arXiv:2611.02671v1 [cs.CV] 2 Nov 2026

TME-ContextNet: Class-Conditioned Spatial Supervision and Grouped Decoding for Joint Nuclei Instance Segmentation and Classification

Rukang Ma[1,7] Bingxue Wang[1] Liwen Zou[2] Yingying Cao[3] Zhenghua Cai[4]
Yu Li[1] Dong Wang[5,*] Huiyu Zhou[6,*] Luying Gui[1,*]

[1]School of Mathematics and Statistics, Nanjing University of Science and Technology, Nanjing, 210094, China
[2]Department of Mathematics, Nanjing University, Nanjing, 210093, China
[3]Department of Radiology, Affiliated Hospital of Nanjing University of Chinese Medicine, Nanjing, 210018, China
[4]Department of Pancreatic and Bariatric Surgery, Nanjing Drum Tower Hospital, the Affiliated Hospital of Nanjing University Medical School, Nanjing, 210008, China
[5]School of Artificial Intelligence and Information Technology, Nanjing University of Chinese Medicine, Nanjing, 210023, China
[6]School of Computing and Mathematical Sciences, University of Leicester, Leicester, LE1 7RH, United Kingdom
[7]School of Computer Science and Engineering, South China University of Technology, Guangzhou, 510006, China

[*]Corresponding authors:
Dong Wang: dongwang@seu.edu.cn Huiyu Zhou: hz143@leicester.ac.uk
Luying Gui: ly.gui@njust.edu.cn

Abstract

Joint nuclei instance segmentation and classification requires precise boundaries and neighborhood information that helps distinguish morphologically similar cell types. We present TME-ContextNet, a multi-task framework that learns class-conditioned spatial context from existing nuclear annotations and organizes decoding into geometry-related and semantic task groups. Class-conditioned spatial supervision extends nuclear labels into a dense target using Euclidean distance transforms, requiring no additional tissue-region annotation. We derive its nearest-region interpretation and characterize its stability to small annotation perturbations. A local–global context module combines dilated convolution and self-attention, while a grouped task decoder reduces redundant computation. The spatial head supplies training supervision and is omitted during inference. Experiments on PanMix, CoNSeP, and PanNuke yield Dice scores of 80.23%, 65.32%, and 79.13%, and mean multi-class panoptic quality (mPQ+) scores of 67.62%, 35.50%, and 43.74%, respectively. PanNuke results are averaged across three test folds. Target and decoder controls support the combination of distance-based supervision and grouped decoding while identifying their dependence on task configuration. The framework uses 36.40 million parameters, compared with 54.74 million for HoVer-Net. These findings support a compact, context-aware formulation for joint nuclei analysis in computational pathology.

Keywords: Computational pathology, nuclei instance segmentation, nuclei classification, spatial supervision, multi-task learning, efficient decoding

1 Introduction

Quantitative analysis of hematoxylin and eosin (H&E) images requires identifying individual nuclei and assigning them biologically meaningful types. The resulting cellular objects support measurements of tissue composition, nuclear morphology, and spatial organization. These measurements are useful because tissue interpretation depends on both cellular appearance and the

arrangement of different cell populations. For example, the spatial relationship between lymphocytes and tumor compartments provides information that a total nuclear count alone cannot express. Joint nuclei instance segmentation and classification is therefore a fundamental step in computational pathology and quantitative imaging [1–3].

The joint task presents a tension between local precision and contextual recognition. Separating touching nuclei requires detailed boundaries and reliable instance geometry. Recognizing nuclear types can require a larger field of view because nuclei with similar appearance may occupy different cellular neighborhoods. Staining variation, heterogeneous morphology, and unequal class frequencies make these requirements harder to reconcile. A representation that improves semantic recognition must still retain the spatial detail needed to recover individual nuclei.

Current nuclei-analysis systems address these requirements through geometric prediction, contextual feature extraction, or interactions among multiple tasks. HoVer-Net combines foreground, displacement, and nuclear type prediction [4]; subsequent systems refine encoders, decoders, and whole-slide processing [5, 6]. These advances leave a specific opportunity in the supervision itself. Nuclear type annotations directly identify cell populations inside nuclear regions, while much of the surrounding non-nuclear space remains undifferentiated background. A larger receptive field exposes that space to the model but does not specify how existing cell annotations should organize its representation. At the same time, sharing all late decoder features can constrain tasks with different spatial requirements, whereas independent task decoders repeat substantial computation.

We investigate whether spatial supervision derived from nuclear classes can complement image context within a compact joint model. TME-ContextNet extends existing nuclear labels into the surrounding space through Class-conditioned Spatial Supervision (CSS). The implemented target uses Euclidean distance transforms (EDT) to associate non-nuclear locations with the nearest annotated class region, requiring no additional tissue-region annotation. A Local–Global Context Module (LGCM) combines dilated convolution with self-attention. A Grouped Task Decoder (GTD) shares intermediate features and then separates geometry-related outputs from semantic outputs. The CSS head shapes learning during training and is omitted from the inference outputs.

The contributions concern both the supervision and its integration. We formulate CSS as a class-conditioned spatial partition and derive the conditions under which its assignments are stable to small annotation perturbations. We introduce grouped decoding that couples CSS to nuclear type prediction while retaining a separate refinement path for nuclear geometry. We evaluate the complete framework on PanMix, CoNSeP, and PanNuke and examine target construction, component interactions, sharing depth, and computational cost. The complete model achieves Dice scores of 80.23%, 65.32%, and 79.13%, respectively, and the highest mPQ+ among the evaluated specialist methods on each dataset. The controls characterize the contributions of spatial supervision, contextual features, and decoder sharing to the joint task.

2 Related Work

2.1 Geometric representations and joint nuclear analysis

Encoder–decoder architectures established the use of multi-resolution features for biomedical segmentation. U-Net combines contextual encoding with localization through skip connections [7], and Micro-Net uses multiple image resolutions for microscopy segmentation [8]. Instance-oriented approaches add explicit object structure. StarDist predicts star-convex polygons [9], while Cellpose reconstructs instances from spatial flows [10]. These representations address object separation, which is necessary before reliable cell-level quantities can be measured.

Joint nuclear analysis adds the requirement that each separated object receive a type. HoVer-Net predicts foreground, horizontal–vertical displacement, and nuclear type in a com-

mon framework [4]. SONNET uses self-guided ordinal regression to stratify nuclear regions [11], and PointNu-Net connects semantic keypoints to dynamically generated instance masks [12]. SMILE studies cost-sensitive multi-task learning and task interaction under class imbalance [13]. Together, these approaches establish that nuclear geometry and semantic identity can benefit from coordinated learning, while leaving several choices for the outputs and their supervision.

2.2 Context and annotation-derived spatial supervision

Contextual methods exploit neighboring cells and tissue appearance to interpret nuclear morphology [1, 14, 15]. Context and attention also recur across the recent literature on nuclei instance segmentation and classification [16]. Transformer-based models provide another mechanism for relating local structures to a wider field of view. CellViT combines a vision-transformer encoder with task-specific decoding for joint segmentation and classification [17], while CellViT++ adapts foundation-model representations to changing cell taxonomies through separate segmentation and classification stages [18]. CFR-SAM introduces local feature adaptation and fine-grained boundary refinement for nuclei analysis [19].

Auxiliary targets provide a complementary way to organize learning. CAS-UNet uses boundary-distance supervision for nuclear-region segmentation [20]. Its distance target describes nuclear geometry, whereas CSS uses class-specific nuclear supports to assign the surrounding non-nuclear space to cell populations. The distinction concerns what the auxiliary output represents: an instance- or boundary-related geometric quantity and a class-conditioned neighborhood partition impose different constraints on a shared representation. CSS requires no additional region annotations: it uses existing nuclear labels to construct class-conditioned spatial supervision. Its geometric target makes cellular neighborhood organization available to representation learning.

2.3 Decoder sharing and computational efficiency

Recent specialist methods also improve the architecture and processing cost of joint analysis. Zhang et al. combine enhanced encoders, nested dense decoding, regularization, and ensemble strategies [5]. HoVer-NeXt emphasizes efficient nuclei analysis within a whole-slide pipeline [6], and NuLite develops lightweight joint models [21]. These studies make computational cost and the organization of decoding important comparison dimensions. GTD studies a particular division of the late representation: foreground and displacement form a geometry group, while nuclear type and CSS form a semantic group. The corresponding controls compare this grouping with independent and plain shared decoding.

General-purpose segmentation architectures provide a different reference for nuclear class-map prediction. PSPNet aggregates context through pyramid pooling [22]; K-Net uses iteratively updated kernels for segmentation [23]; Mask2Former applies masked attention to segmentation [24]; and SegFormer combines a hierarchical transformer encoder with lightweight decoding [25]. They are included as representative segmentation architectures with different ways of using context. Their class maps test whether the proposed framework improves pixel-level nuclear type assignments beyond established general segmentation designs. Specialist nuclei-analysis methods provide the complementary comparison for correctly delineated and typed instances. Keeping these roles explicit makes the pixel and instance results answer distinct questions.

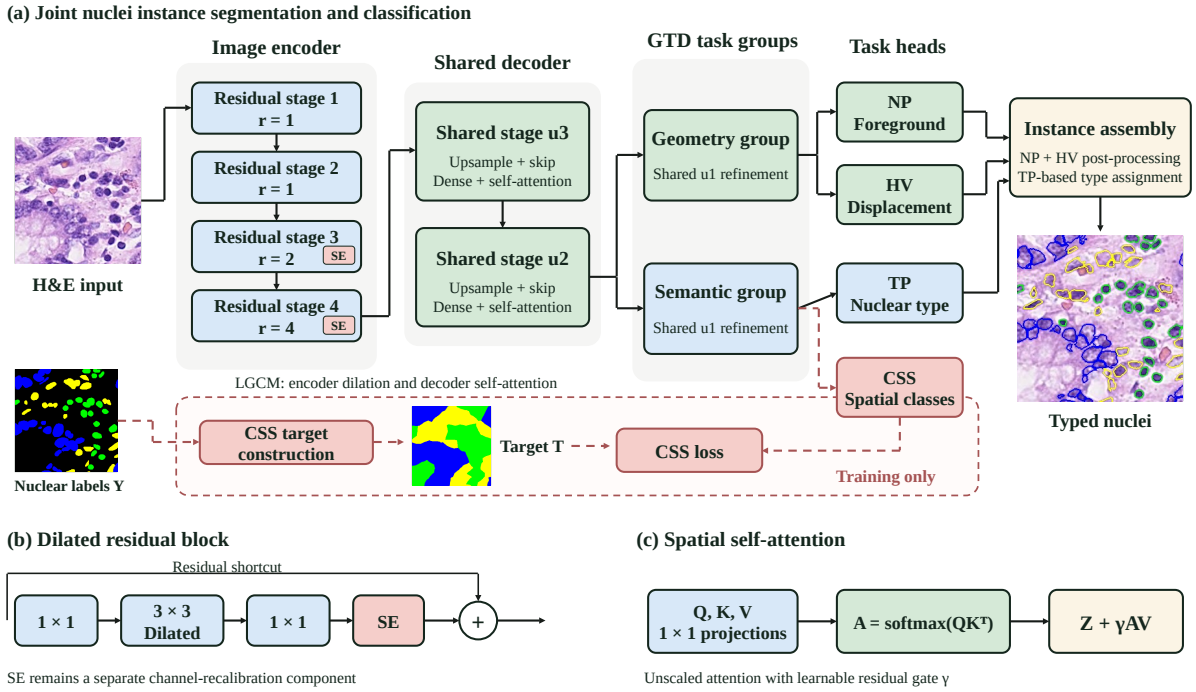


Fig. 1 Overview of TME-ContextNet. (a) The image encoder and shared decoder feed geometry-related and semantic task groups. Nuclear foreground (NP), horizontal–vertical displacement (HV), and nuclear type (TP) predictions are used for typed instance assembly. Class-conditioned spatial supervision (CSS) provides an annotation-derived auxiliary training target, and its head is omitted at inference. Encoder dilation and decoder self-attention form the local–global context module (LGCM), while squeeze-and-excitation (SE) supplies separate channel recalibration. (b) Dilated residual block with a shortcut connection. (c) Spatial self-attention with a learnable residual gate. GTD denotes the grouped task decoder

3 Materials and Methods

3.1 Task and framework

Let $X \in \mathbb{R}^{H\times W\times 3}$ be an H&E image on a pixel domain Ω. The annotations consist of an instance map I and a nuclear type map $Y \in \{0,1,\dots,C\}^{H\times W}$, where zero denotes background. The required outputs are an instance map $\widehat{I}$ and one type label for each predicted nucleus. During training, the network additionally predicts nuclear foreground probabilities $\widehat{N}$, horizontal–vertical displacement maps $\widehat{V}$, nuclear type probabilities $\widehat{P}$, and an auxiliary spatial target $\widehat{T}$.

An encoder produces multi-resolution features $E(X)$. Shared upsampling stages combine these features before the decoder separates them into two groups: a geometry group for foreground and displacement, and a semantic group for nuclear type and CSS. Figure 1 summarizes the computation. The CSS target is generated from the training annotations. At test time, the network takes the image alone; foreground and displacement predictions produce instances through HoVer-style post-processing, and type predictions assign a label to each instance [4].

3.2 Class-conditioned spatial supervision (CSS)

For every class c present in an annotated image, define its nuclear support and distance transform by

$$F_c = \{p \in \Omega : Y(p) = c\}, \qquad d_c(p) = \min_{q\in F_c} \|p - q\|_2. \tag{1}$$

Thus, distance is measured to the class-specific nuclear region rather than to a nuclear centroid. Let $\mathcal{C}_X = \{c : F_c \neq \varnothing\}$. The implemented influence score is

$$S_c(p) = \max\left\{\exp\left[-\frac{d_c(p)^2}{2\sigma^2}\right], \varepsilon\right\}, \quad c \in \mathcal{C}_X, \tag{2}$$

with a common scale $\sigma = 5$ pixels and numerical floor $\varepsilon = 10^{-6}$. The target preserves the annotated nuclear labels and extends the class assignments into background:

$$T(p) = \begin{cases} Y(p), & Y(p) > 0, \\ \arg\max\limits_{c \in \mathcal{C}_X} S_c(p), & Y(p) = 0,\ \mathcal{C}_X \neq \varnothing, \\ 0, & \mathcal{C}_X = \varnothing. \end{cases} \tag{3}$$

Ties are resolved by the first class in the ordered list of present classes, as in the generation code. Figure 2 illustrates the construction.

Geometric interpretation. When the largest influence is unique and above the numerical floor, the common exponential in Eq. (2) is strictly decreasing in distance. Consequently,

$$\arg\max_c S_c(p) = \arg\min_c d_c(p). \tag{4}$$

The EDT target is therefore a class-conditioned nearest-region partition of the non-nuclear space. It gives adjacent cells a shared contextual region when they belong to the same class while maintaining a transition between neighborhoods of different classes. Nuclear appearance supplies the image evidence, and the spatial partition supplies an additional organization of the supervision.

The common σ changes the continuous scores but, away from numerical-floor ties, does not change the hard class assignments. The main construction consequently does not require a set of class-specific influence strengths. The auxiliary signal expresses spatial proximity to annotated cell populations. Its biological motivation is the relevance of cellular organization to tissue interpretation [2]; its mathematical content is a geometric representation derived from those annotations.

The construction also has a useful stability property. If a class support changes from F_c to F_c' by at most δ in Hausdorff distance, then $|d_c(p) - d_c'(p)| \leq \delta$. A winning nearest class therefore remains unchanged wherever its distance margin over the runner-up exceeds 2δ, provided that numerical clipping does not produce a tie. Appendix A gives the derivation. This property characterizes the stability of the spatial target under small annotation perturbations.

3.3 Local–global context module (LGCM)

LGCM combines dilated residual feature extraction in the encoder with self-attention inside decoder dense blocks. Residual learning and atrous convolution provide the basis for its convolutional context extraction [26, 27], while dense connections support feature reuse within the decoder [28]. Together, these operations provide local and non-local image context. SE remains a separate channel-recalibration component. For a residual unit with input Z_{k-1}, write

$$Z_k = \mathcal{A}_k(\mathcal{G}_k(Z_{k-1}; r_k)) + \mathcal{P}_k(Z_{k-1}), \tag{5}$$

where $\mathcal{G}_k$ is a bottleneck convolutional transformation with dilation r_k, $\mathcal{P}_k$ aligns shortcut channels and resolution, and $\mathcal{A}_k$ denotes the spatial alignment used before addition. Dilation increases the spatial sampling extent while retaining the convolutional kernel's parameter count.

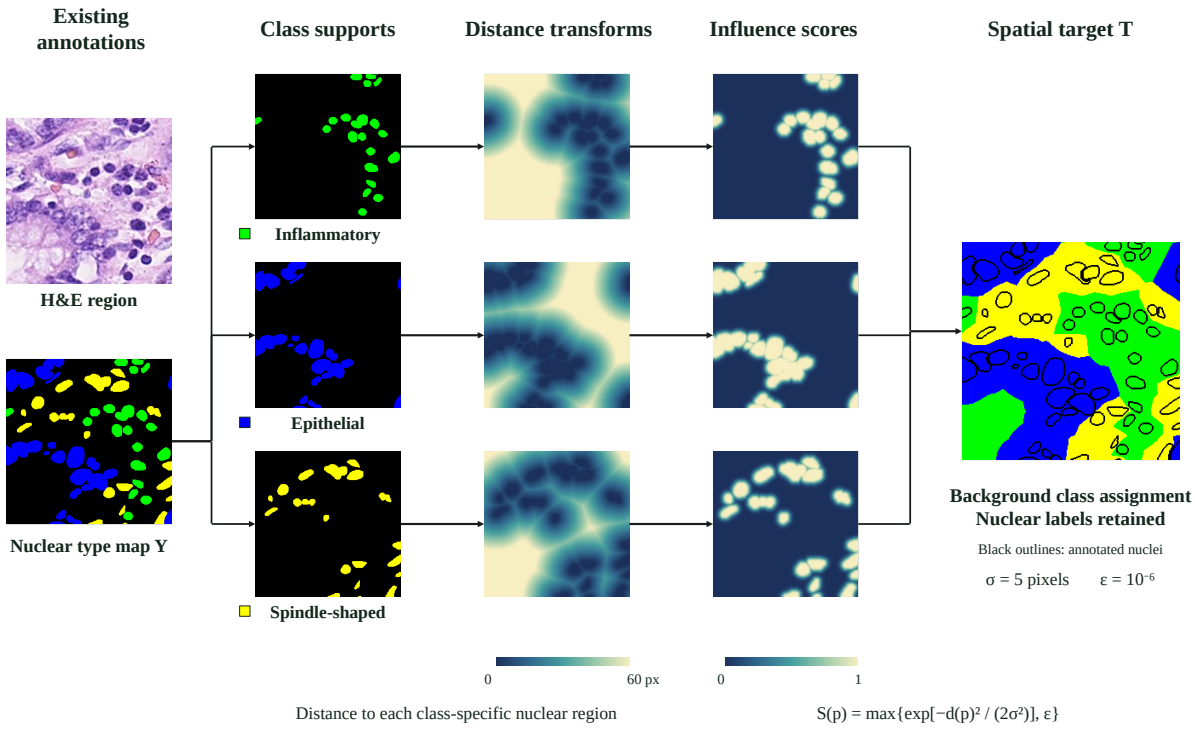


Fig. 2 Construction of the EDT-based CSS target. The nuclear type map is separated into class-specific supports. Each support produces a Euclidean distance transform and an influence score with $\sigma = 5$ pixels and $\varepsilon = 10^{-6}$. Background pixels receive the class with the largest score, while annotated nuclei retain their original labels. Black outlines identify nuclear regions on the final target. Distance and score maps use continuous color scales, and categorical maps use the corresponding nuclear class colors

With kernel width k_l, dilation r_l, stride s_l, receptive-field width R_l, and sampling jump j_l, the standard recurrence is

$$R_l = R_{l-1} + (k_l - 1) r_l j_{l-1}, \qquad j_l = s_l j_{l-1}. \tag{6}$$

This relationship describes theoretical convolutional support. Self-attention then enables feature exchange across the spatial locations represented at the attention stage.

For a flattened feature matrix $Z \in \mathbb{R}^{n \times d}$, let $Q = ZW_Q$, $K = ZW_K$, and $V = ZW_V$. Spatial aggregation follows the non-local attention formulation [29]. The implemented attention and residual output are

$$A = \mathrm{softmax}_{\mathrm{row}}(QK^\top), \qquad Z' = Z + \gamma AV. \tag{7}$$

The learnable gate γ is initialized to zero, so the block initially follows its local feature path and learns the contribution of non-local aggregation during optimization. Equation (7) follows the unscaled dot-product attention in the implementation. Squeeze-and-excitation (SE) applies channel recalibration [30] and is controlled separately from LGCM in the experiments. This separation allows the contribution of dilation and spatial attention to be examined without conflating it with channel gating.

3.4 Grouped task decoder (GTD)

GTD uses a shared decoder representation followed by group-specific transformations and separate prediction heads. Let $Z = D_{\mathrm{sh}}(E(X))$ be the shared representation. Its outputs are

$$(\widehat{N}, \widehat{V}) = (h_N(G_{\mathrm{geom}}(Z)), h_V(G_{\mathrm{geom}}(Z))), \tag{8}$$

$$(\widehat{P}, \widehat{T}) = (h_P(G_{\mathrm{sem}}(Z)), h_T(G_{\mathrm{sem}}(Z))). \tag{9}$$

The grouping reflects the outputs' roles: foreground and displacement concern nuclear geometry, whereas type and CSS assign semantic classes at different spatial supports. The decoder shares intermediate features rather than treating every task as either fully independent or identical throughout decoding.

The parameter saving can be expressed without assuming a particular channel width. Suppose m independent task decoders each contain a trunk with P_D parameters and a head with P_{h_t} parameters. For an encoder with P_E parameters,

$$P_{\text{independent}} = P_E + mP_D + \sum_t P_{h_t}, \tag{10}$$

$$P_{\text{grouped}} = P_E + P_{\text{sh}} + \sum_g P_{G_g} + \sum_t P_{h_t}, \tag{11}$$

$$\Delta P = mP_D - P_{\text{sh}} - \sum_g P_{G_g}. \tag{12}$$

Sharing is parameter-efficient when $\Delta P > 0$. The accuracy associated with a particular sharing arrangement depends on the tasks and supervision, which motivates the decoder controls in Table 5. Appendix Fig. 8 illustrates the decoder structures and sharing stages.

During inference, h_T is omitted. In Eq. (9), the retained outputs do not use its predicted value, so omitting the auxiliary head preserves their forward computation for fixed shared weights. CSS can thus alter the representation learned during training without adding a required prediction stage to the deployed nuclei-analysis output.

3.5 Training objective and auxiliary gradient

Foreground and type prediction use cross-entropy and soft Dice losses following HoVer-Net [4]. For one-hot targets y_{uc} and class probabilities π_{uc} over supervised pixels $u \in \Omega_s$, define

$$\begin{aligned} \mathcal{L}_{\text{CE}}(y, \pi) &= -\frac{1}{|\Omega_s|} \sum_{u \in \Omega_s} \sum_c y_{uc} \log \pi_{uc}, \\ \mathcal{L}_{\text{Dice}}(y, \pi) &= \sum_c \left[1 - \frac{2 \sum_u y_{uc} \pi_{uc} + \epsilon_D}{\sum_u y_{uc} + \sum_u \pi_{uc} + \epsilon_D} \right]. \end{aligned} \tag{13}$$

The classification branches use softmax cross-entropy with Dice, consistent with mutually exclusive type targets. Foreground prediction uses the corresponding binary two-class form. Displacement supervision combines mean squared error with a foreground-focused gradient error. The total objective is

$$\mathcal{L} = \mathcal{L}_N + \mathcal{L}_V + \mathcal{L}_P + \omega \mathcal{L}_T, \qquad \mathcal{L}_T = \mathcal{L}_{\text{CE}}(T, \widehat{T}) + \mathcal{L}_{\text{Dice}}(T, \widehat{T}), \tag{14}$$

where branch coefficients may equivalently be absorbed into the individual loss terms and ω controls auxiliary supervision. For the controlled CoNSeP target experiments, both auxiliary loss terms have coefficient 0.5 and both primary classification loss terms have coefficient one; the no-CSS condition sets the auxiliary coefficient to zero. The CSS head supplies gradients to the shared representation even though its output is omitted at inference.

The choice and weighting of auxiliary objectives matter in multi-task learning: uncertainty-based weighting addresses relative loss scales [31], while gradient surgery addresses interference between task updates [32]. Here, the focus is the spatial target supplied to the shared representation. To make this role explicit, let $g = \nabla_\theta \mathcal{L}_{\text{main}}$ and $a = \nabla_\theta \mathcal{L}_T$ for shared parameters θ, where $\mathcal{L}_{\text{main}} = \mathcal{L}_N + \mathcal{L}_V + \mathcal{L}_P$. For fixed ω, assume that the primary loss has a locally Lipschitz gradient near the current shared parameters. Under a small gradient step $\theta' = \theta - \eta(g + \omega a)$, a first-order expansion gives

$$\mathcal{L}_{\text{main}}(\theta') - \mathcal{L}_{\text{main}}(\theta) = -\eta \|g\|^2 - \eta\omega \langle g, a \rangle + O(\eta^2). \tag{15}$$

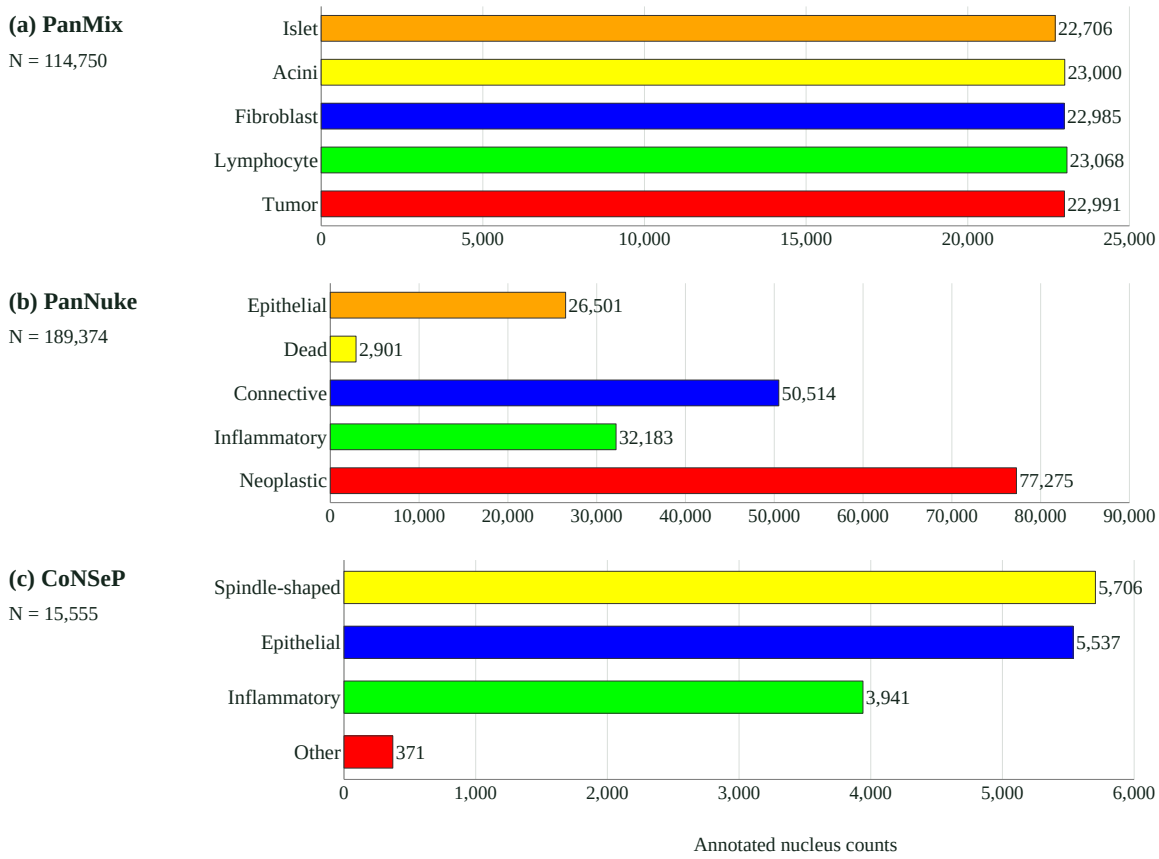


Fig. 3 Nuclear class composition of the complete PanMix, PanNuke, and CoNSeP datasets. Bars show annotated nucleus counts from the dataset statistics. Colors follow the class order used for each dataset in the qualitative figures. Axis scales differ among panels

The auxiliary contribution $-\eta\omega\langle g, a\rangle$ favors the immediate primary-loss decrease when $\omega > 0$ and the gradients align. A negative inner product reduces that decrease, although the primary term can still dominate. The remainder is of order η^2 for a fixed update direction; Appendix A gives a finite-step bound. Equation (15) motivates testing alternative spatial targets rather than assuming that any dense target is beneficial. The target controls evaluate this design choice through observed performance; gradient alignment itself was not measured in those experiments.

3.6 Datasets and experimental protocol

The study uses PanMix, CoNSeP, and PanNuke. PanMix is a private pancreatic histopathology collection with annotations for tumor, lymphocyte, fibroblast, acinar, and islet nuclear categories. Access is organized through an application procedure described on the associated GitHub page. CoNSeP provides colorectal H&E images with nuclear instance annotations; the analysis uses the four-category grouping associated with joint nuclei analysis [4]. PanNuke contains diverse tissue types and five nuclear classes: neoplastic, inflammatory, connective, dead, and epithelial [33, 34].

Figure 3 summarizes the annotated nuclear class counts over each entire dataset. PanMix has similar counts across its five categories, whereas PanNuke contains a much smaller dead-cell population than its neoplastic and connective populations. The grouped CoNSeP labels also have unequal counts. This composition provides context for interpreting overall overlap and the class-specific results. The dataset-specific taxonomies are retained throughout evaluation.

All models were trained without pretrained weights for 50 epochs, and the best checkpoint was selected using the validation set. PanNuke was evaluated with three fold rotations, each

providing an independent test fold. PanMix and CoNSeP use the established training, validation, and test partitions of the experiments. Test data were reserved for final evaluation. For TME-ContextNet, the implementation uses Adam with an initial learning rate of 10^{-4}. The controlled CoNSeP experiments use batch size four, Adam coefficients $(0.9, 0.999)$, and a tenfold learning-rate reduction every 25 epochs. Their validation criterion is the sum of the foreground, displacement, and type losses, excluding the training-only auxiliary loss. Input crops are 270×270 pixels with an 80×80 supervised central region. Image preprocessing and augmentation follow the established experimental pipeline.

The comparison has two complementary scopes. PSPNet, K-Net, Mask2Former, and SegFormer [22–25] evaluate the pixel classification component using their predicted nuclear class maps. Their selection spans pyramid context, kernel-based segmentation, masked attention, and hierarchical transformer encoding. HoVer-Net, Zhang et al., HoVer-NeXt, and CellViT [4–6, 17] additionally evaluate joint typed-instance prediction. CellViT adds a vision-transformer specialist to the convolutional and ConvNeXt-based comparators. The Zhang implementation uses a SEResNeXt-50 encoder, and the HoVer-NeXt implementation uses ConvNeXtV2-Tiny. All reported comparisons use the trained implementations under the 50-epoch, random-initialization protocol described above. CellViT uses a trainable ViT-S encoder with embedding dimension 384, 12 transformer blocks, six attention heads, and feature extraction at blocks 3, 6, 9, and 12. It retains the nuclear foreground, displacement, and type branches and their original loss coefficients. Training uses 256×256 patches, batch size four, FP32, and AdamW with an initial learning rate of 3×10^{-4}, weight decay 10^{-4}, and cosine decay. Its tissue-classification auxiliary loss is disabled because the prepared training manifests supply nuclear labels only; patches are sampled uniformly. The validation nuclear-branch loss selects the checkpoint, and native HoVer-style post-processing produces typed instances.

All methods contribute Dice and IoU where a nuclear class map is available. mPQ+ additionally requires nuclear instance identities and is reported for the evaluated specialist outputs. Dashes mark outputs without nuclear instance identities. Appendix Fig. 10 supplies qualitative class-map comparisons for the general segmentation methods, while Fig. 4 focuses on specialist joint analysis.

The controls serve separate questions. PanMix component experiments examine the configurations in Table 3. To isolate the construction of CSS, the CoNSeP target experiments use a dedicated HoVer-Net-based control model with an auxiliary spatial prediction head. They compare EDT with no auxiliary target, Gaussian constructions, class-weight controls, and a class-agnostic distance target. This target-design experiment is reported separately from the complete TME-ContextNet comparison in Table 1. The no-CSS and EDT conditions each use seeds 10, 20, and 30; the remaining target conditions use seed 10. The auxiliary head architecture is retained across these target controls, and its loss is disabled in the no-CSS condition. Appendix B specifies each target. Decoder and LGCM comparisons are organized into the matched comparison blocks in Table 5, retaining their respective experimental settings.

The archived auxiliary-weight sensitivity experiment uses PanMix and the EDT target, varying ω from 0.0 to 1.0 in increments of 0.1. Each weight uses one training run with the remaining settings held fixed; each boxplot summarizes scores across test images. This experiment addresses the strength of spatial supervision, whereas the CoNSeP controls address its construction. The Dice and IoU boxplots describe the per-image response to auxiliary-loss strength.

For efficiency measurements, network latency denotes the timed forward computation, whereas image latency includes image assembly and post-processing and excludes disk input/output. The benchmark uses 32-bit floating-point precision (FP32) with batch size one. Image and P95 latency refer to a 256×256 image; GFLOPs refer to one native network input patch. All latency and memory measurements in Table 6 use RTX 4090. The main-model and decoder blocks identify separate measurement sessions, with models within each block evaluated sequentially on the same GPU.

3.7 Evaluation and reporting

Dice and intersection over union (IoU) follow the pixel-level evaluator used for the result collection. For image i, let G_{ic} and P_{ic} be the ground-truth and predicted regions of class c. The evaluator computes

$$\mathrm{Dice}_{ic} = \frac{2|G_{ic} \cap P_{ic}| + \epsilon_m}{|G_{ic}| + |P_{ic}| + \epsilon_m}, \tag{16}$$

$$\mathrm{IoU}_{ic} = \frac{|G_{ic} \cap P_{ic}| + \epsilon_m}{|G_{ic} \cup P_{ic}| + \epsilon_m}, \qquad \epsilon_m = 10^{-5}. \tag{17}$$

The overall pixel score is the unweighted mean across images and classes $c = 0, \ldots, C$, including background. With this smoothing, an empty ground-truth and predicted class mask scores one. The reported Dice and IoU therefore measure class-map overlap.

Mean multi-class panoptic quality (mPQ+) evaluates type-aware instances, building on the panoptic quality formulation [35] and its use in joint nuclei analysis [4]. For foreground instances of class c, matches require the same class and IoU strictly greater than 0.5. With matched pairs TP_c and unmatched predictions and annotations FP_c and FN_c, respectively,

$$\mathrm{DQ}_c = \frac{|TP_c|}{|TP_c| + \frac{1}{2}|FP_c| + \frac{1}{2}|FN_c|}, \tag{18}$$

$$\mathrm{SQ}_c = \frac{\sum_{(g,p)\in TP_c} \mathrm{IoU}(g,p)}{|TP_c|}, \tag{19}$$

$$\mathrm{PQ}_c^{+} = \frac{\sum_{(g,p)\in TP_c} \mathrm{IoU}(g,p)}{|TP_c| + \frac{1}{2}|FP_c| + \frac{1}{2}|FN_c|} = \mathrm{DQ}_c\,\mathrm{SQ}_c, \tag{20}$$

$$\mathrm{mPQ}^{+} = \frac{1}{|\mathcal{C}_{\mathrm{eval}}|} \sum_{c\in\mathcal{C}_{\mathrm{eval}}} \mathrm{PQ}_c^{+}\,. \tag{21}$$

Here, DQ denotes detection quality and SQ denotes segmentation quality. The class-wise matching statistics are accumulated across the evaluation set before calculating each PQ_c^{+}. The set $\mathcal{C}_{\mathrm{eval}}$ contains foreground classes with a nonzero PQ denominator; undefined empty/empty classes are omitted. SQ is set to zero when there are no matched instances. This metric evaluates type-aware detection and instance delineation together. Table 2 additionally reports the product of smoothed pixel precision and recall, defined for each image by

$$\mathrm{PR}_{ic} = \frac{(|G_{ic} \cap P_{ic}| + \epsilon_m)^2}{(|P_{ic}| + \epsilon_m)(|G_{ic}| + \epsilon_m)}. \tag{22}$$

The product is averaged across images and labeled Pixel $P \times R$ in the class-wise table. Instance-level quality is measured by mPQ+ in Eq. (21).

PanNuke values are reported as the mean and sample standard deviation across its three test folds. For the repeated CSS conditions, Table 4 reports mean and sample standard deviation across three random seeds; the remaining target conditions are single-run results. For measurements $z_1, \ldots, z_n$, the mean and sample standard deviation are $\bar{z} = n^{-1}\sum_i z_i$ and $s = [\sum_i (z_i - \bar{z})^2/(n-1)]^{1/2}$, respectively. Single-run results are shown without an estimated standard deviation. Comparisons are descriptive.

4 Results

4.1 Joint analysis across three datasets

Table 1 reports the complete comparison. TME-ContextNet achieves the highest Dice, IoU, and mPQ+ values on all three datasets. On PanMix, its Dice and mPQ+ are 80.23% and 67.62%,

respectively. On CoNSeP, the corresponding values are 65.32% and 35.50%. On PanNuke, it obtains Dice of $79.13 \pm 0.80\%$, IoU of $74.85 \pm 0.81\%$, and mPQ+ of $43.74 \pm 2.89\%$ across the three test folds.

The pixel-level comparison is strongest when measured against the best general segmentation result for each dataset. TME-ContextNet exceeds K-Net by 1.06 Dice points on PanMix and 0.67 points on PanNuke, and exceeds Mask2Former by 1.16 points on CoNSeP. These consistent overlap differences extend the comparison to established general segmentation architectures. The general segmentation models remain competitive, particularly on the overlap measures for which they produce directly comparable class maps.

The specialist baselines provide the instance-level comparison. HoVer-Net has the strongest competing mPQ+ on PanMix, Zhang et al. on CoNSeP, and CellViT on PanNuke. The corresponding margins for TME-ContextNet are 19.77, 1.10, and 1.83 percentage points. CellViT achieves $76.33 \pm 0.38\%$ Dice, $72.04 \pm 0.36\%$ IoU, and $41.91 \pm 0.30\%$ mPQ+ on PanNuke. TME-ContextNet exceeds these means by 2.80, 2.81, and 1.83 points, respectively. CellViT also obtains Dice/mPQ+ of 73.51%/41.09% on PanMix and 57.76%/29.74% on CoNSeP, providing a transformer-based reference for the complete joint task.

The smaller CoNSeP margin and larger PanMix margin show that the size of the advantage depends on the dataset and metric. Pixel overlap and mPQ+ should therefore be read together: a plausible class region can still contain a merged nucleus, a missed object, or an incorrect instance type.

Figure 4 compares all five specialist methods at identical image coordinates. The PanMix field combines a dense lymphocyte region with elongated fibroblast nuclei, exposing differences in both object separation and class assignment. In the CoNSeP field, the larger epithelial nuclei reveal fragmentation in several predictions, including CellViT. In the PanNuke field, CellViT and the other baselines assign several annotated epithelial nuclei to the neoplastic class, while TME-ContextNet preserves their epithelial assignments. These examples connect the joint instance metric to visible differences in contours and types. The supplementary class-map comparison in Appendix Fig. 10 addresses the general segmentation baselines separately.

4.2 Class-specific behavior on PanNuke

Table 2 shows the five nuclear categories. TME-ContextNet has the highest Dice and IoU among the five listed methods for neoplastic, inflammatory, connective, and epithelial nuclei. Its neoplastic Dice is 78.18%, compared with 75.49% for CellViT and 67.67% for HoVer-Net; its connective Dice is 56.13%, compared with 52.03% and 48.92%, respectively. These class-specific results show that the overall gain is accompanied by improvements across several cell populations.

The dead-class result for Zhang et al. is an empty-prediction case. Across all three test folds, the evaluated implementation predicts no dead-class instances: both TP and FP are zero, and all 2,825 annotated dead nuclei are missed. Its dead-class instance PQ is consequently zero. The apparently high pixel Dice, IoU, and Pixel $P \times R$ values of $97.07 \pm 0.57\%$ arise because the evaluator assigns one to an empty/empty class mask. Images without dead-cell annotations account for approximately 97% of each fold on average and dominate these image-averaged scores. The marked row in Table 2 therefore records class absence, rather than successful recognition of dead nuclei. This case explains why the pixel measures are complemented by type-aware mPQ+ in Table 1.

Appendix Fig. 9 provides additional PanNuke examples with enlarged details, and Appendix Fig. 11 illustrates a local disagreement in neoplastic and dead-cell assignments.

Table 1 Comparison on PanMix, CoNSeP, and PanNuke. Values are percentages; bold indicates the highest value in each dataset and metric.

PanMix			
Method	Dice (%)	IoU (%)	mPQ+ (%)
PSPNet	75.62	73.04	—
KNet	79.17	76.01	—
Mask2Former	77.77	75.67	—
Segformer	76.18	73.49	—
HoVer-Net	69.80	67.22	47.85
Zhang et al. (2024)	71.27	68.85	42.56
HoVer-NeXt	69.85	67.18	29.16
CellViT	73.51	70.85	41.09
TME-ContextNet	**80.23**	**78.52**	**67.62**
CoNSeP			
Method	Dice (%)	IoU (%)	mPQ+ (%)
PSPNet	54.66	46.04	—
KNet	55.91	48.13	—
Mask2Former	64.16	55.33	—
Segformer	60.79	51.67	—
HoVer-Net	49.94	41.49	23.35
Zhang et al. (2024)	59.02	51.24	34.40
HoVer-NeXt	45.94	37.51	20.51
CellViT	57.76	49.07	29.74
TME-ContextNet	**65.32**	**56.81**	**35.50**
PanNuke			
Method	Dice (%)	IoU (%)	mPQ+ (%)
PSPNet	75.03 ± 1.11	71.33 ± 1.06	—
KNet	78.46 ± 0.36	74.46 ± 0.27	—
Mask2Former	66.67 ± 2.37	62.82 ± 2.88	—
Segformer	69.11 ± 6.87	65.32 ± 7.10	—
HoVer-Net	72.52 ± 5.45	68.14 ± 5.48	35.92 ± 0.78
Zhang et al. (2024)	73.95 ± 1.46	69.89 ± 1.50	36.06 ± 1.45
HoVer-NeXt	69.61 ± 0.57	65.23 ± 0.58	33.46 ± 0.39
CellViT	76.33 ± 0.38	72.04 ± 0.36	41.91 ± 0.30
TME-ContextNet	**79.13 ± 0.80**	**74.85 ± 0.81**	**43.74 ± 2.89**

PanNuke: mean ± sample standard deviation across three test folds. Dice and IoU follow the original pixel evaluator, including background. mPQ+ uses foreground instances with same-class matching at IoU > 0.5. Dashes indicate that instance outputs are unavailable.

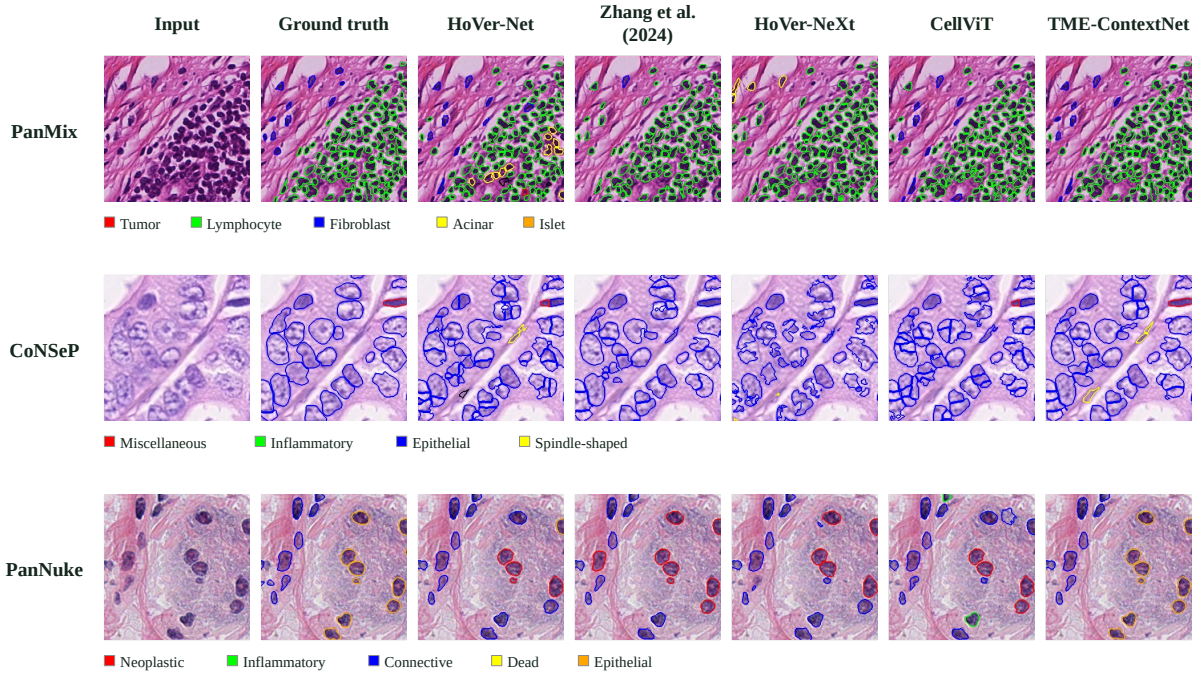


Fig. 4 Qualitative examples of joint nuclei instance segmentation and classification on PanMix, CoNSeP, and PanNuke. Columns show the H&E input, ground truth, HoVer-Net, Zhang et al. (2024), HoVer-NeXt, CellViT, and TME-ContextNet. Identical image regions and class colors are used across methods within each dataset. Colored contours indicate individual nuclei and their assigned types, with dataset-specific class legends below each row

4.3 Component configurations on PanMix

The PanMix configurations are reported in Table 3. Adding CSS to the HoVer-Net configuration raises Dice from 69.80% to 73.83% and mPQ+ from 47.85% to 52.48%. The complete configuration has the highest values, reaching Dice of 80.23%, IoU of 78.52%, and mPQ+ of 67.62%.

The intermediate variants clarify the roles of the architectural components. The configuration without self-attention has Dice of 77.45% and mPQ+ of 56.19%; removing both LGCM and SE gives 75.69% and 54.03%. The configuration without SE has Dice of 70.50% and mPQ+ of 32.54%, compared with 80.23% and 67.62% for the complete model. The larger difference in the joint instance score indicates that a change in feature recalibration can affect the combined detection, delineation, and type-assignment output more strongly than the pixel average. Because SE is controlled independently, its removal is reported explicitly rather than being described as a pure LGCM removal. The configuration containing CSS and attention on HoVer-Net also performs below the CSS-only configuration, indicating that attention's benefit depends on the feature-sharing arrangement in which it is used.

4.4 Spatial target controls on a HoVer-Net-based model

Table 4 compares CSS target constructions in the dedicated HoVer-Net-based control model. The experiment examines the auxiliary target while holding the control architecture fixed. EDT gives the highest mean Dice and IoU among the tested constructions, reaching 53.60% and 45.50%, respectively. Compared with the no-CSS condition (51.91% Dice and 43.60% IoU), these values represent improvements of 1.69 and 1.90 percentage points. They support the use of the class-conditioned distance target within this control experiment.

Uniform Gaussian, class-weighted Gaussian, constant class weights, and permuted class weights all yield lower Dice and IoU than the no-CSS condition. Permuted class weights yield

Table 2 Per-class comparison on PanNuke. Bold indicates the highest mean within each class and metric.

Class	Method	Dice (%)	IoU (%)	Pixel $P \times R$ (%)
Neoplastic	HoVer-Net	67.67 ± 4.26	61.41 ± 4.33	60.30 ± 4.31
	Zhang et al. (2024)	63.75 ± 5.42	57.39 ± 5.28	56.23 ± 5.31
	HoVer-NeXt	55.04 ± 0.43	48.46 ± 0.84	47.12 ± 0.90
	CellViT	75.49 ± 0.51	69.43 ± 0.12	68.33 ± 0.09
	TME-ContextNet	**78.18 ± 1.37**	**72.22 ± 1.06**	**71.18 ± 1.03**
Inflammatory	HoVer-Net	58.31 ± 2.14	53.13 ± 2.29	52.02 ± 2.36
	Zhang et al. (2024)	57.75 ± 1.47	53.02 ± 1.40	52.05 ± 1.34
	HoVer-NeXt	57.79 ± 0.60	53.13 ± 0.80	52.04 ± 0.83
	CellViT	60.96 ± 0.22	55.75 ± 0.29	54.64 ± 0.27
	TME-ContextNet	**62.83 ± 1.75**	**57.56 ± 1.99**	**56.49 ± 2.04**
Connective	HoVer-Net	48.92 ± 1.28	39.42 ± 1.07	36.84 ± 1.07
	Zhang et al. (2024)	46.88 ± 0.33	38.34 ± 0.67	35.95 ± 0.84
	HoVer-NeXt	44.46 ± 1.88	35.39 ± 1.81	32.67 ± 1.86
	CellViT	52.03 ± 0.88	43.04 ± 0.61	40.59 ± 0.51
	TME-ContextNet	**56.13 ± 0.39**	**47.09 ± 0.74**	**44.72 ± 0.89**
Dead	HoVer-Net	85.92 ± 19.86	85.87 ± 19.95	85.85 ± 19.99
	Zhang et al. (2024)*	**97.07 ± 0.57**	**97.07 ± 0.57**	**97.07 ± 0.57**
	HoVer-NeXt	89.10 ± 4.83	88.94 ± 4.87	88.87 ± 4.89
	CellViT	86.82 ± 2.03	86.66 ± 2.00	86.59 ± 2.00
	TME-ContextNet	94.01 ± 5.69	93.85 ± 5.86	93.79 ± 5.93
Epithelial	HoVer-Net	77.88 ± 5.44	75.78 ± 5.19	75.38 ± 5.15
	Zhang et al. (2024)	81.58 ± 2.80	79.77 ± 2.89	79.38 ± 2.92
	HoVer-NeXt	75.51 ± 2.78	73.47 ± 2.83	73.05 ± 2.86
	CellViT	86.32 ± 0.99	84.23 ± 1.11	83.84 ± 1.13
	TME-ContextNet	**87.31 ± 2.94**	**85.13 ± 3.04**	**84.75 ± 3.07**

Values are percentages, reported as mean ± sample standard deviation across three test folds. Pixel $P \times R$ is the product of pixel precision and recall, averaged over images. Instance-level quality is reported as mPQ+ in Table 1. *No dead-class nuclei were predicted in any fold. The high image-averaged pixel values reflect the empty/empty-mask score of one; dead-class instance PQ is zero.

Table 3 Component configurations on PanMix.

Configuration	GTD	LGCM	CSS	Dice (%)	IoU (%)	mPQ+ (%)
HoVer-Net	—	—	—	69.80	67.22	47.85
HoVer-Net + CSS	—	—	✓	73.83	71.36	52.48
HoVer-Net + CSS + attention	—	Partial	✓	71.89	69.34	50.70
w/o self-attention	✓	Partial	✓	77.45	75.10	56.19
w/o LGCM and SE	✓	—	✓	75.69	73.28	54.03
w/o SE	✓	✓	✓	70.50	68.18	32.54
Full model	✓	✓	✓	**80.23**	**78.52**	**67.62**

Values are percentages. Partial denotes the attention-only or dilation-only LGCM configuration identified by the row label. SE is an independent switch; its removal is stated explicitly. mPQ+ uses foreground instances and IoU > 0.5.

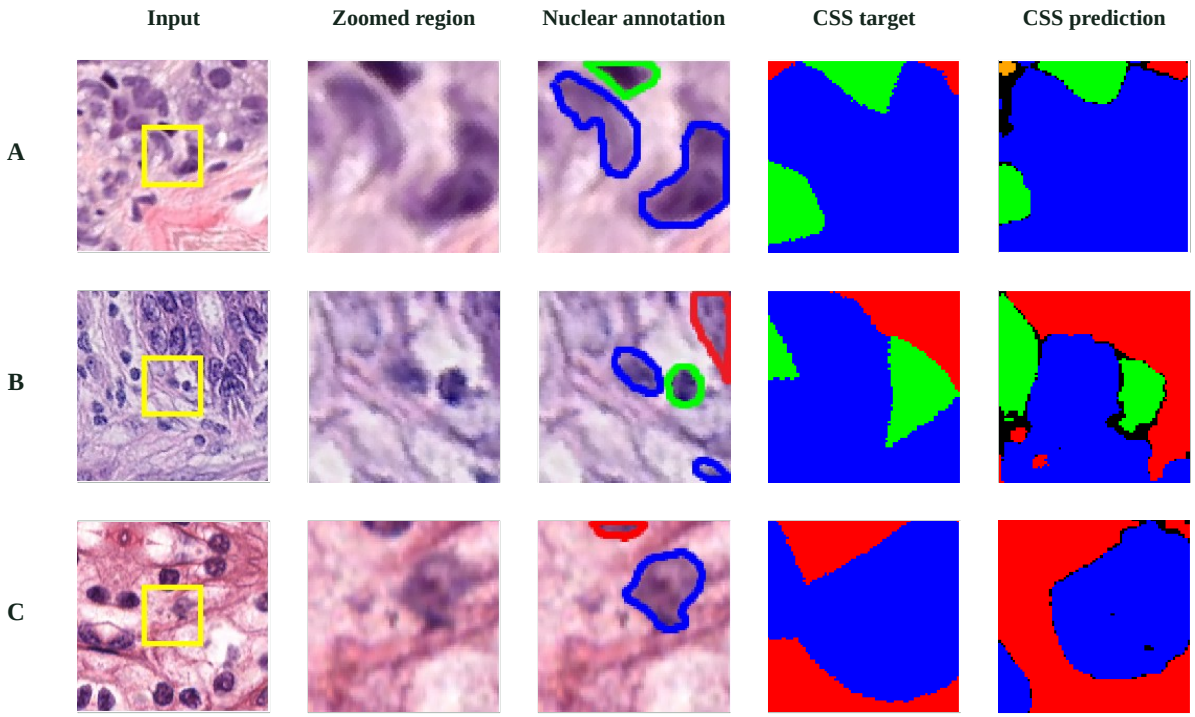


Fig. 5 Examples of spatial supervision targets and predictions. From left to right, each row shows the input image, enlarged region of interest, nuclear annotations, annotation-derived CSS target, and predicted CSS classes. The spatial target extends class assignments into non-nuclear regions while retaining the annotated nuclear labels

51.32% Dice and 41.47% IoU. Class-agnostic distance also performs below the no-CSS condition, with 49.39% Dice and 41.08% IoU; EDT exceeds this construction by 4.21 and 4.42 percentage points. The results favor extending class-specific nuclear neighborhoods through EDT over the tested Gaussian and type-agnostic alternatives. They also show that the benefit of auxiliary spatial supervision depends on its target definition. These controls distinguish the use of class information from the mere presence of an auxiliary distance task. Removing class identity changes what the model is asked to encode, while class weights and Gaussian accumulation alter the spatial assignments. Their observed scores support the particular EDT construction in this control setting. The alternatives trained with one seed provide descriptive comparisons rather than a statistical ranking. Figure 5 illustrates the spatial regions learned by the auxiliary task.

Table 4 CSS target constructions in a dedicated HoVer-Net-based auxiliary-supervision model on CoNSeP. EDT is the distance-transform construction adopted for CSS.

Target	Seeds	Dice (%)	IoU (%)
No CSS	3	51.91 ± 3.30	43.60 ± 3.15
EDT	3	53.60 ± 1.30	45.50 ± 0.99
Uniform Gaussian	1	50.45	42.49
Class-weighted Gaussian	1	48.86	40.81
Constant class weights	1	51.26	42.73
Permuted class weights	1	51.32	41.47
Class-agnostic distance	1	49.39	41.08

No CSS and EDT: mean ± sample standard deviation across three seeds. Other targets: one seed. The control architecture is held fixed across targets. Values are percentages and follow the original pixel evaluator.

4.5 Decoder organization and LGCM

Table 5 separates sharing depth, decoder architecture, and LGCM effects. In the PanMix integration-depth block, Stage II gives the highest Dice and IoU among the three integration stages. Stage III uses fewer parameters but has lower overlap, consistent with a trade-off between sharing and task-specific representation.

The controlled CoNSeP decoder blocks show that the preferred arrangement depends on auxiliary supervision. Without CSS, the independent decoder has the highest Dice, at 56.55%, followed by the plain shared decoder and GTD. With CSS, GTD reaches Dice of 54.98%, compared with 51.72% for the independent decoder, while using 36.40 million rather than 65.18 million parameters, a reduction of 44.2%. This combination of higher overlap and reduced decoder duplication supports pairing GTD with CSS.

LGCM has a positive effect in both dedicated blocks. With SE disabled, adding LGCM to the HoVer-Net control raises Dice from 47.47% to 56.61% and IoU from 39.65% to 48.19%. In the full model with SE enabled, the effect is smaller: Dice is 54.98% with LGCM and 54.33% without it. The two blocks indicate that LGCM can supply useful context independently, while its marginal contribution changes when the remaining components are present.

Table 5 Decoder structures and the contribution of LGCM. Comparisons are grouped by supervision and architectural setting.

Configuration	Params (M)	Dice (%)	IoU (%)
PanMix: integration depth			
HoVer-Net	54.74	69.80	67.22
Integration Stage I	40.60	75.03	68.71
Integration Stage II	36.80	78.72	75.51
Integration Stage III	35.31	73.61	70.94
CoNSeP: decoder / no CSS			
Independent decoder	65.18	56.55	47.87
Plain shared decoder	35.99	54.66	46.88
GTD	36.40	53.75	45.33
CoNSeP: decoder / with CSS			
Full / GTD	36.40	54.98	46.62
Independent decoder	65.18	51.72	43.43
CoNSeP: LGCM only / SE off			
HoVer-Net	54.74	47.47	39.65
HoVer-Net + LGCM	54.79	56.61	48.19
CoNSeP: LGCM in full / SE on			
Full model	36.40	54.98	46.62
Full model w/o LGCM	36.38	54.33	45.96

Params: million parameters. Accuracy values are percentages. The no-CSS and with-CSS blocks evaluate decoder organization under their stated supervision. LGCM-only comparisons disable SE; the full-model comparison enables SE. The rows are interpreted within their respective comparison blocks.

4.6 Auxiliary-weight sensitivity on PanMix

Figure 6 examines how the strength of the EDT auxiliary objective changes overlap on PanMix. The recorded boxplots show higher central Dice and IoU values for moderate weights, with $\omega = 0.6$ giving the highest visible medians among the examined settings. Increasing the weight beyond this range does not yield a monotonic improvement. The pattern is consistent

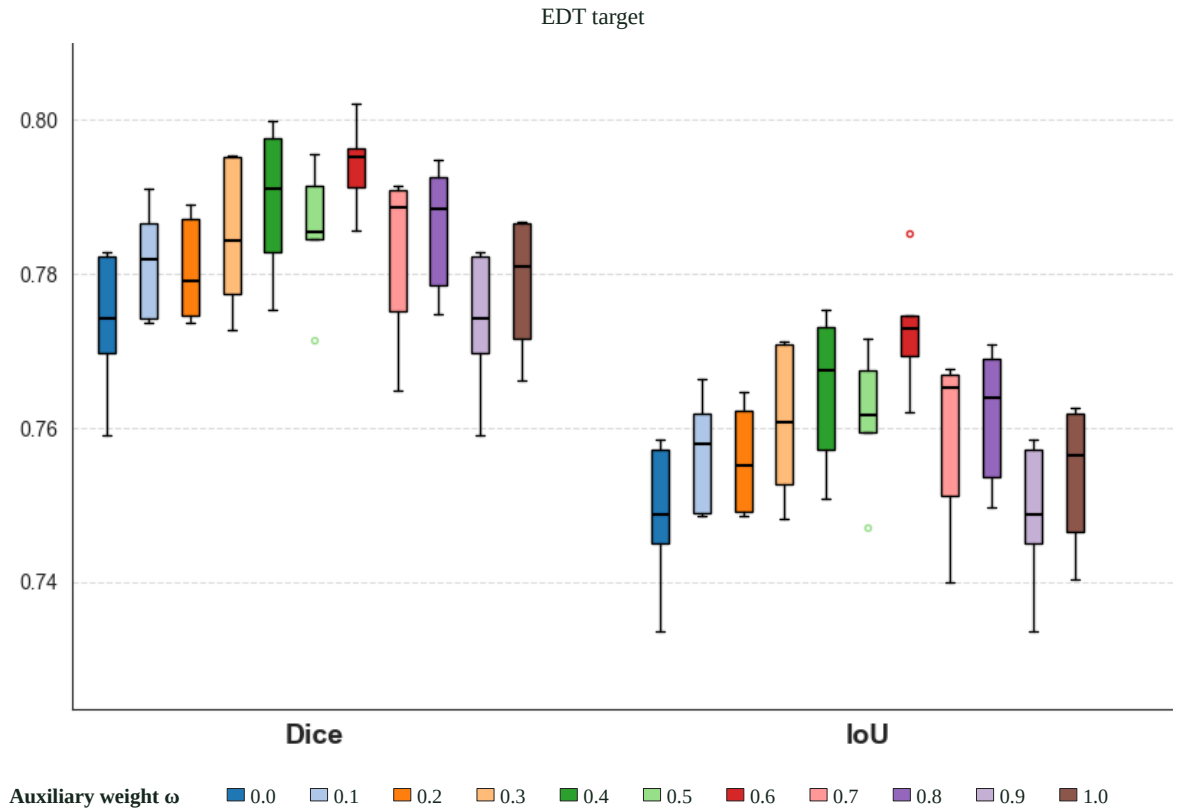


Fig. 6 Effect of auxiliary-loss weight ω on PanMix with the EDT-based CSS target. Colors denote weights from 0.0 to 1.0 in increments of 0.1. Each weight uses one training run with the remaining settings held fixed. The boxplots summarize per-image test scores, rather than variation across random seeds. Scores are shown as fractions.

with auxiliary supervision being useful when it contributes to representation learning without dominating the primary tasks.

This observation complements the target controls. A target construction determines the auxiliary gradient direction, while ω changes the magnitude of its contribution. Equation (15) separates these two effects through the inner-product term. The sensitivity experiment shows how changing the auxiliary-loss weight affects the observed overlap, complementing the optimization role of ω in Eq. (15).

4.7 Computational cost

Table 6 reports model size, arithmetic cost, latency, and peak GPU memory. In the RTX 4090 main-model session, TME-ContextNet uses 36.40 million parameters and 267.77 GFLOPs, compared with 54.74 million and 398.32 GFLOPs for HoVer-Net. These values reduce parameters by 33.5% and arithmetic cost by 32.8%. Its mean network latency is 19.37 ms and its mean image latency is 357.24 ms, compared with 24.78 ms and 444.17 ms for HoVer-Net. The CSS and no-CSS rows have the same listed parameter count and arithmetic cost, consistent with auxiliary supervision being confined to training. Both retain the same inference prediction path.

The RTX 4090 decoder blocks provide a separate comparison of sharing arrangements. In the block that includes all three decoder structures, GTD and the plain shared decoder use fewer parameters, fewer operations, and less peak memory than the independent decoder. The plain shared decoder has the lower forward latency, while GTD has the lower image latency in that block. Repeated timings in the two 4090 sessions vary, so the measurements are presented

in their original blocks. Network and image latency are complementary: post-processing and image assembly account for a substantial part of the latter.

Table 6 Computational cost of TME-ContextNet and decoder variants on RTX 4090. Measurement sessions are presented separately.

Method	Params (M)	GFLOPs	Network (ms)	Image (ms)	P95 (ms)	Memory (GiB)
RTX 4090: main-model session						
HoVer-Net	54.74	398.32	24.78	444.17	463.02	0.50
TME-ContextNet	36.40	267.77	19.37	357.24	410.72	0.43
TME-ContextNet w/o CSS	36.40	267.77	19.21	351.31	381.46	0.43
RTX 4090: session 1						
GTD / no CSS	36.40	267.77	21.33	388.49	397.46	0.43
Plain shared / no CSS	35.99	262.53	20.84	385.69	412.30	0.43
RTX 4090: session 2						
GTD / no CSS	36.40	267.77	19.02	339.07	341.82	0.43
Plain shared / no CSS	35.99	262.53	18.75	345.39	385.64	0.43
Independent / no CSS	65.18	410.79	33.23	587.62	607.95	0.54

FP32, batch size one. Network: mean forward latency. Image: mean latency including post-processing and excluding disk input/output for a 256×256 image. P95: 95th-percentile image latency. GFLOPs: one native network input patch. Memory: peak GPU memory. The main-model block was remeasured on October 28. Decoder sessions 1 and 2 retain the October 21 and October 22 measurements, respectively. Models within each block were timed sequentially in the same GPU session.

5 Discussion

5.1 Spatial supervision for quantitative cellular analysis

TME-ContextNet combines dense spatial supervision with grouped decoding to address the different contextual requirements of nuclear delineation and type recognition. The complete framework has the highest overall pixel and instance scores among the evaluated methods across PanMix, CoNSeP, and PanNuke. Improvements across several PanNuke cell populations complement this overall comparison. The computational results show that the joint outputs can be obtained with fewer parameters and less arithmetic than the HoVer-Net comparator, making the design relevant to quantitative cellular analysis in imaging informatics.

CSS makes a specific part of the annotation information available to representation learning. Nuclear type labels identify cell populations at annotated nuclear locations. Their distance-based extension additionally organizes the surrounding space according to proximity to those populations. This connects the biological motivation of tissue organization to a reproducible geometric target. The nearest-region interpretation and perturbation analysis explain what that target represents and how its assignments respond to small changes in nuclear support. This spatial partition provides dense contextual supervision for the joint nuclear outputs.

The existing controls support that interpretation at two levels. The PanMix component comparison shows higher joint metrics when CSS is added to the HoVer-Net baseline, and the dedicated HoVer-Net-based CoNSeP target controls show higher pixel overlap for EDT than for no CSS. Across the tested target constructions, EDT has the highest Dice and IoU, whereas the Gaussian and class-agnostic alternatives perform below the no-CSS condition. These results favor the class-conditioned EDT construction and show that the spatial target definition influences the benefit of auxiliary learning. Equation (15) explains how different targets can alter shared-parameter updates through their alignment with the primary objective. It explains the contribution of the auxiliary objective to shared-feature optimization.

For imaging informatics, the useful output is a population of individually delineated and typed nuclei from which cellular summaries can be derived. The overall mPQ+ results address this joint output more directly than class-map overlap alone. The inclusion of general segmentation methods establishes a demanding pixel-level reference, while the specialist comparison tests whether that representation also supports typed instances. These results demonstrate the potential of TME-ContextNet as an efficient component for quantitative cellular analysis in computational pathology.

5.2 Interaction between supervision and task sharing

GTD addresses the accompanying feature-sharing problem. Shared decoding is established in nuclei analysis, including the enhanced framework of Zhang et al. [5]. The present design groups the late representations by geometry and semantics and pairs CSS with nuclear type prediction. Its advantage is clearest in the with-CSS comparison, where it combines higher Dice with fewer parameters than independent decoding. Independent decoding has higher Dice in the no-CSS block. These observations support the coupling of grouped decoding and spatial supervision as the relevant design choice, with a clear accuracy–cost trade-off.

LGCM provides complementary context through encoder dilation and decoder self-attention. The dedicated SE-off comparison shows that LGCM can improve the simpler architecture on its own, while the full-model comparison shows a smaller positive increment when the other components are present. The difference in marginal benefit is consistent with contextual information being supplied by several parts of the network. The ablations support complementary contributions from contextual features and grouped spatial supervision.

The PanMix weight analysis provides a related optimization perspective. Its non-monotonic trend indicates that increasing the amount of auxiliary supervision is not sufficient by itself. The finite-step argument explains why auxiliary direction and weight both enter the update, while the experiments show the behavior of specific targets and settings. The gradient analysis explains the optimization role of the auxiliary objective, while the weight experiment characterizes its empirical behavior.

5.3 Computational implications and further development

Grouped decoding reduces redundant computation in repeated cellular analysis. The image-level timings identify tiling and instance assembly as substantial additional processing costs, making efficient assembly a useful companion to network acceleration. Whole-slide throughput also depends on tile scheduling and batching. The challenging examples in Fig. 7 identify three concrete development targets: nuclear extent and separation, false-positive suppression, and nuclear type discrimination. Adaptive spatial supervision and efficient image-level assembly offer complementary directions for improving these outputs and their processing cost.

6 Conclusions

TME-ContextNet combines a class-conditioned distance target, hierarchical local–global features, and grouped multi-task decoding for joint nuclei instance segmentation and classification. The mathematical formulation connects the implemented auxiliary field to a nearest-class spatial partition and explains how its training signal reaches shared features. Across the three evaluated datasets, the complete framework achieves the highest overall Dice, IoU, and mPQ+ values, while the computational measurements show reduced cost relative to the evaluated HoVer-Net implementation. Target and decoder controls identify the conditions supporting these gains, providing a practical basis for further development of context-aware cellular analysis in computational pathology.

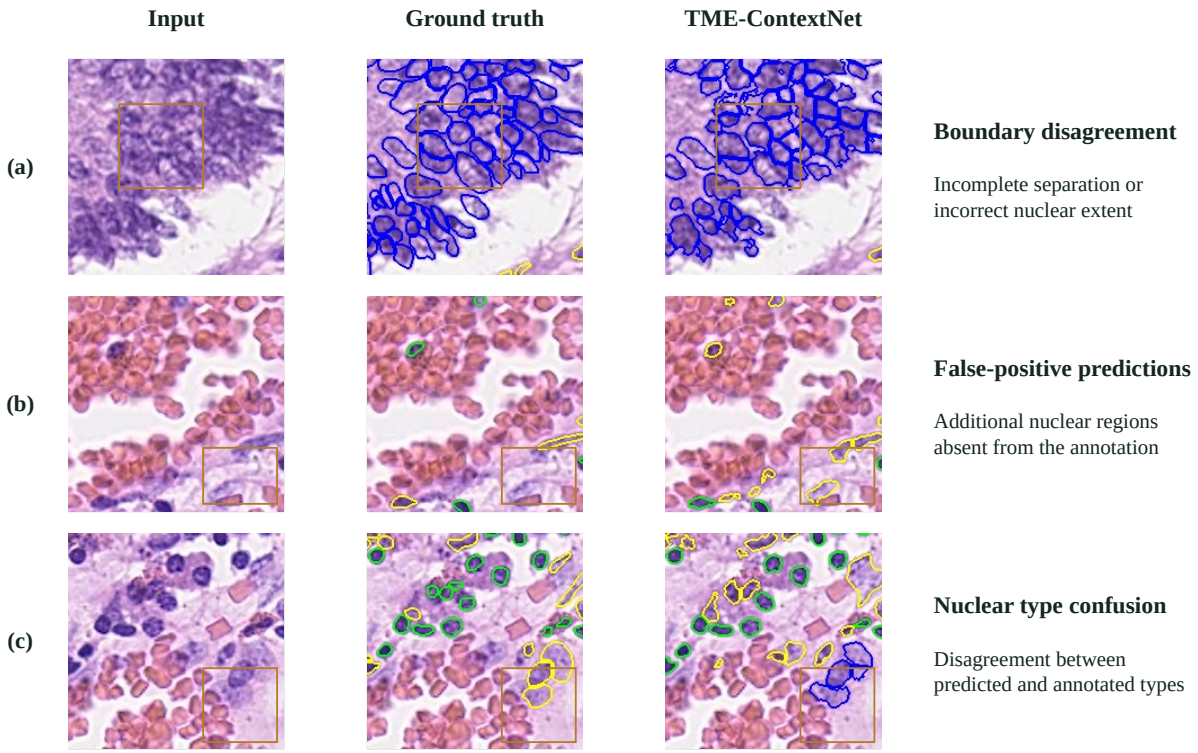


Fig. 7 Examples of residual prediction errors on CoNSeP. Each row shows the input, ground truth, and TME-ContextNet output. Outlined regions highlight (a) incomplete separation or disagreement in nuclear extent, (b) additional predicted nuclear regions absent from the annotation, and (c) disagreement between predicted and annotated nuclear types. Nuclear contour colors follow the CoNSeP legend in Fig. 4

A Derivations for spatial supervision and optimization

A.1 Nearest-class assignment and numerical clipping

For any two present classes a and b whose scores are not clipped,

$$\log S_a(p) - \log S_b(p) = \frac{d_b(p)^2 - d_a(p)^2}{2\sigma^2}. \tag{23}$$

Thus $S_a(p) > S_b(p)$ exactly when $d_a(p) < d_b(p)$, proving Eq. (4) for a unique, unclipped winner. The common scale cancels in the ordering. If every class distance is at least $\sigma\sqrt{2\log(1/\varepsilon)}$, all scores equal the floor and the implementation's ordered tie rule determines the assigned class. This threshold identifies the far-field region governed by the ordered tie rule.

For comparison, an unclipped class-weighted score $S_c = w_c \exp[-d_c^2/(2\sigma_c^2)]$ has assignment rule

$$\arg\max_c S_c = \arg\min_c \left\{ \frac{d_c^2}{2\sigma_c^2} - \log w_c \right\}. \tag{24}$$

Weights and class-specific scales move spatial decision boundaries. A Gaussian sum over multiple nuclear pixels is a different construction: it can also change with the number and distribution of contributing pixels. These distinctions explain why the EDT, Gaussian, and weight-control conditions are treated as different target designs.

A.2 Perturbation of class supports

Assume nonempty supports F_c and F'_c satisfy $d_H(F_c, F'_c) \leq \delta$, where d_H is their Hausdorff distance. For a nearest point $q \in F_c$, there is a point $q' \in F'_c$ with $\|q - q'\| \leq \delta$. The triangle inequality gives

$$d'_c(p) \leq \|p - q'\| \leq \|p - q\| + \delta = d_c(p) + \delta. \tag{25}$$

Interchanging the two supports proves $|d'_c(p) - d_c(p)| \leq \delta$. If the initial winning class is a and every other class b satisfies $d_b(p) - d_a(p) > 2\delta$, then

$$d'_b(p) - d'_a(p) \geq d_b(p) - d_a(p) - 2\delta > 0. \tag{26}$$

Hence the nearest class is unchanged under the perturbation. This statement assumes the same set of present classes and excludes target pixels relabeled as nuclear foreground and score-floor ties. Under these conditions, the distance margin determines the robustness of the spatial assignment.

A.3 A finite-step form of the auxiliary-gradient argument

If the primary loss is L-smooth in the shared parameters, the descent lemma bounds the update in Eq. (15) by

$$\mathcal{L}_{\text{main}}(\theta') - \mathcal{L}_{\text{main}}(\theta) \leq -\eta \left(\|g\|^2 + \omega \langle g, a \rangle \right) + \frac{L\eta^2}{2} \|g + \omega a\|^2. \tag{27}$$

If $B = \|g\|^2 + \omega\langle g, a\rangle > 0$ and $g + \omega a \neq 0$, the bound is negative for $0 < \eta < 2B/(L\|g + \omega a\|^2)$. This bound relates the allowable step size to the auxiliary weight and target-induced gradient direction, explaining their joint role in primary-loss descent.

B Definitions of the CoNSeP target controls

The Gaussian controls use an unnormalized sum over nuclear pixels,

$$G_c(p) = w_c \sum_{q \in F_c} \exp\left[-\frac{\|p - q\|_2^2}{2\sigma_c^2} \right]. \tag{28}$$

The largest score assigns each background pixel; nuclear labels remain unchanged. In class-ID order $(1, 2, 3, 4)$, Uniform Gaussian uses $w = (1, 1, 1, 1)$ and $\sigma = (5, 5, 5, 5)$ pixels. Class-weighted Gaussian uses $w = (1, 1, 2, 2)$ and $\sigma = (5, 5, 8, 8)$. Constant class weights uses $w = (1, 1, 1, 1)$, and Permuted class weights uses $w = (2, 2, 1, 1)$, both retaining $\sigma = (5, 5, 8, 8)$. Far-field assignments use the log domain where needed to avoid numerical underflow.

Class-agnostic distance uses $d(p) = \min_{q \in F} \|p - q\|_2$ for the union of nuclei $F = \{p : I(p) > 0\}$. Its five labels encode $d = 0$, $0 < d \leq 4$, $4 < d \leq 8$, $8 < d \leq 16$, and $d > 16$ pixels; nucleus-free images receive the farthest band. Thus, the target supervises geometric distance bands. No CSS disables the auxiliary loss; EDT follows Eqs. (1)–(3).

C Additional architecture and qualitative illustrations

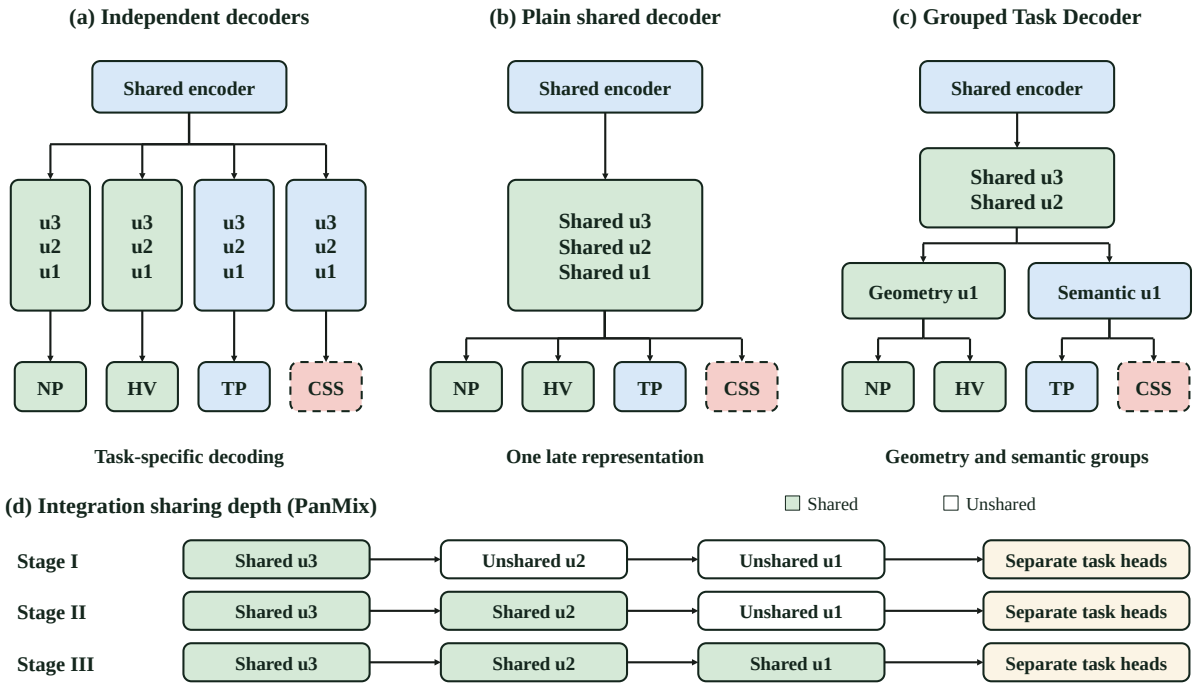


Fig. 8 Decoder organization and integration sharing depth. (a) Independent decoders process NP, HV, TP, and CSS separately. (b) A plain shared decoder uses one representation through stages u3, u2, and u1 before separate task heads. (c) GTD shares u3 and u2, then separates u1 refinement into geometry-related (NP/HV) and semantic (TP/CSS) groups. (d) For the PanMix integration-depth controls in Table 5, Stage I shares u3, Stage II shares u3 and u2, and Stage III shares u3, u2, and u1. The task heads remain separate in all three stages

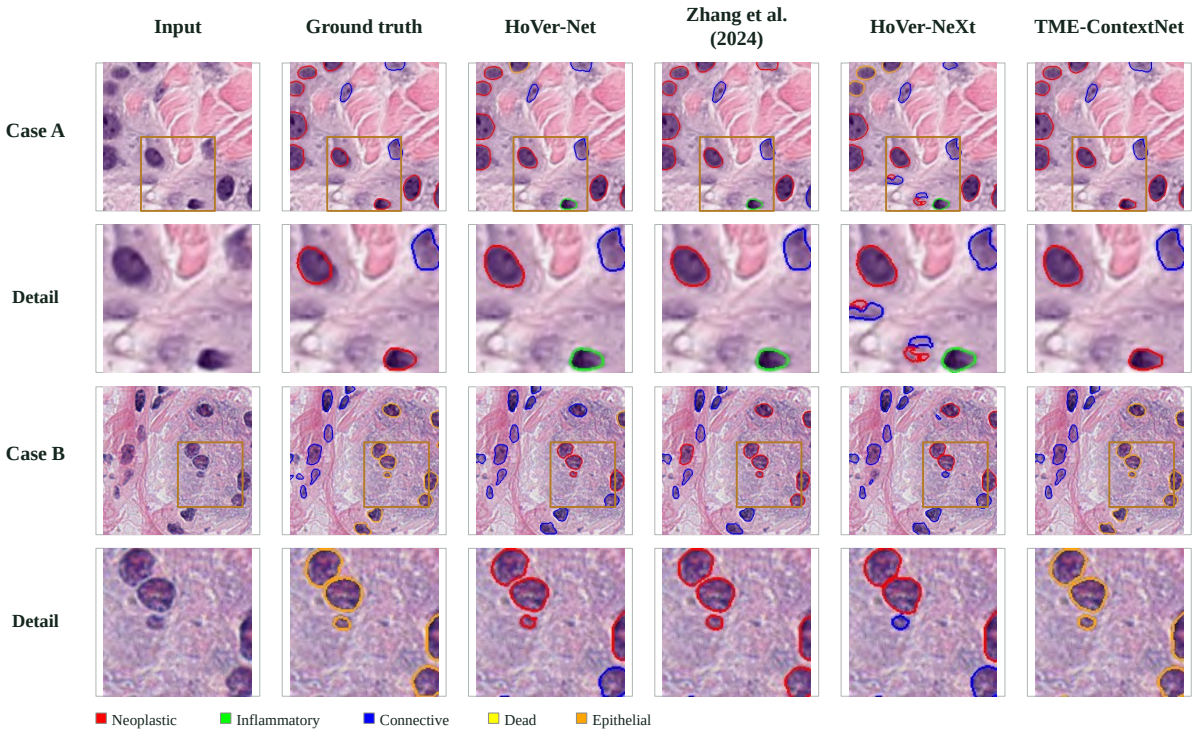


Fig. 9 PanNuke examples with enlarged details. Columns show the H&E input, ground truth, HoVer-Net, Zhang et al. (2024), HoVer-NeXt, and TME-ContextNet. Each overview is followed by an enlarged view of the outlined region. All methods use the same image coordinates and class colors. The examples illustrate nuclear delineation and type assignments for the five PanNuke categories

D Additional comparisons with general segmentation methods

The following illustrations complement the specialist comparison in Fig. 4. General segmentation methods provide nuclear class regions; their displayed contours should be interpreted as boundaries of those class regions. The same H&E field and class colors are retained across methods in each row.

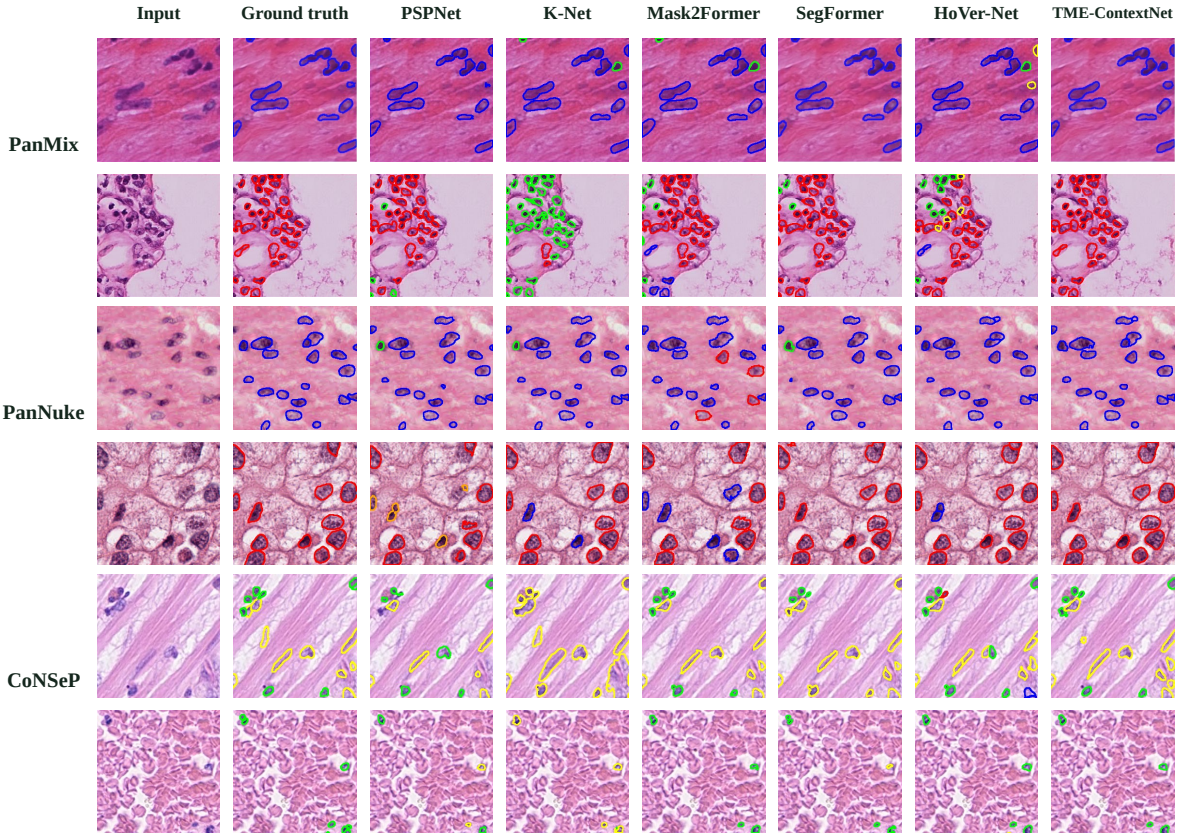


Fig. 10 Additional qualitative class-map comparisons on PanMix, PanNuke, and CoNSeP. Two examples per dataset are shown for the H&E input, ground truth, PSPNet, K-Net, Mask2Former, SegFormer, HoVer-Net, and TME-ContextNet. These examples compare local class assignments and region boundaries. All methods use the same image coordinates within each example

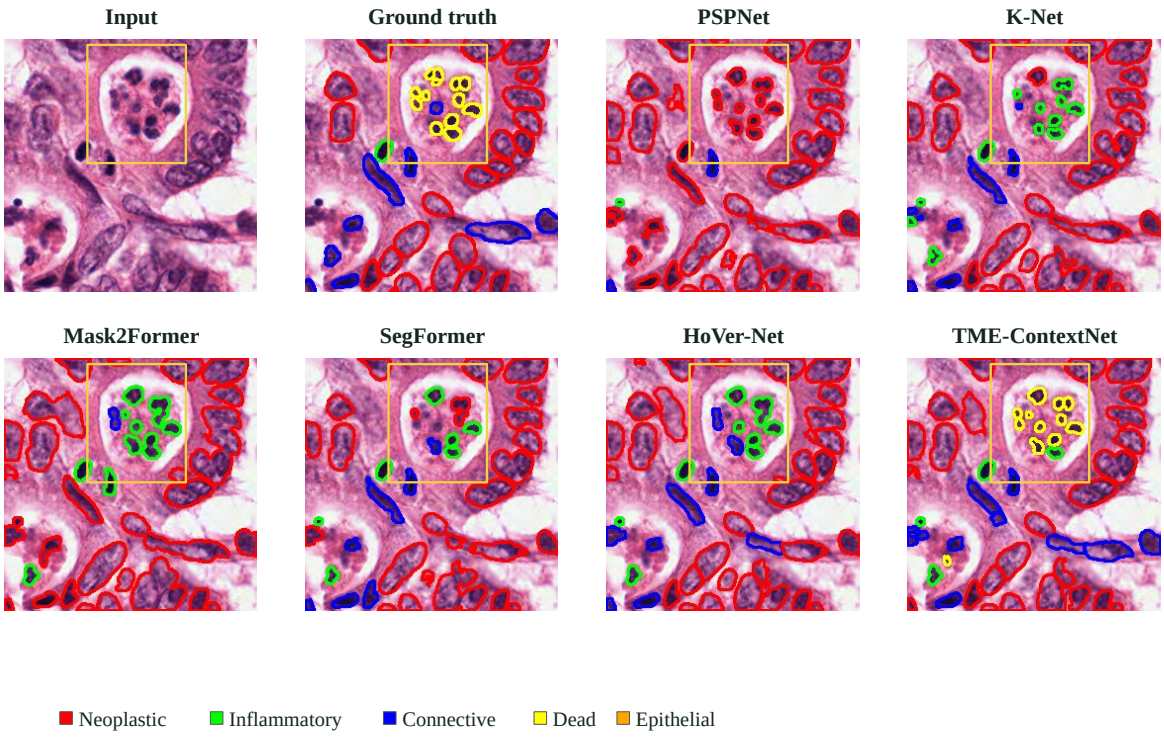


Fig. 11 PanNuke example of local nuclear type assignments. The outlined region identifies the same group of nuclei across the input, ground truth, and six method outputs. The example includes disagreement between neoplastic and dead-cell labels among methods. It illustrates local type ambiguity and complements the class-wise quantitative results

Declarations

Funding. This work was supported by the Ministry of Science and Technology of China under Grant SQ2020YFA0713800 and the National Natural Science Foundation of China under Grants 12001273 and 12201109.

Competing interests. The authors declare that they have no competing interests.

Ethics and consent. The pathological images were anonymized before analysis. No identifiable individual participant information is presented.

Data and code availability. CoNSeP and PanNuke are available from their respective public sources [4, 33, 34]. Source-code access and PanMix research data requests are managed through the project page: https://github.com/rukang5413-jpg/TME-ContextNet.

References

[1] Sirinukunwattana K, Raza SEA, Tsang YW, et al (2016) Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. IEEE Transactions on Medical Imaging 35(5):1196–1206

[2] Yuan Y, Failmezger H, Rueda OM, et al (2012) Quantitative image analysis of cellular heterogeneity in breast tumors complements genomic profiling. Science Translational Medicine 4(157):157ra143–157ra143

[3] Srinidhi CL, Ciga O, Martel AL (2021) Deep neural network models for computational histopathology: A survey. Medical Image Analysis 67:101813

[4] Graham S, Vu QD, Raza SEA, et al (2019) HoVer-Net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images. Medical Image Analysis 58:101563

[5] Zhang W, Yang S, Luo M, et al (2024) Keep it accurate and robust: An enhanced nuclei analysis framework. Computational and Structural Biotechnology Journal 24:699–710

[6] Baumann E, Dislich B, Rumberger JL, et al (2024) HoVer-NeXt: A fast nuclei segmentation and classification pipeline for next generation histopathology. In: Proceedings of the International Conference on Medical Imaging with Deep Learning, pp 61–86, URL https://proceedings.mlr.press/v250/baumann24a.html

[7] Ronneberger O, Fischer P, Brox T (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, pp 234–241, https://doi.org/10.1007/978-3-319-24574-4_28

[8] Raza SEA, Cheung L, Shaban M, et al (2019) Micro-Net: A unified model for segmentation of various objects in microscopy images. Medical Image Analysis 52:160–173. https://doi.org/10.1016/j.media.2018.12.003

[9] Schmidt U, Weigert M, Broaddus C, et al (2018) Cell Detection with Star-Convex Polygons. In: Medical Image Computing and Computer Assisted Intervention – MICCAI 2018, pp 265–273, https://doi.org/10.1007/978-3-030-00934-2_30

[10] Stringer C, Wang T, Michaelos M, et al (2021) Cellpose: a generalist algorithm for cellular segmentation. Nature Methods 18(1):100–106. https://doi.org/10.1038/s41592-020-01018-x

[11] Doan TNN, Song B, Vuong TTL, et al (2022) SONNET: A self-guided ordinal regression neural network for segmentation and classification of nuclei in large-scale multi-tissue histology images. IEEE Journal of Biomedical and Health Informatics 26(7):3218–3228. https://doi.org/10.1109/JBHI.2022.3149936

[12] Yao K, Huang K, Sun J, et al (2024) PointNu-Net: Keypoint-Assisted Convolutional Neural Network for Simultaneous Multi-Tissue Histology Nuclei Segmentation and Classification. IEEE Transactions on Emerging Topics in Computational Intelligence 8(1):802–813. https://doi.org/10.1109/tetci.2023.3281864

[13] Pan X, Cheng J, Hou F, et al (2023) SMILE: Cost-sensitive multi-task learning for nuclear segmentation and classification with imbalanced annotations. Medical Image Analysis 88:102867. https://doi.org/10.1016/j.media.2023.102867

[14] Nguyen K, Bredno J, Knowles DA (2015) Using contextual information to classify nuclei in histology images. In: 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), IEEE, pp 995–998

[15] Bai T, Xu J, Zhang Z, et al (2022) Context-aware learning for cancer cell nucleus recognition in pathology images. Bioinformatics 38(10):2892–2898. https://doi.org/10.1093/bioinformatics/btac167

[16] Nunes JD, Montezuma D, Oliveira D, et al (2025) A survey on cell nuclei instance segmentation and classification: Leveraging context and attention. Medical Image Analysis 99:103360. https://doi.org/10.1016/j.media.2024.103360

[17] Hörst F, Rempe M, Heine L, et al (2024) CellViT: Vision transformers for precise cell segmentation and classification. Medical Image Analysis 94:103143. https://doi.org/10.1016/j.media.2024.103143

[18] Hörst F, Rempe M, Becker H, et al (2026) CellViT++: Energy-efficient and adaptive cell segmentation and classification using foundation models. Computer Methods and Programs in Biomedicine 277:109206. https://doi.org/10.1016/j.cmpb.2025.109206

[19] Su J, Zhu T, Cai J, et al (2026) Adapting SAM to nuclei instance segmentation and classification via Cooperative Fine-Grained Refinement. Medical Image Analysis 112:104144. https://doi.org/10.1016/j.media.2026.104144

[20] Chen W, Chang F, Qi J (2026) CAS-UNet: A Context-Adaptive Encoder-Decoder Network with Boundary-Distance Auxiliary Supervision for Histopathological Nuclei Segmentation. Journal of Imaging Informatics in Medicine https://doi.org/10.1007/s10278-026-02390-4

[21] Tommasino C, Russo C, Rinaldi AM (2026) NuLite - lightweight and fast model for nuclei instance segmentation and classification. Biomedical Signal Processing and Control 114:109333. https://doi.org/10.1016/j.bspc.2025.109333

[22] Zhao H, Shi J, Qi X, et al (2017) Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2881–2890

[23] Zhang W, Pang J, Chen K, et al (2021) K-Net: Towards unified image segmentation. In: Advances in Neural Information Processing Systems, vol 34. Curran Associates, Inc., pp 10326–10338

[24] Cheng B, Misra I, Schwing AG, et al (2022) Masked-attention mask transformer for universal image segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1290–1299

[25] Xie E, Wang W, Yu Z, et al (2021) SegFormer: Simple and efficient design for semantic segmentation with transformers. In: Advances in Neural Information Processing Systems, vol 34. Curran Associates, Inc., pp 12077–12090

[26] He K, Zhang X, Ren S, et al (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778

[27] Chen LC, Zhu Y, Papandreou G, et al (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on

computer vision (ECCV), pp 801–818

[28] Huang G, Liu Z, Van Der Maaten L, et al (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708

[29] Wang X, Girshick R, Gupta A, et al (2018) Non-local neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 7794–7803

[30] Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7132–7141

[31] Kendall A, Gal Y, Cipolla R (2018) Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 7482–7491

[32] Yu T, Kumar S, Gupta A, et al (2020) Gradient surgery for multi-task learning. In: Advances in Neural Information Processing Systems, vol 33. Curran Associates, Inc., pp 5824–5836

[33] Gamper J, Alemi Koohbanani N, Benet K, et al (2019) PanNuke: an open pan-cancer histology dataset for nuclei instance segmentation and classification. In: Digital Pathology: 15th European Congress, ECDP 2019, Warwick, UK, April 10–13, 2019, Proceedings 15, Springer, pp 11–19

[34] Gamper J, Koohbanani NA, Benes K, et al (2020) PanNuke dataset extension, insights and baselines. arXiv https://doi.org/10.48550/arXiv.2003.10778

[35] Kirillov A, He K, Girshick R, et al (2019) Panoptic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 9404–9413