{d}\tau + \sqrt{2\eta_\tau}\,\mathrm{d}W_\tau, \tag{56}$$

where $\eta_\tau = \eta_0 \tau(1-\tau)$ and W_τ denotes standard Brownian motion.

Euler-Maruyama sampling. For $\tau_j = \tau_{\min} + j\Delta\tau$, the Brownian increment satisfies $\Delta W_j \overset{d}{=} \sqrt{\Delta\tau}\,\zeta_j$, where $\zeta_j \sim \mathcal{N}(0, I)$. Using Euler-Maruyama sampling (Albergo et al., 2025; Higham, 2001), we obtain the update

$$\boxed{x_{j+1} = x_j + \big(v_\Theta + \eta_{\tau_j} s_\Theta\big)\,\Delta\tau + \sqrt{2\eta_{\tau_j}\Delta\tau}\,\zeta_j.} \tag{57}$$

All fields are evaluated at (x_j, τ_j), and each update uses the corrected velocity v_Θ. Initialization and final emission follow Algorithm 2.

D.5 Fixed-scale Kullback–Leibler divergence

For two Gaussian distributions with shared isotropic covariance,

$$q_\phi = \mathcal{N}(\mu_q, \sigma_z^2 I), \qquad p_\psi = \mathcal{N}(\mu_p, \sigma_z^2 I), \tag{58}$$

let d_z denote the residual dimension. The KL divergence between two Gaussians is

$$D_{\mathrm{KL}}(q_\phi \| p_\psi) = \frac{1}{2}\Big[\operatorname{tr}\big(\Sigma_p^{-1}\Sigma_q\big) + (\mu_p - \mu_q)^\top \Sigma_p^{-1}(\mu_p - \mu_q) - d_z + \log \frac{\det \Sigma_p}{\det \Sigma_q}\Big]. \tag{59}$$

Here $\Sigma_q = \Sigma_p = \sigma_z^2 I$, so the log-determinant ratio vanishes and $\operatorname{tr}(\Sigma_p^{-1}\Sigma_q) = d_z$, which cancels the $-d_z$ term. Since $\Sigma_p^{-1} = \sigma_z^{-2} I$, the remaining quadratic term gives

$$D_{\mathrm{KL}}(q_\phi \| p_\psi) = \frac{1}{2\sigma_z^2}(\mu_p - \mu_q)^\top (\mu_p - \mu_q) = \frac{\|\mu_q - \mu_p\|_2^2}{2\sigma_z^2}, \tag{60}$$

as stated in Eq. 5.

D.6 Proof of Proposition 2

Because z_0 and ξ are independent standard Gaussian variables,

$$B_\tau z_0 + G_\tau \xi \sim \mathcal{N}\big(0, (B_\tau^2 + G_\tau^2) I\big)\,. \tag{61}$$

Thus, with $Q_\tau = \sqrt{B_\tau^2 + G_\tau^2}$,

$$\epsilon_\tau = \frac{B_\tau z_0 + G_\tau \xi}{Q_\tau} \sim \mathcal{N}(0, I), \qquad x_\tau = A_\tau z_1 + Q_\tau \epsilon_\tau. \tag{62}$$

Dividing by $N_\tau = \sqrt{A_\tau^2 + Q_\tau^2}$ gives

$$y_\tau = \frac{x_\tau}{N_\tau} = a_\tau z_1 + \sigma_\tau \epsilon_\tau, \qquad a_\tau = \frac{A_\tau}{N_\tau}, \quad \sigma_\tau = \frac{Q_\tau}{N_\tau}. \tag{63}$$

Therefore,

$$a_\tau^2 + \sigma_\tau^2 = \frac{A_\tau^2 + Q_\tau^2}{N_\tau^2} = 1, \tag{64}$$

which proves the proposition.

E Experimental Setup and Data Preparation

E.1 Human Protein Atlas benchmark

The HPA benchmark is derived from the Human Protein Atlas (Thul et al., 2017). Each example contains a protein amino-acid sequence, a target protein-fluorescence image, and three cellular-context channels corresponding to the nucleus, endoplasmic reticulum (ER), and microtubules (MT). Images are cropped to 1024×1024 pixels and resized to 256×256 pixels before the frozen variational autoencoder (VAE) maps them to $4 \times 64 \times 64$ latent representations.

The released HPA training set contains 87,763 stored images from 12,732 proteins and the test split contains 720 stored images from 100 proteins[1]. Following Zheng & Huang (2025), we use a maximum tokenized sequence length of 2048. This filtering removes 24 images from three proteins, yielding the 696-image, 97-protein evaluation set used throughout the paper. The longest retained sequence has 1938 residues, whereas the shortest excluded sequence has 2157 residues; therefore, the filtering boundary does not remove sequences close to the cutoff. During FACET module adaptation and semantic memory construct, we use a random subset of the HPA training split, formed by adding whole proteins in a fixed random order until an image budget is reached and keeping every image of each selected protein. This yields 24,001 images from 3,491 of the 12,732 training proteins (≈27% of training images); the pretrained backbone, in contrast, was trained on the full split. A protein-disjoint 10% of this pool (349 proteins, 2,467 images) forms a validation split that is excluded from every fit, is used only for hyperparameter tuning, and never overlaps the test proteins. Of the pool, 3,007 proteins carry a UniProt localization annotation; the 2,704 of them outside the validation split define

[1] https://virtualcellmodels.cziscience.com/dataset/hpa-subcellular-section

the prototypes. Unannotated proteins are used only for the context map, the ridge and the projections; prototype construction, prototype prior and posterior training, adapter pretraining, transport training, and negative sampling all restrict to annotated proteins.

For the HPA experiments, we evaluate four cellular-context regimes formed from different combinations of the three available channels: nucleus, nucleus+ER, nucleus+MT, and nucleus+ER+MT as done in prior work (Zheng & Huang, 2025).

E.2 OpenCell benchmark

The OpenCell benchmark is derived from endogenous-tagging microscopy (Cho et al., 2022). The released OpenCell training split contains 5,828 images from 1,211 proteins and the test split contains 473 stored examples from 100 proteins[2]. Applying the same maximum tokenized sequence length of 2048 removes 19 examples, leaving 454 images from 97 proteins.

The OpenCell training split used for OpenCell adaptation contains 5,659 images from 1,179 proteins after applying the same sequence-length filter. From these 10% samples are held-out as validation set, containing 117 proteins and 544 corresponding images. OpenCell test proteins are also strictly disjoint from the HPA training split used to fit the semantic memory.

For OpenCell, we use the nucleus image as the cellular-context input. We evaluate HPA-module transfer of the HPA-trained FACET modules and further OpenCell adaptation of those modules on the OpenCell training split.

E.3 Backbone-matched instantiation

Choice of backbone. We instantiate FACET on CELL-Diff because, to our knowledge, it is the only publicly available sequence-conditioned latent diffusion model for this task with released checkpoints for both HPA and OpenCell. Other recent methods, such as CELL-E2 (Khwaja et al., 2023) and Vermeer (Kambhampati et al., 2026) are autoregressive; PUPS (Zhang et al., 2025) is a deterministic single-cell inpainting model, for which distributional metrics and sampling-based calibration are not defined; and ProtiCelli (Sun et al., 2026) conditions on protein identities from a fixed vocabulary, rather than protein sequences, and therefore cannot be evaluated on proteins held out from training.

Instantiation. For both HPA and OpenCell, the pretrained ε-predictor, VAE, and ESM-2 protein language model (Lin et al., 2023) are taken from the corresponding dataset-specific CELL-Diff checkpoint (Zheng & Huang, 2025). The CELL-Diff baseline uses the same checkpoint without modification. Sharing these components ensures that differences between CELL-Diff, FACET, and its internal variants reflect the conditioning design rather than differences in backbone capacity or pretraining. This initialization is a choice of experimental design, not part of FACET itself, which applies to any pretrained variance-preserving ε-predictor (Proposition 2).

All original backbone weights remain frozen. In variants with backbone LoRA, trainable rank-16 matrices are inserted into 90 image-attention projections. The ε-adapter and prototype prior p_η are pretrained in Stage I and then frozen, while the LoRA parameters, hierarchical correction head, and continuous prior p_ψ are optimized during transport training (Appendix B.1). The posterior modules q_ω and q_ϕ are used only during training and are not required for generation. In total, the deployed modules add 9.83M trainable parameters (0.588% of the 1.67B-parameter instantiation; Table 11).

The LoRA rank, number of adapted projections, module depths, and optimization and sampling constants specify the evaluated implementation; they are not claimed to be optimal choices.

E.4 Hardware specifications

All experiments were conducted on a compute node with eight CPU cores from an AMD EPYC 9354 processor running at 3.25 GHz, 64 GB of system memory, and a single NVIDIA L40S GPU with 48 GB of GDDR6 memory.

[2] https://virtualcellmodels.cziscience.com/dataset/opencell-microscopy-images

F METRIC DEFINITIONS AND EVALUATION PROTOCOLS

F.1 IMAGE-QUALITY METRICS

We report mutual information (MI), multiscale structural similarity (MS-SSIM), Haralick texture agreement, corpus-level peak signal-to-noise ratio (PSNR), continuous intensity overlap ($\mathrm{IoU}_{\mathrm{cont}}$), and Fréchet inception distance (FID).

Mutual information. MI measures statistical dependence between target and generated image intensities. For target image intensities X and generated image intensities $\widehat{X}$, it is defined as

$$I(X;\widehat{X}) = \sum_{x,\widehat{x}} p(x,\widehat{x}) \log \frac{p(x,\widehat{x})}{p(x)p(\widehat{x})}. \tag{65}$$

Multiscale SSIM. Following the multiscale SSIM formulation of Wang et al. (2003) and the fluorescence-image evaluation protocol of Sun et al. (2026), we compute five-scale MS-SSIM on images normalized to $[0, 1]$, using an 11×11 uniform window and 2×2 average pooling between scales. At scales 1-4, the SSIM map is spatially averaged before combining the scales. The same implementation is used for all methods before protein-level macro aggregation.

Haralick texture agreement. We quantify texture similarity using six Haralick descriptors (contrast, dissimilarity, homogeneity, energy, correlation, and angular second moment) computed from gray-level co-occurrence matrices at a unit pixel offset and four orientations, following Haralick et al. (1973). We average each descriptor over orientations and define texture agreement as the mean of the corresponding per-feature relative similarities between real and generated images (Sun et al., 2026).

Peak signal-to-noise ratio. For corpus-level mean squared error $\mathrm{MSE}_{\mathrm{corpus}}$ and maximum image intensity L, PSNR is

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{L^2}{\mathrm{MSE}_{\mathrm{corpus}}} \right). \tag{66}$$

All normalized image intensities use $L = 1$.

Continuous intensity overlap. For nonnegative target and generated fluorescence-intensity maps Y and $\widehat{Y}$, we compute the threshold-free weighted Jaccard similarity, also known as the generalized Jaccard coefficient or Ruzicka similarity:

$$\mathrm{IoU}_{\mathrm{cont}}(Y,\widehat{Y}) = \frac{\sum_u \min\{Y_u, \widehat{Y}_u\}}{\sum_u \max\{Y_u, \widehat{Y}_u\}}, \tag{67}$$

where u indexes spatial locations. We compute this score for each target-generated pair, average scores within each protein, and then macro-average across proteins.

Table 11: Parameter counts for the trainable FACET modules. Deployed modules are used during generation; posterior modules are used only during training.

Module	Parameters	Deployed
ε-adapter	1,318,533	✓
Prototype prior p_η	923,145	✓
Hierarchical correction head	4,918,340	✓
LoRA (rank 16)	2,406,400	✓
Continuous prior p_ψ	262,785	✓
Total deployed	**9,829,203**	–
Continuous posterior q_ϕ	164,481	–
Prototype posterior q_ω	3,020,297	–

Fréchet Inception Distance. We measure distributional similarity using the Fréchet Inception Distance (FID) (Heusel et al., 2017). Each evaluation panel is represented as a three-channel image whose red channel is zero, green channel is the protein-fluorescence image, and blue channel is the corresponding nucleus image (Zheng & Huang, 2025; Khwaja et al., 2023). We extract 2048-dimensional pool3 features from Inception-v3 and compute FID using the same panel construction and feature-extraction pipeline for every method. Let (μ_r, Σ_r) and (μ_g, Σ_g) denote the empirical means and covariances of the real and generated feature embeddings. Then

$$\mathrm{FID} = \|\mu_r - \mu_g\|_2^2 + \mathrm{Tr}\Big(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}\Big). \tag{68}$$

Lower values indicate closer agreement between the real and generated image distributions.

Network function evaluations. NFE denotes the number of learned-model evaluations required during sampling and measures sampling cost. The primary FACET operating point uses 25 NFE, while the ablations consider operating points down to 10 NFE. The backbone-matched baseline uses 100 NFE.

F.2 Protein-association evaluation

For the protein-association analysis, we follow the protocol introduced by Sun et al. (2026). We use STRING (Szklarczyk et al., 2023) solely to obtain external evaluation labels. For each protein pair, we compute within-cell Pearson similarity after removing the per-cell mean image across proteins. This removes similarity caused by shared cellular morphology.

For each system and STRING threshold, we compare STRING-associated pairs with localization-matched non-associated pairs when both proteins share a primary compartment; otherwise, we use all non-associated pairs as the unmatched null. We report

$$\Delta = \mathbb{E}\left[r(a,b) \mid \text{associated}\right] - \mathbb{E}\left[r(a,b) \mid \text{matched null}\right], \tag{69}$$

where $r(a,b)$ is the within-cell Pearson similarity. Positive values indicate stronger association-consistent spatial patterns.

Statistical significance tests compare associated pairs with their corresponding null pairs and are assessed separately within each system using a two-sided Mann-Whitney U test (Mann & Whitney, 1947).

F.3 Predictive calibration metrics

We evaluate the expected calibration error (ECE), maximum calibration error (MCE), and the Brier score (Guo et al., 2017; Brier, 1950). For each input condition, stochastic generation produces an empirical foreground probability p_u at spatial location u. Let $\mathcal{B}_1, \ldots, \mathcal{B}_M$ denote fixed probability bins. Expected calibration error (ECE) is

$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{|\mathcal{B}_m|}{N} \left|\mathrm{acc}(\mathcal{B}_m) - \mathrm{conf}(\mathcal{B}_m)\right|, \tag{70}$$

where $\mathrm{acc}(\mathcal{B}_m)$ is the empirical foreground frequency and $\mathrm{conf}(\mathcal{B}_m)$ is the mean predicted probability in bin m.

Maximum calibration error (MCE) is

$$\mathrm{MCE} = \max_m \left|\mathrm{acc}(\mathcal{B}_m) - \mathrm{conf}(\mathcal{B}_m)\right|. \tag{71}$$

The Brier score is

$$\mathrm{Brier} = \frac{1}{N}\sum_{u=1}^{N}(p_u - y_u)^2, \tag{72}$$

where y_u is the binary foreground label. Lower ECE and MCE indicate smaller calibration gaps, whereas lower Brier score indicates better overall probabilistic forecast quality.

G STATISTICAL AND SAMPLING PROTOCOL

All methods are evaluated on identical held-out examples, protein sequences, cellular-context inputs, and target images for each comparison. Unless otherwise specified, the reported image metrics use the same preprocessing, feature extraction, aggregation rules, and checkpoint across methods.

G.1 PREDICTIVE CALIBRATION PROTOCOL

The calibration experiment uses a strictly held-out fitting partition rather than test-set fitting or protein-wise cross-validation. We generate a second predictive grid on a held-out training-validation set containing 96 images from 81 proteins. This partition shares zero proteins, image identifiers, or storage keys with the 696-image, 97-protein HPA test benchmark.

Both partitions use the identical checkpoint, sampler, 25-NFE operating point, stochasticity scale, adapter and hierarchical-head settings, and forecast reduction procedure.

For each FACET image, four residual-latent draws are crossed with four stochastic-differential-equation noise seeds, producing 16 predictive samples. The backbone-matched reference uses the same total predictive-sample budget. Before test scoring, we determine the foreground threshold from the 349 held-out validation proteins as

$$t = \mathrm{median}_{p\in\mathcal{V}} \left[\frac{1}{|\mathcal{I}_p|} \sum_{i\in\mathcal{I}_p} Q_{0.99}(\{v_{ij} : j \in B_i\}) \right], \tag{73}$$

where $\mathcal{V}$ is the set of held-out validation proteins, $\mathcal{I}_p$ is the set of validation images for protein p, v_{ij} is the intensity at pixel j in image i, and B_i is the valid image region excluding the dilated cell mask. This gives $t = 6.111$ on the stored 0–255 uint8 scale ($t/255 \approx 0.02397$). The threshold is then held fixed for test evaluation. For HPA calibration, each sample is binarized using this threshold, and the empirical foreground probability is defined from the 16 binarized samples:

$$p_u = \frac{1}{16} \sum_{s=1}^{16} \mathbf{1}[x_s(u) > t].$$

Here $x_s(u)$ is the intensity at location u in sample s. Calibration uses 15 equal-width bins on $[0, 1]$, with the final bin including the upper endpoint; the Brier score is computed directly from the pixel-level probabilities and labels (Guo et al., 2017).

For post-hoc calibration, we follow the temperature-scaling procedure introduced by Guo et al. (2017). Note that the threshold in Eq. 73 is computed over all 349 held-out validation proteins, whereas the temperature is fit on the 96-image, 81-protein subset of those proteins for which the 16-sample predictive grid was generated. Because raw sample proportions can be exactly 0 or 1, we first use Jeffreys smoothing (Jeffreys, 1946; Brown et al., 2001) $p_a = (k + \alpha)/(16 + 2\alpha)$ with k the number of foreground samples and $\alpha = 0.5$, transform to logit space, and apply $p' = \sigma(\mathrm{logit}(p_a)/T)$, with probabilities clipped to $[10^{-12}, 1 - 10^{-12}]$ when evaluating the negative log-likelihood. One scalar temperature is fit for each model by validation negative log-likelihood and then applied unchanged to that model's test forecasts. No calibration parameter is estimated from test pixels or test proteins; Table 9 reports the post-hoc calibrated values.

G.2 SAMPLING EFFICIENCY

The HPA operating point uses 75% fewer network function evaluations than the backbone-matched diffusion reference (Zheng & Huang, 2025). On matched hardware (Appendix E.4), the measured generation loop for the 696-image HPA evaluation set takes approximately 15 minutes for FACET and 78 minutes for the reference, corresponding to an approximately 80.8% reduction in wall-clock time.

Because context loading and other system-level operations may introduce small run-to-run variations, NFE is used as the primary compute measure. The wall-clock measurements are reported as a complementary hardware-specific observation.

H Additional Analyses

H.1 Conditioning sensitivity

We estimate the local sensitivity of the learned correction to z_S and z_C using Hutchinson trace estimators (Hutchinson, 1989) of $\mathrm{tr}(J_{z_a} J_{z_a}^{\mathsf{T}}) = \|J_{z_a}\|_F^2$, where $J_{z_a} = \partial \Delta v / \partial z_a$ is the Jacobian of the learned velocity correction with respect to z_a, for $a \in \{S, C\}$. We evaluate 20 images from 18 proteins and five timesteps.

In the implemented latent parameterization, the median raw ratio

$$\frac{\|J_{z_S}\|_F}{\|J_{z_C}\|_F}$$

is 28.0, with 10th and 90th percentiles of 13.0 and 48.3, respectively. Because this ratio uses unnormalized Jacobian norms and z_S and z_C have different latent scales, we treat it as a descriptive diagnostic rather than a scale-independent comparison. To account for the observed scale of each latent, we additionally report the normalized sensitivities

$$\rho_{z_a} = \frac{\|J_{z_a}\|_F \|z_a\|_2}{\|\Delta v\|_F}, \qquad a \in \{S, C\},$$

where Δv is the learned velocity correction. The median normalized sensitivities are 3.645 for z_S and 0.606 for z_C, with 10th and 90th percentiles of $[1.236, 9.638]$ and $[0.431, 1.157]$, respectively. Thus, under the observed latent scales, the median normalized sensitivity to z_S is approximately sixfold that to z_C.

This diagnostic measures local operating sensitivity of the learned correction. It does not establish a causal benefit of the FiLM ordering because no reversed-order model was trained. Instead, it provides supporting evidence that the protein and cellular-context pathways have distinct and measurable effects within the implemented hierarchical conditioning mechanism.

H.2 Sampling and schedule representability

For the reported uniform Euler-Maruyama implementation, integration uses the fixed range $\tau \in [0.05, 1]$. The lower cutoff is an implementation choice associated with the inherited finite native schedule and numerical endpoint handling; it is not a universal FACET constant.

With n evaluations on the uniform grid $\tau_j = \tau_{\min} + j\Delta\tau$, $j = 0, \ldots, n-1$, the final (lowest-noise) evaluation occurs at

$$\tau_{\mathrm{emit}} = \tau_{\max} - \frac{\tau_{\max} - \tau_{\min}}{n} = 1 - \frac{0.95}{n}. \tag{74}$$

Each native noise level k corresponds to the unique coordinate τ_k satisfying $a_{\tau_k} = \sqrt{\bar{\alpha}_k}$, since a_τ is strictly increasing. The least-noisy level of the pretrained predictor's 200-level cosine training schedule ($\bar{\alpha} \approx 0.99975$) maps to $\tau \approx 0.98588$. We call an operating point supported if every evaluation lies within the range spanned by the native schedule, so that nearest-index lookup never falls back to the boundary level. Requiring $\tau_{\mathrm{emit}} \leq 0.98588$ gives $n \leq 67.3$, i.e., $\mathrm{NFE} \leq 67$: at NFE 67, $\tau_{\mathrm{emit}} = 0.985821$, whereas at NFE 68, $\tau_{\mathrm{emit}} = 0.986029$ falls outside this range. The lower cutoff $\tau_{\min} = 0.05$ lies well within the native range.

I Additional Qualitative Results

The following figures complement the qualitative comparison presented in the main paper. They use the same visualization conventions and method names as the main results.

I.1 Cellular-context ablation

Figure 5 visualizes the cellular-context ablation. The protein sequence is held fixed while the combination of cellular-context channels is altered.

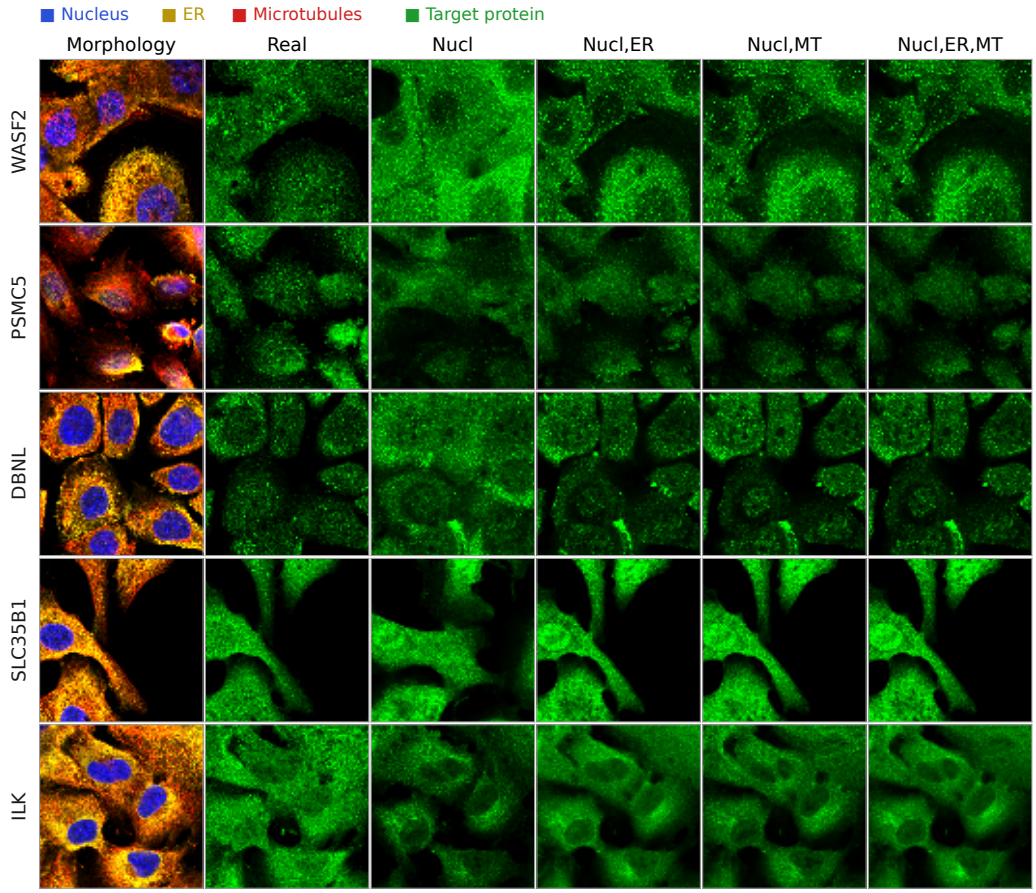

Figure 5: Cellular-context ablation. The protein sequence is held fixed while the cellular-context input is removed or modified. The examples illustrate context-conditioned changes in the generated spatial phenotype; they are qualitative evidence rather than a causal ablation beyond the quantitative comparison reported in the main text.

I.2 Same-protein generation across different contexts

Figure 6 provides additional examples in which the same protein sequence is evaluated under distinct cellular contexts.

I.3 Additional protein and context combinations on HPA

Figure 7 shows a broader grid of different proteins and cellular contexts on HPA. This figure provides broader qualitative coverage beyond the representative examples shown in the main paper.

I.4 Additional qualitative comparison on OpenCell

Figure 8 presents additional qualitative comparisons on OpenCell.

I.5 Additional qualitative comparison on HPA

Figure 9 provides additional HPA qualitative comparisons beyond the main-text qualitative figure. It uses the same method ordering and visualization conventions as the main comparison.

J DISCLOSURE OF GENERATIVE AI USE

Large Language Models (LLMs) were **used only to polish the text**; they **were not involved** in retrieval, discovery, or research ideation.

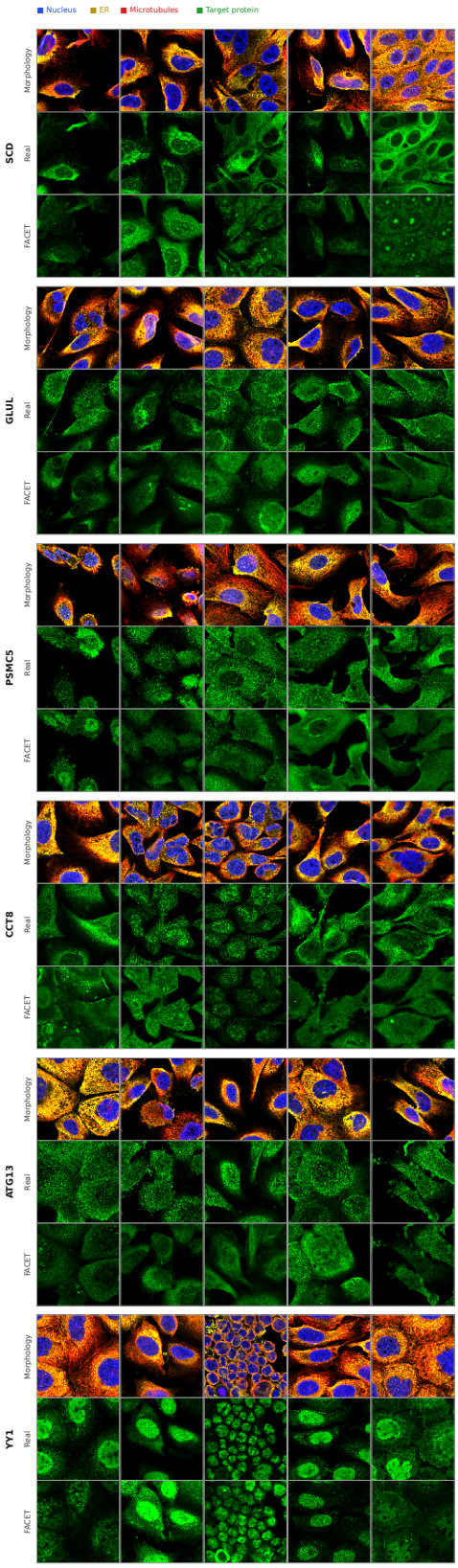

Figure 6: Same-protein generation across HPA cellular contexts. The sequence input is fixed across contexts, allowing context-conditioned changes in the generated spatial phenotype to be visualized.

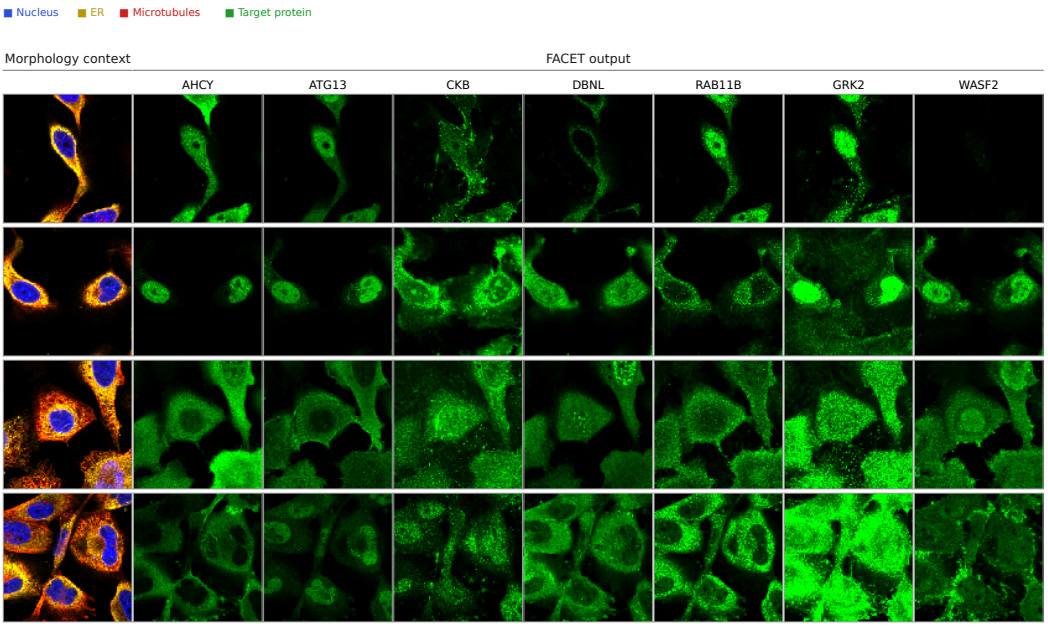

Figure 7: Additional HPA generations across multiple proteins and cellular contexts. The examples provide broader coverage of the conditional-generation setting beyond the main-text qualitative comparison.

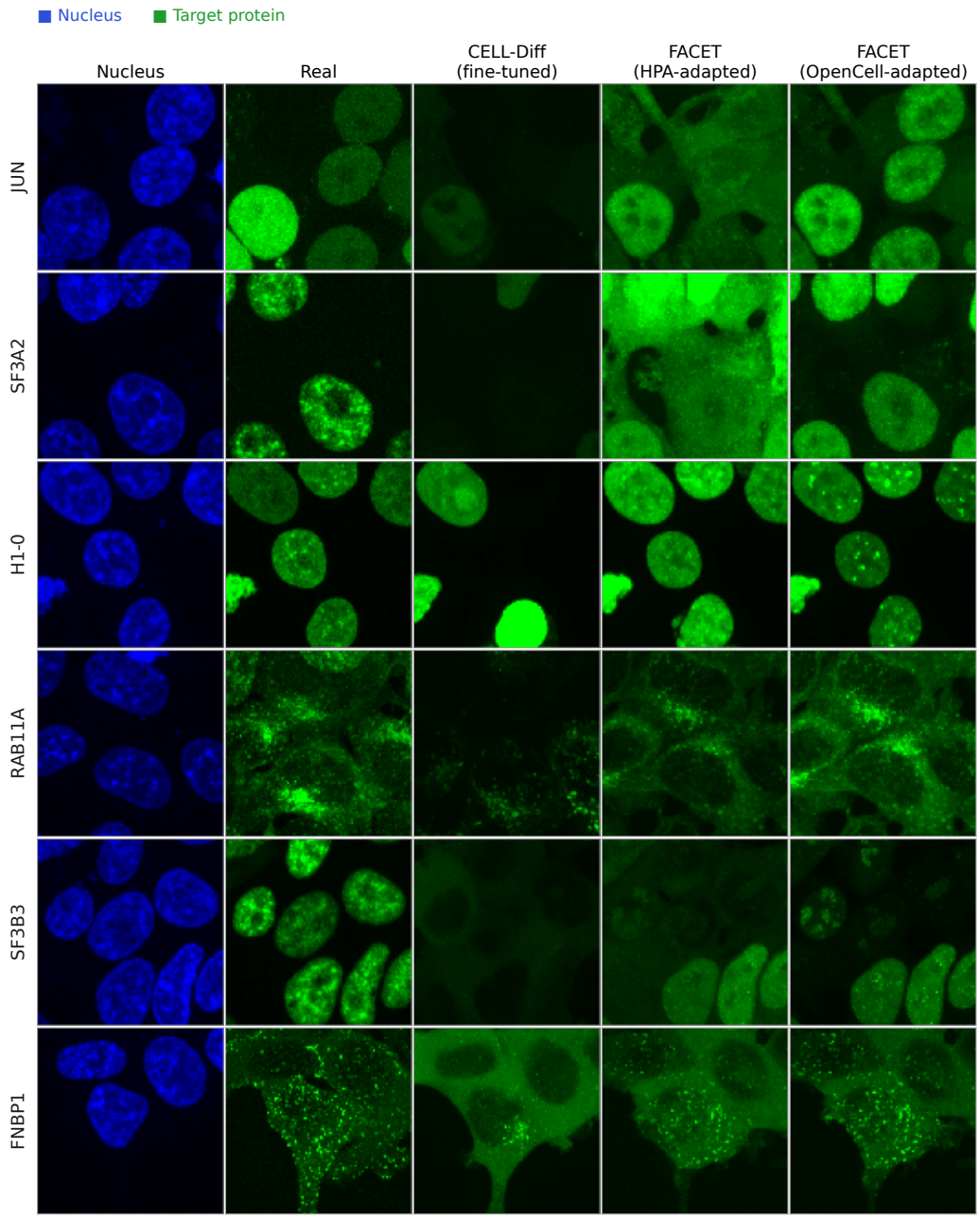

Figure 8: Additional qualitative results on OpenCell. The figure uses the same method names and visualization conventions as the main OpenCell results and complements the quantitative comparison. Here "CELL-Diff (fine-tuned)" represents the CELL-Diff checkpoint fine-tuned on OpenCell data, the same one reported in the main text and the source (Zheng & Huang, 2025).

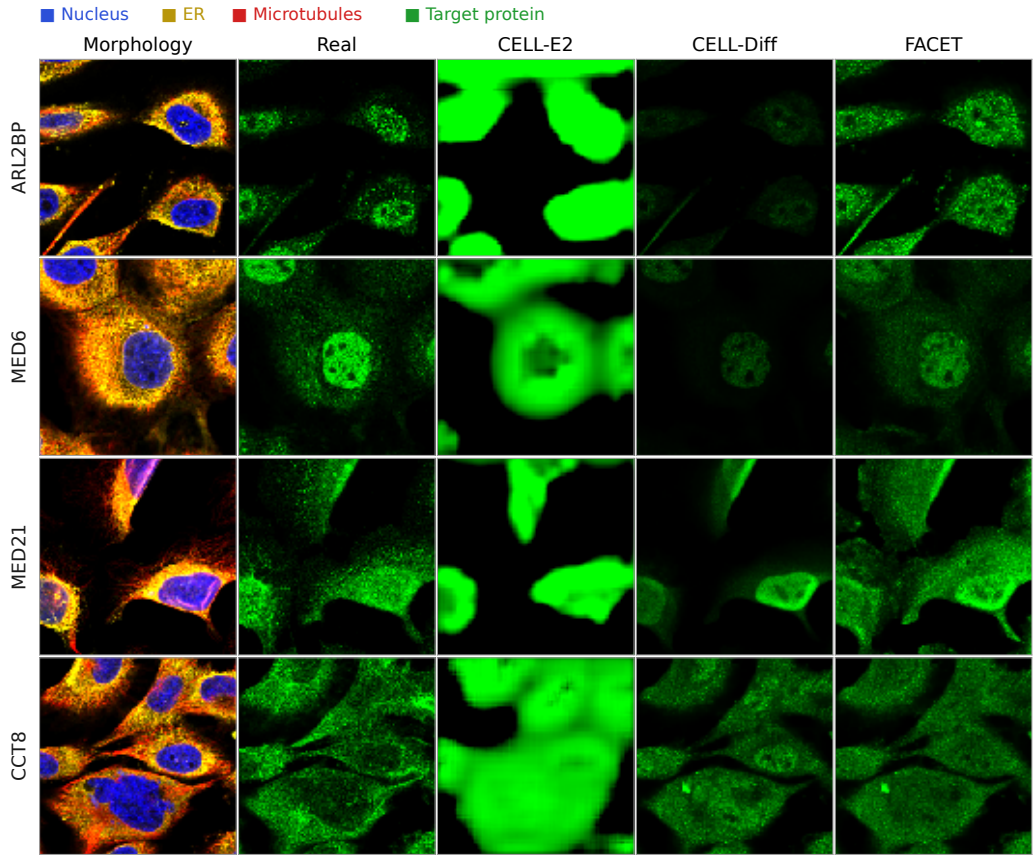

Figure 9: Additional qualitative results on HPA. CELL-E2 uses nucleus-only context, whereas CELL-Diff and FACET use the full three-channel context. These examples extend the main-text qualitative comparison using the same evaluation setting, method names, and visualization conventions.