IAUNet: Instance-Aware U-Net

Yaroslav Prytula[1,2] Illia Tsiporenko[1] Ali Zeynalli[1] Dmytro Fishman[1,3]
[1]Institute of Computer Science, University of Tartu
[2]Ukrainian Catholic University, [3]STACC OÜ, Tartu, Estonia
{yaroslav.prytula, illia.tsiporenko, ali.zeynalli, dmytro.fishman}@ut.ee
s.prytula@ucu.edu.ua

Abstract

Instance segmentation is critical in biomedical imaging to accurately distinguish individual objects like cells, which often overlap and vary in size. Recent query-based methods, where object queries guide segmentation, have shown strong performance. While U-Net has been a go-to architecture in medical image segmentation, its potential in query-based approaches remains largely unexplored. In this work, we present IAUNet, a novel query-based U-Net architecture. The core design features a full U-Net architecture, enhanced by a novel lightweight convolutional Pixel decoder, making the model more efficient and reducing the number of parameters. Additionally, we propose a Transformer decoder that refines object-specific features across multiple scales. Finally, we introduce the 2025 Revvity Full Cell Segmentation Dataset, a unique resource with detailed annotations of overlapping cell cytoplasm in brightfield images, setting a new benchmark for biomedical instance segmentation. Experiments on multiple public datasets and our own show that IAUNet outperforms most state-of-the-art fully convolutional, transformer-based, and query-based models and cell segmentation-specific models, setting a strong baseline for cell instance segmentation tasks. Code is available at https://github.com/SlavkoPrytula/IAUNet

1. Introduction

Accurate cell instance segmentation is crucial in biomedical imaging [1], as it enables the precise identification and analysis of individual cells. This process is essential for understanding cellular behaviors and disease mechanisms [24]. However, the diverse and irregular shapes of cells present significant challenges for segmentation algorithms [35, 46]. Variations in cell morphology, overlapping structures, and differing imaging conditions can lead to segmentation errors [1]. Addressing these challenges requires the development of advanced segmentation models capable of handling the complexities associated with cell shapes. Deep learning models have driven substantial progress in cell segmentation, often surpassing traditional methods [21, 42, 52]. However, cell segmentation remains challenging due to heterogeneous cell appearances, overlaps, and varied object densities across different microscopy modalities, requiring models that generalize well across conditions.

Brightfield microscopy, valued for its simplicity and affordability, presents unique challenges for segmentation [1]. Unlike fluorescence microscopy, which requires staining, and phase-contrast microscopy, which relies on specialized optics to enhance contrast in transparent specimens, brightfield uses natural light alone [45]. This makes brightfield ideal for real-time observation in both research and clinical settings [1, 36, 51]. However, brightfield images are inherently low-contrast, noisy, and variable, making precise cell segmentation difficult and underscoring the need for specialized approaches tailored to this modality.

Many previous works have adapted instance segmentation models from natural images to medical imaging without model-specific adjustments [22, 40, 50]. In contrast to many of these methods, U-Net [42] has long been a go-to architecture for semantic segmentation. Its lightweight framework, characterized by skip connections and an encoder-decoder structure, enables precise localization and the effective capture of intricate details, making it especially well-suited for biomedical applications. U-Net's efficiency is particularly advantageous when working with smaller microscopy datasets, as it typically requires less data to train compared to more complex models. This is why we chose to focus on U-Net in our work, building on its established popularity and applicability to microscopy data.

Building on the success of DETR [5] in object detection, query-based single-stage instance segmentation methods [7–9, 14, 20, 28] have gained prominence. These methods move away from traditional convolutional approaches, utilizing the powerful attention mechanism [49] together with learnable queries to directly predict object classes and

segmentation masks in an end-to-end fashion. However, these models typically rely on single-level features to generate queries, refining them without leveraging the full range of features available from skip connections and decoder feature maps. This limits their ability to capture the rich multi-scale context necessary for precise instance refinement.

To address these limitations, we bridge the gap between the U-Net model, widely used in biomedical imaging, and the task of instance segmentation. We present IAUNet, a novel architecture that enhances U-Net with instance-awareness through query-based mechanisms. This design incorporates a lightweight convolutional Pixel decoder, enabling the model to scale effectively with larger backbones while maintaining strong performance across both small and large datasets. IAUNet also introduces a Transformer decoder for multi-scale object feature refinement.

As part of our contributions, we introduce the 2025 Revvity Full Cell Segmentation Dataset, specifically designed for benchmarking model performance. The dataset includes hundreds of carefully annotated cell instances in high-resolution brightfield images, each thoroughly hand-labeled and validated. One of its unique features is the precise annotation of cell borders, even in cases of overlapping cells, allowing it to capture complex cell interactions. This dataset is a valuable resource for evaluating model accuracy in capturing fine details and handling challenging segmentation tasks with intricate cell morphologies.

Our main contributions are as follows:

- We introduce a lightweight Pixel-Transformer decoder within U-Net for multi-scale object feature refinement, efficiently scaling with larger backbones.
- We introduce a novel 2025 Revvity Full Cell Segmentation Dataset with detailed annotations and provide a benchmark for instance segmentation.

2. Related Work

Instance segmentation methods are generally categorized into region-based, query-based, and specialized approaches that often require preprocessing.

Region-based Methods exemplified by Mask R-CNN [17, 22, 41], have set a standard in natural image segmentation with their proposal-based structure. Building on Faster R-CNN [41], Mask R-CNN adds a mask prediction branch for end-to-end instance segmentation by first detecting bounding boxes and then applying Region of Interest (RoI) operations like RoI-Pooling [17] or RoI-Align [22] to extract features for classification and mask generation. However, these two-stage methods often generate numerous redundant region proposals, reducing efficiency [10, 20]. Although they perform well on many benchmarks, their reliance on small RoI regions frequently leads to coarse mask predictions. Some methods focus on enhancing the precision of detected bounding boxes [3], while others, like PointRend [26], specifically address low-quality segmentation masks by refining boundaries at uncertain points to improve segmentation quality. However, even with these advancements, traditional region-based methods face limitations in biomedical image segmentation [7], where objects have complex shapes, orientations, and sizes. In these settings, traditional axis-aligned bounding boxes struggle to capture detailed contours, particularly for irregular and overlapping cellular structures [15, 25].

Specialized Cell Instance Segmentation Methods like StarDist [43] segment biomedical images by representing objects as star-convex polygons, predicting distances from a central point to boundaries in multiple directions. This method, along with other similar approaches like DeepWatershed [2] and Micro-Net [39], works well for star-shaped or rounded cells but struggles with irregular, elongated shapes and overlapping cells. CellPose [48], by contrast, similar to Hover-Net [18], uses a U-Net to predict horizontal and vertical gradients alongside a binary cell map, creating a vector field that directs pixels toward the cell center. While this method effectively separates individual cells, it often relies on an additional size model [37] to estimate object diameters, which becomes challenging with varying cell sizes and shapes. Although these methods offer advancements over traditional techniques, they remain limited in accurately segmenting overlapping cells and handling complex cellular morphologies.

Query-based Methods have gained popularity since the introduction of DETR [5], which demonstrated the potential of Transformer-based architectures for instance segmentation. Unlike traditional region-based models, query-based methods use object queries to directly predict object instances, removing the need for predefined bounding boxes. Building on DETR, models like Mask2Former [9] and FastInst [20] introduced masked attention to improve convergence and segmentation precision. These models heavily rely on producing fine features using MSDeformAttn Transformer [54] Pixel decoder. MaskDINO [28] further advances instance segmentation by adding a mask prediction branch that generates high-resolution binary masks through query embeddings for unified segmentation tasks. Recently, adaptations of query-based models have also emerged in the biomedical domain. For example, Cell-DETR [38] adapts DETR specifically for cell segmentation by leveraging queries to detect individual instances. The model uses the final feature map of the encoder for query initialization, limiting multi-scale query refinement across decoder features. Its segmentation head applies multi-head attention between encoder and decoder features, followed by a CNN decoder. However, it merges queries with decoder features only at the lowest layer, forcing the CNN decoder to handle most of the instance separation. This makes the model inefficient for high-resolution inputs with many queries. Ad-

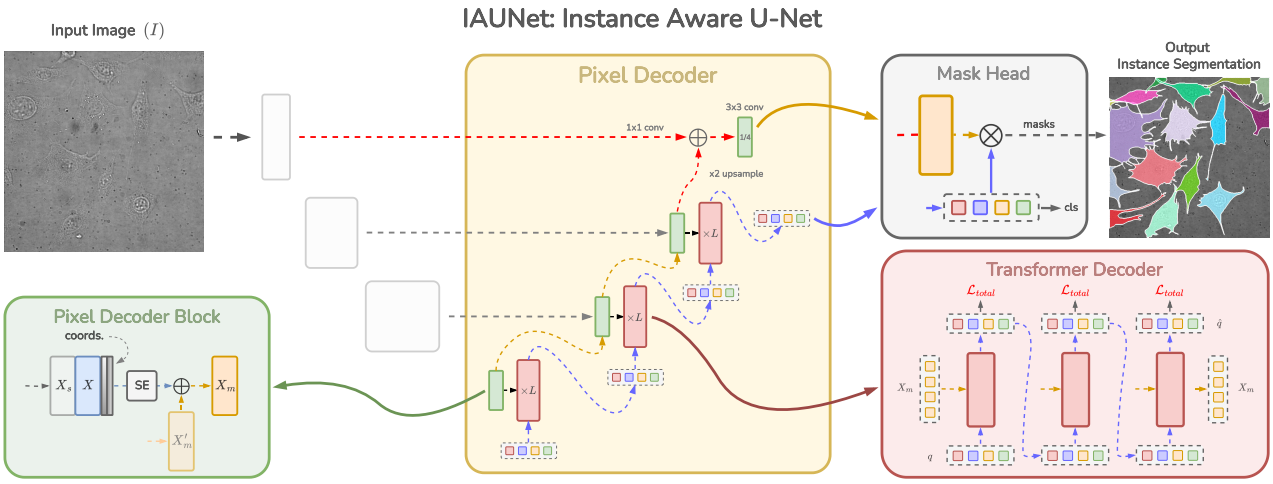

Figure 1. **Model overview.** Overview of the IAUNet architecture, highlighting the Pixel and Transformer Decoder stages. Given an input image I, the encoder extracts multi-scale features as skip connections for the Pixel decoder. At each decoder block, we add skip connections X_s to the main features X and inject normalized coordinate features for CoordConv. Stacked depth-wise convolutions with an SE block refine spatial information, generating mask features X_m. The Transformer decoder then processes learnable queries q through three Transformer blocks per layer, iteratively refining them with X_m. Deep supervision loss is applied after each Transformer block using updated queries $\hat{q}$ and high-resolution mask features.

ditionally, Cell-DETR applies softmax to suppress overlapping predictions, reducing its ability to segment occluding cells effectively. Recent work, such as PCTrans [7], built on Mask2Former, introduces a position-guided transformer with a query contrastive loss. Similar to DETR, position guidance is done by predicting the normalized center coordinates of each object. While natural objects are often convex, cells present more complex shapes, with centers that often fall outside boundaries, particularly in elongated structures [11], making mask representation less effective.

All previous query-based models [5, 7, 9, 28] have been designed around the idea of a Transformer-based Pixel decoder, which raises concerns about scalability to smaller datasets. Unlike these models, we propose a lightweight Pixel decoder that improves performance on smaller datasets. In Tab. 2, we show that IAUNet consistently outperforms state-of-the-art models across different backbones while maintaining strong results on large-scale datasets (Tab. 1). Our experiments show that IAUNet outperforms most alternatives while using fewer parameters and achieving higher efficiency.

3. Model Overview

The IAUNet model follows a U-Net design, illustrated in Fig. 1. The model consists of three main components: an encoder, a Pixel decoder, and a Transformer decoder. Given an input image $I \in \mathbb{R}^{H\times W\times 3}$, the encoder produces four multi-scale semantic feature maps at resolutions of $1/4$, $1/8$, $1/16$, and $1/32$ relative to the original image. These feature maps are utilized as skip connections in the decoder. The Pixel decoder first processes these features to generate the main decoder features X. At each decoder layer, these features pass through a lightweight mask branch to produce refined mask features, X_m, which then interact with object queries. The Transformer decoder further refines instance queries with mask features. This process is iterative, with updated queries passing through each decoder stage. In the final stage, the mask head combines mask features and instance queries to produce output instance masks.

3.1. Pixel Decoder

In the biomedical domain, U-Net [42], with all its variants [4, 6, 19, 52], still holds the ground as the most superior network for accurate segmentation. This is primarily due to the design of U-Net's decoder, which maintains high semantic consistency through the use of skip connections. We include a convolutional decoder, referred to as the Pixel decoder. Our Pixel decoder (Fig. 1, middle panel) works with two feature types: main features X and mask features X_m. The main features serve a similar role to those in the vanilla U-Net, aggregating spatial context across the image using skip connections X_s. The mask features refine X and capture richer semantic information. All these features are specifically designed to support instance segmentation and are tightly integrated with the Transformer decoder (see Sec. 3.2).

$$X = SE\Big(G_x\big([X_s, X']\big) + X'\Big) \tag{1}$$

$$X_m = G_m\Big(X'_m + X\Big) \tag{2}$$

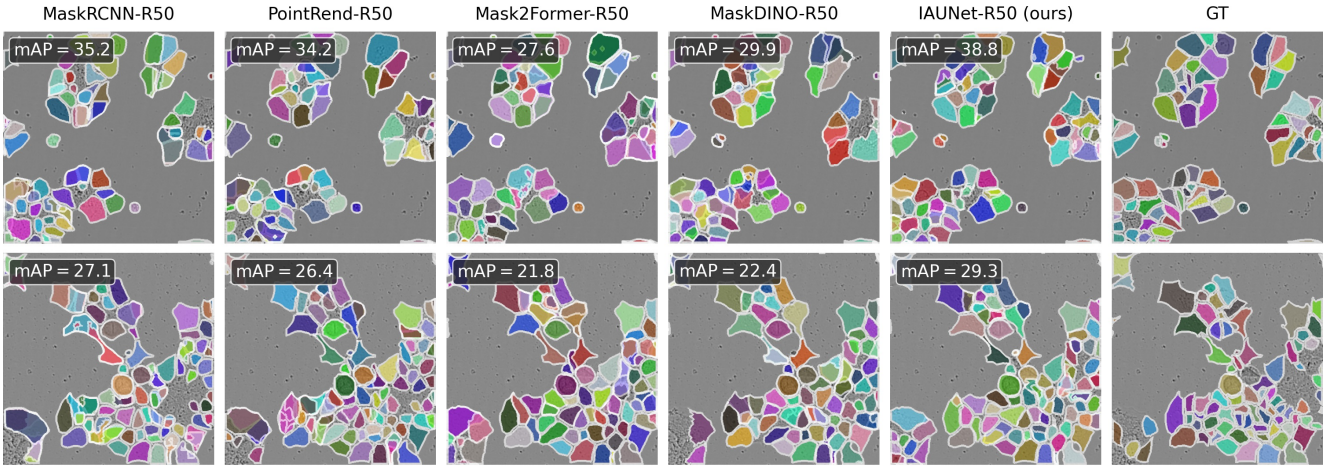

Figure 2. **LIVECell**. Visualization of instance segmentation predictions on the LIVECell dataset across different state-of-the-art models (using R50 backbone). We also report per-image AP score. Last columns shows ground-truth annotations.

At each level, the corresponding skip connection X_s is first mapped to a 256-dimensional feature map. Then it gets concatenated with the upscaled decoder features X' from the previous layer and passed through a lightweight double 3×3 point-wise convolution, batch normalization, and ReLU layer G_x (Eq. (1)). Next, we apply a Squeeze-and-Excitation (SE) [23] block to produce the final main features X. Next, we update mask features by adding the main features X and the upscaled mask features X'_m from the previous layer followed by two stacked 3×3 convolutional layers G_m Eq. (2). The whole process preserves multi-scale semantic information while maintaining a lightweight structure. The updated mask features are then used for query refinement in the corresponding Transformer blocks. Finally, we use bilinear upscaling to propagate all features to the next decoder layer.

3.2. Transformer Decoder

Object queries are central to instance segmentation [9, 12, 14, 20], serving as learnable embeddings that represent each object as a unique D-dimensional feature vector. These queries group pixel features relevant to each specific object, typically through a cross-attention mechanism. They are particularly important in Transformer architectures [5], where they are processed and refined in an end-to-end manner. In existing models such as DETR [5], Deformable DETR [53], MaskFormer [8], and Mask2Former [9], queries are central to representing objects for segmentation or detection tasks. In our work, we use N learnable queries $q \in \mathbb{R}^{N \times 256}$. Each query is thus a 256-dimensional representation, capturing the finer semantic object features. These instance queries are progressively refined with mask features X_m through a multi-layer Transformer decoder (see Sec. 3.2). At each decoder layer $l \in [1, L]$, we use three Transformer decoder layers. Queries from the previous decoder layer are iteratively processed through these layers (Fig. 1, red block) with the corresponding flattened mask features $X_m \in \mathbb{R}^{L \times 256}$, where $L = H_l \times W_l$ for the l-th decoder layer.

3.2.1. Positional Embeddings

To maintain spatial awareness, which is crucial for Transformer-based models, we add N learnable positional embeddings to both instance queries. Following the previous work [5], we add sinusoidal positional embeddings $e_{pos} \in \mathbb{R}^{H_l W_l \times D}$ to the mask X_m.

3.2.2. Instance Queries Update

We update N instance queries with the mask features X_m using the cross-attention layer (Fig. 1, red block) followed by the self-attention layer between queries and FFN layer. Thus, all queries attend to each other, ensuring better object separation. The update is expressed as follows:

$$\hat{X}_l = \text{softmax}\left(Q_l K_l^T\right) V_l + X_{l-1} \tag{3}$$

$$X_l = \text{FFN}(\hat{X}_l) \tag{4}$$

where $Q_l = f_Q(q_l) \in \mathbb{R}^{N \times 256}$ represents the transformed queries at layer l, and the keys and values $K_l, V_l \in \mathbb{R}^{H_l W_l \times 256}$ are computed from the mask features X_m. The queries are updated sequentially within Transformer blocks at each decoder layer.

3.2.3. Mask Head

To keep the prediction process lightweight without performance loss, we fuse only high-resolution features. As shown in (Fig. 1, red arrows), we construct a pixel embedding map by combining the 1/4 resolution backbone feature map X_b with an upsampled 1/8 resolution mask features X_m from the Pixel decoder. Specifically, we apply two linear projections on the refined instance queries q to

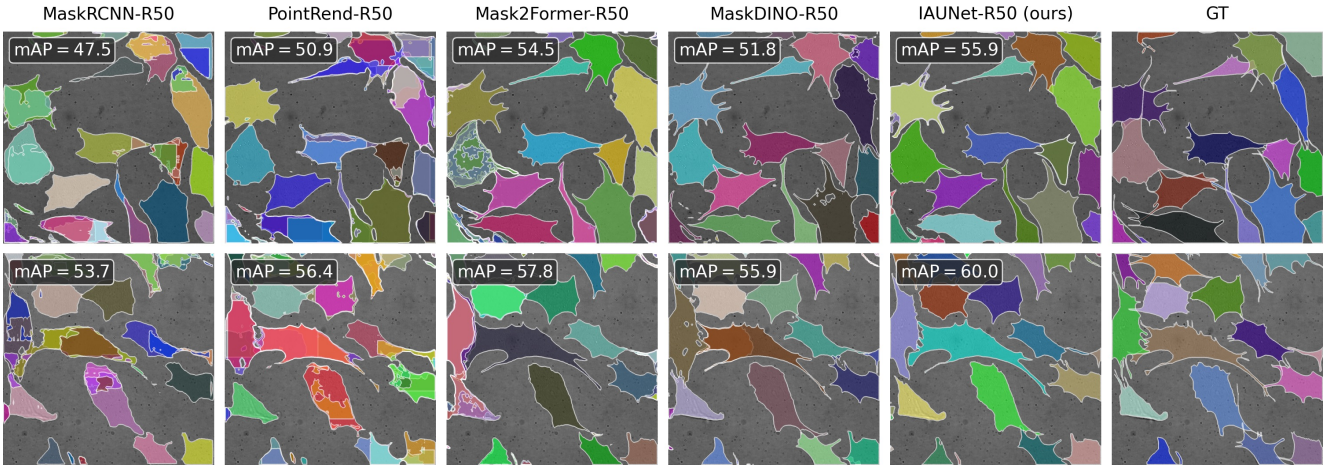

Figure 3. **Revvity-25**. Visualization of instance segmentation predictions on the Revvity-25 dataset across different state-of-the-art models (using R50 backbone). Last columns shows ground-truth annotations. IAUNet as well as MaskDINO show good generalization across tiny details and overlaping instances. We also report per-image AP score.

obtain mask embeddings q_c and object class scores. The final mask prediction is obtained by taking the dot product of each mask embedding with this fused feature map:

$$m = q_c \otimes \mathcal{M}\left(\mathcal{F}(X_b) + \mathcal{U}(X_m)\right), \tag{5}$$

where $\mathcal{M}$ is the segmentation head, $\mathcal{F}$ is a convolutional layer that adjusts the channel dimensions to match the Transformer hidden space, and $\mathcal{U}$ is a simple $2\times$ upsampling function applied to X_m. Besides, each instance query predicts the object class probability, including a "no object" ($\varnothing$). During inference, we re-score the predicted masks. For each instance, we calculate the maskness metric [9], denoted as $p_i = \frac{1}{N}\sum_{i=1}^{N} m_i$, where $m \in \{M_n\}_{n=1}^{N}$ is the predicted instance mask. The combined confidence score for each instance is then computed by multiplying the class probability score c_i with the maskness score p_i: $\hat{c}_i = c_i \cdot p_i$.

3.3. Mask Level Matching

During training, the model outputs $\{M_n\}_{n=1}^{N}$ predicted masks, where $N > M$, the number of ground truth masks $\{G_k\}_{k=1}^{M}$. To compute losses on matched predictions, we perform bipartite matching between $\{M_n\}$ and $\{G_k\}$ using the Hungarian algorithm [47], which finds the optimal permutation σ that minimizes the matching cost:

$$\sigma = \arg\min_{\sigma \in S} \sum_{i=1}^{M} \mathcal{L}_{\text{match}}(M_{\sigma(i)}, G_i). \tag{6}$$

For the matching cost, we use a combination of classification and mask costs:

$$\mathcal{L}_{\text{match}} = \lambda_{\text{cls}} \cdot \mathcal{L}_{\text{cls}} + \lambda_{\text{dice}} \cdot \mathcal{L}_{\text{dice}} + \lambda_{\text{bce}} \cdot \mathcal{L}_{\text{bce}} \tag{7}$$

Following [9] we set $\lambda_{\text{cls}} = 1.0$, $\lambda_{\text{dice}} = 2.0$, and $\lambda_{\text{bce}} = 5.0$ to control the weight of each cost term. Here, $\mathcal{L}_{\text{cls}}$ represents the cross-entropy loss for object classification, with a "no object" class weighted at 0.1. The terms $\mathcal{L}_{\text{bce}}$ and $\mathcal{L}_{\text{dice}}$ denote the binary cross-entropy loss and Dice loss, respectively, for the segmentation masks [34].

For the loss function, we align it with the matching cost by applying the same coefficients to ensure consistency. The final loss function is defined as:

$$\mathcal{L} = \lambda_{\text{cls}} \cdot \mathcal{L}_{\text{cls}} + \lambda_{\text{dice}} \cdot \mathcal{L}_{\text{dice}} + \lambda_{\text{bce}} \cdot \mathcal{L}_{\text{bce}} \tag{8}$$

4. Experiments

In this section, we evaluate our IAUNet on multiple datasets, including our novel Revvity-25 dataset. We also compare it with multiple state-of-the-art models in terms of segmentation performance. Besides, we conduct ablation studies and show the effectiveness of our model components. To provide a comprehensive comparison, we use a range of datasets:

LIVECell [13] is one of the most extensive datasets regarding images and annotated cells for instance segmentation. It consists of 5,239 high-resolution phase-contrast images (520×704 pixels) with over 1.6 million expert-validated annotated cells. It includes eight cell types with varied shapes and densities.

EVICAN2 [44] is the most heterogeneous dataset for cell segmentation, containing 5,237 microscopy images across brightfield, phase contrast, and fluorescence modalities, with 52,959 annotated cell and nucleus instances. It includes training and validation sets with 4,640 partially annotated images and a test set of 98 fully annotated images. The test set is categorized by difficulty based on image quality: easy, medium, and difficult.

ISBI2014 [33] is a dataset from the Overlapping Cervical Cytology Image Segmentation Challenge. It includes 16

			LIVECell		*EVICAN2$_E$*		*EVICAN2$_M$*		*EVICAN2$_D$*		*ISBI2014*			
Models	backbones	num_queries	AP	AP_{50}	AP	AP_{50}	AP	AP_{50}	AP	AP_{50}	AP	AP_{50}	#params.	FLOPs
Models with Convolution-Based Backbones														
Mask R-CNN [22]	R50	100	44.7	74.2	48.1	75.9	20.7	42.5	19.1	39.8	58.9	**88.7**	44M	115G
PointRend [26]	R50	100	44.0	73.5	26.6	47.9	18.0	38.5	13.4	28.3	**60.0**	**88.7**	56M	66G
Mask2Former [9]	R50	100	43.7	73.8	53.4	89.1	29.1	54.9	24.2	**50.4**	58.5	87.5	44M	67G
MaskDINO [28]	R50	100	43.3	73.5	50.7	83.9	29.3	57.9	22.0	41.9	55.4	86.8	44M	64G
IAUNet (ours)	R50	100	**45.3**	**75.3**	**58.0**	**91.8**	**32.1**	**59.0**	**24.9**	45.4	56.0	85.0	39M	49G
Mask R-CNN [22]	R101	100	44.2	73.2	41.5	69.9	23.3	46.9	17.8	36.7	**60.7**	88.8	63M	134G
PointRend [26]	R101	100	44.0	73.7	41.3	65.2	20.2	39.3	14.8	32.1	60.3	**89.2**	75M	86G
Mask2Former [9]	R101	100	44.0	73.5	54.4	87.8	27.1	51.7	20.4	42.4	59.5	88.6	63M	86G
MaskDINO [28]	R101	100	43.4	73.6	53.7	85.0	31.8	59.2	**27.1**	**51.3**	55.7	87.4	63M	84G
IAUNet (ours)	R101	100	**45.4**	**75.5**	**58.3**	**92.7**	**32.9**	**59.6**	26.9	50.0	56.5	87.1	58M	69G
Models with Transformer-Based Backbones														
Mask R-CNN [22]	Swin-S	100	44.3	73.3	52.6	91.7	27.0	59.2	20.2	50.2	61.9	90.7	69M	141G
PointRend [26]	Swin-S	100	43.9	73.5	55.1	89.2	30.1	61.6	24.4	54.6	**62.1**	**91.0**	81M	93G
Mask2Former [9]	Swin-S	100	44.6	74.3	**65.2**	**96.8**	**36.2**	66.7	**30.9**	62.7	57.1	87.3	69M	93G
MaskDINO [28]	Swin-S	100	43.9	73.8	57.0	86.9	33.6	64.9	27.6	56.9	52.7	85.3	71M	181G
MaskDINO [28]	Swin-S	300	44.8	75.1	56.5	91.8	35.0	**70.7**	30.2	**64.3**	51.2	83.4	71M	187G
IAUNet (ours)	Swin-S	100	45.4	75.4	58.8	93.1	32.2	61.9	27.7	54.1	61.1	90.1	64M	76G
IAUNet (ours)	Swin-S	300	**45.6**	**76.4**	60.9	93.6	33.2	62.0	29.6	58.0	61.8	89.8	64M	87G
Mask R-CNN [22]	Swin-B	100	44.2	73.1	52.0	89.0	26.7	60.3	24.8	55.5	62.4	**91.5**	107M	186G
PointRend [26]	Swin-B	100	44.0	73.7	58.6	91.0	34.1	64.6	25.8	52.0	62.7	**91.5**	119M	137G
Mask2Former [9]	Swin-B	100	44.9	74.7	55.0	92.5	31.4	60.9	27.7	56.6	58.1	88.4	107M	138G
MaskDINO [28]	Swin-B	100	44.3	74.1	57.3	91.1	37.3	75.7	30.1	65.6	53.5	86.6	110M	226G
MaskDINO [28]	Swin-B	300	45.2	75.8	57.9	91.6	**39.1**	**78.8**	**34.0**	**72.3**	53.3	84.8	110M	232G
IAUNet (ours)	Swin-B	100	45.5	75.6	59.6	93.5	34.2	65.7	28.9	56.9	61.5	90.8	102M	120G
IAUNet (ours)	Swin-B	300	**45.8**	**76.7**	**61.2**	**94.8**	38.0	69.6	30.7	59.9	**63.0**	**91.5**	102M	132G
Specialized Cell Segmentation Methods														
CellPose [48]		-	34.5	60.1	0.9	2.8	0.1	0.3	0.0	0.0	40.5	69.3	6.6M	163.6G
CellPose + SM [37]		-	34.9	60.4	8.7	16.8	1.6	4.4	2.3	6.8	41.6	70.4	6.6M	163.6G
CellDETR [38]	R34	100	13.9	32.7	0	0.1	0.0	0.0	0.0	0.0	0.046	0.135	57M	3.6T
IAUNet (ours)	R50	100	**45.3**	**75.3**	**58.0**	**91.8**	**32.1**	**59.0**	**24.9**	**45.4**	**56.0**	**85.0**	39M	49G
YOLO Family														
YOLOv8-M [40]		-	37.5	72.2	43.8	82.3	27.5	57.1	20.0	46.2	54.9	90.7	27.2M	110.4G
YOLOv8-L [40]		-	40.5	72.5	44.7	83.1	28.1	58.2	20.3	46.1	55.1	91.1	45.9M	220.8G
YOLOv8-X [40]		-	41.1	73.1	45.8	85.6	28.9	59.2	20.7	47.3	55.3	**91.4**	71.8M	344.5G
IAUNet (ours)	Swin-S	100	**45.4**	**75.4**	**58.8**	**93.1**	**32.2**	**61.9**	**27.7**	**54.1**	**61.1**	90.1	64M	76G
YOLOv9-E [50]		-	41.2	73.2	45.6	84.4	27.2	57.9	20.1	47.3	53.3	**91.6**	27.8M	159.1G
YOLOv9-C [50]		-	41.4	73.1	45.9	85.6	28.3	59.8	22.2	49.9	55.7	91.1	60.5M	248.1G
IAUNet (ours)	Swin-S	100	**45.4**	**75.4**	**58.8**	**93.1**	**32.2**	**61.9**	**27.7**	**54.1**	**61.1**	90.1	64M	76G
SAM Family														
SAM-B *(points)* [27]		-	5.0	12.4	28.4	56.0	5.4	13.8	3.2	7.2	33.8	51.8	90M	742G
SAM-B *(boxes)* [27]		-	24.3	56.9	55.0	**96.6**	**38.6**	**91.2**	**34.8**	**82.3**	59.6	**92.8**	90M	742G
IAUNet (ours)	Swin-S	100	**45.4**	**75.4**	**58.8**	93.1	32.2	61.9	27.7	54.1	**61.1**	90.1	64M	76G
SAM-L *(points)* [27]		-	6.3	13.6	28.1	54.1	4.9	12.4	3.2	7.5	32.8	51.0	308M	2.6T
SAM-L *(boxes)* [27]		-	29.2	65.2	57.2	**96.6**	**45.8**	**95.3**	**39.7**	**88.6**	60.8	**93.6**	308M	2.6T
IAUNet (ours)	Swin-B	300	**45.8**	**76.7**	**61.2**	94.8	38.0	69.6	30.7	59.9	**63.0**	91.5	102M	132G

Table 1. **Instance segmentation on LIVECell, EVICAN2 (Easy, Medium, Difficult), and ISBI2014.** IAUNet outperforms strong query-based Mask2Former and MaskDINO baselines for both AP and AP_{50} when training with fewer parameters. For a fair comparison, we only consider single-scale inference and models trained until full convergence. IAUNet remains efficient across different backbones.

real extended depth-of-focus (EDF) cervical cytology images and 945 synthetic images. The dataset provides high-quality pixel-level annotations for nuclei and cytoplasm, with a resolution of 512 $\times$ 512. We follow the challenge setting [33], using 45 synthetic images for training, 90 for validation, and 810 for testing.

One of our key contributions in this paper is a novel cell instance segmentation dataset named **Revvity-25**. It includes 110 high-resolution 1080 $\times$ 1080 brightfield images, each containing, on average, 27 manually labeled and expert-validated cancer cells, totaling 2937 annotated cells. To our knowledge, this is the first dataset with accurate and detailed annotations for cell borders and overlaps, with each cell annotated using an average of 60 polygon points, reaching up to 400 points for more complex structures. Revvity-25 dataset provides a unique resource that opens new possibilities for testing and benchmarking models for modal and amodal semantic and instance segmentation.

4.1. Implementation Details

All experiments were conducted on a single Tesla V100 GPU with 32GB memory. We adopt the training scheme

Revvity-25										
Models	backbones	num_queries	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	#params.	FLOPs
Models with Convolution-Based Backbones										
Mask R-CNN [22]	R50	100	39.7	77.2	37.4	0.6	19.0	44.6	44M	115G
PointRend [26]	R50	100	42.2	79.4	40.9	0.4	21.7	47.3	56M	66G
Mask2Former [9]	R50	100	46.4	79.8	49.9	0.7	25.7	52.8	44M	67G
MaskDINO [28]	R50	100	45.6	80.4	48.2	**1.8**	22.3	51.8	44M	64G
IAUNet (ours)	R50	100	**49.7**	**82.1**	**54.8**	0.6	**27.3**	**56.0**	39M	49G
Mask R-CNN [22]	R101	100	40.7	77.5	39.9	0.4	20.1	45.8	63M	134G
PointRend [26]	R101	100	42.9	79.3	42.5	0.0	18.4	48.9	75M	86G
Mask2Former [9]	R101	100	47.2	80.1	51.8	**1.7**	25.7	53.3	63M	86G
MaskDINO [28]	R101	100	47.3	81.0	50.4	0.9	23.0	53.5	63M	84G
IAUNet (ours)	R101	100	**51.5**	**84.7**	**56.1**	**1.7**	**29.2**	**57.8**	58M	69G
Models with Transformer-Based Backbones										
Mask R-CNN [22]	Swin-S	100	24.7	63.4	12.5	0.0	7.3	28.9	69M	141G
PointRend [26]	Swin-S	100	43.6	80.0	43.0	0.5	21.5	48.9	81M	93G
Mask2Former [9]	Swin-S	100	51.2	83.3	56.4	2.7	27.7	58.0	69M	93G
MaskDINO [28]	Swin-S	100	50.3	83.2	53.9	**4.7**	27.6	56.1	71M	181G
MaskDINO [28]	Swin-S	300	49.4	83.6	53.3	2.9	25.8	55.3	71M	187G
IAUNet (ours)	Swin-S	100	53.0	85.7	57.0	1.3	**29.7**	59.1	64M	76G
IAUNet (ours)	Swin-S	300	**53.3**	**86.0**	**59.6**	1.6	29.4	**59.8**	64M	87G
Mask R-CNN [22]	Swin-B	100	27.1	64.9	17.2	0.1	9.7	31.2	107M	186G
PointRend [26]	Swin-B	100	45.2	80.1	47.9	0.1	23.0	50.9	119M	137G
Mask2Former [9]	Swin-B	100	52.0	83.6	58.4	1.1	27.8	59.0	107M	138G
MaskDINO [28]	Swin-B	100	50.5	83.5	54.9	**2.0**	27.1	56.4	110M	226G
MaskDINO [28]	Swin-B	300	50.4	84.3	54.8	0.8	26.3	56.6	110M	232G
IAUNet (ours)	Swin-B	100	53.5	86.1	**59.4**	0.8	**30.5**	59.7	102M	120G
IAUNet (ours)	Swin-B	300	**53.7**	**86.5**	**59.4**	1.0	30.0	**60.3**	102M	132G

Table 2. **Instance segmentation on our Revvity-25 dataset.** IAUNet outperforms strong query-based Mask2Former and MaskDINO baselines as well as other state of the art models when training with fewer parameters. For a fair comparison, we only consider single-scale inference and models trained until full convergence. IAUNet also efficiently scales with more queries while remaining efficient.

published in earlier works [9]. We use the CosineAnnealingLR scheduler [31] with a minimum learning rate of 1e-6, and the AdamW optimizer [32] with an initial learning rate of 1e-4 and weight decay of 0.05. During training, we employ longest-side resizing to scale all images to 512 × 512 pixels, preserving the original aspect ratio. For augmentation, we apply scale jittering [16] within a scale range of 0.8 to 1.5, followed by fixed-size cropping to 512 × 512 and random flipping. All models were trained to full convergence with a batch size of 8. Unless specified, we apply the same resizing process during inference, using a consistent mask prediction threshold of 0.5 across all models.

4.2. Main Results

In this section, we outline the dataset setup for training and present the results. For the LIVECell dataset, we preprocess images by randomly cropping them to a maximum of 100 instances, ensuring consistency in prediction counts across datasets. We use the original train, validation, and test splits are used for all models. For the ISBI2014 dataset, we follow the original train, validation, and test splits. All models, except CellPose [48], are trained to segment both cell and nuclei classes. Since CellPose does not support multi-class segmentation by default, we train separate models for each class and average the performance. The Revvity-25 dataset is divided equally into train and test sets, each containing 55 images. For EVICAN2, we report results on the easy, medium, and difficult test sets. A maximum of 100 queries is set across all datasets. For example, in Tab. 1, IAUNet is compared with state-of-the-art models across diverse datasets. In models with convolution-based backbones, IAUNet with ResNet-50 achieves an AP of 45.3 and AP_{50} of 75.3 on LiveCell. It outperforms Mask R-CNN, PointRend, Mask2Former, and MaskDINO while using fewer parameters (39M) and lower FLOPs (49G). With a ResNet-101 backbone, IAUNet records an AP of 45.4 and AP_{50} of 75.5. IAUNet also scales better compared to MaskDINO when using transformer-based backbones. While IAUNet performs best on LIVECell but has room for improvement on ISBI2014, where the low object count leads to some queries predicting duplicates. Among specialized cell segmentation methods, IAUNet outperforms CellPose, CellPose + SM, and CellDETR. CellDETR, scaled to 100 objects with a softmax head on high-resolution images, has high computational cost and parameter count, making it unsuitable for some datasets. CellPose struggles to generalize when object sizes differ significantly between train and test sets, as seen in EVICAN2, due to its reliance on object diameter for post-processing.

In Fig. 3, we visualize the predictions and compute an image-wise AP score. IAUNet consistently outperforms other state-of-the-art models. IAUNet visibly offers more detailed segmentation, capturing longer pixel relationships and effectively handling overlapping regions in

Pixel Decoder	AP	AP_{50}	AP_{75}	FLOPs
+ full skip	**44.7**	**73.9**	**48.9**	146G
+ 1 × 1 skip concat	44.2	73.8	48.3	135G
+ 1 × 1 skip add	44.3	73.3	48.2	132G
+ light mask head	43.8	73.1	47.4	42G

Table 3. **Pixel Decoder Variants (Skip Connections).** We retain skip connection concatenation as in Eq. (1) and introduce a lightweight mask head.

some cases. In Tab. 2, we demonstrate IAUNet's strengths on the Revvity-25 dataset, where it achieves the highest scores across multiple backbones, with an AP of 49.7 using ResNet-50 and 53.7 with Swin-B.

4.3. Ablation Studies

In this section, we present an ablation study to evaluate the impact of each component in our model architecture. We focus on analyzing the contributions of the Pixel decoder and the Transformer decoder to overall model performance. All ablation studies were conducted on the LIVECell dataset.

Skip Connections. IAUNet builds on the U-Net architecture. Tab. 3 presents the impact of different skip connection configurations. The model performs best with full skip connections over main features X, where channels are not reduced. To balance computational efficiency, skip channels are reduced to 256 via 1×1 convolutions before fusing features using concatenation or addition. Concatenation produces optimal performance and stability, while addition creates an FPN-like [29] structure in the decoder with a further performance drop. Finally, adding a light mask head to produce high-resolution features further reduces the FLOP count to 42G without a significant performance drop.

Pixel Decoder. In Tab. 4, we study each component of the Pixel decoder separately. To further refine features in the Pixel decoder, decoupling mask features with a dedicated mask branch helps. To improve scalability, we reduce the feed-forward dimension to 1024 and add a Squeeze-and-Excitation [23] block to enhance feature representation. We observe that the model benefits from additional spatial information for multiple grouped objects of irregular shapes. Using CoordConv [30] at each level enriches the main features X before further processing, helping the model better capture object locations and improve translation awareness. This modification improves segmentation performance, increasing AP to 44.7.

Transformer Decoder. We evaluate the impact of scaling the Transformer decoder in Tab. 4. First, we introduce three Transformer decoder blocks per decoder layer, resulting in a total of $3L$ Transformer blocks. We explore two main strategies for refining object queries. The first approach, inspired by [9], follows a Round-Robin cycle update, where queries are refined in one Transformer block from each decoder layer at a time and passed to the next, forming a cycle that returns to low-resolution features. In contrast, we propose a sequential (seq.) update strategy, where object queries are refined within all decoder blocks per decoder layer first, increasing AP to 45.1. Building on this, we apply deep supervision by computing the loss after each Transformer decoder layer using the updated queries and high-resolution Pixel decoder features X_m. Additionally, in Tab. 5, we evaluate the scalability of the number of queries, showing that the model achieves peak performance as the number of queries increases.

Decoder	AP	AP_{50}	AP_{75}	#params.	FLOPs
IAUNet (R50)	43.8	73.1	47.4	34M	42G
+ mask branch X_m	44.0	73.2	47.9	34M	42G
+ FFN (2048 → 1024)	44.1	73.2	48.0	32M	42G
+ SE block [23]	44.2	73.3	48.1	32M	42G
+ CoordConv [30]	44.7	74.1	48.7	32M	42G
+ L (1 → 3) (cycle.)	44.3	74.0	48.1	39M	49G
+ L (1 → 3) (seq.)	45.1	74.4	**49.4**	39M	49G
+ deep_supervision	**45.3**	**75.3**	**49.4**	39M	49G

Table 4. **Decoder**. We investigate the benefit of adding different decoder components. Adding CoordConv [30] improves object localization. Scaling the Transformer decoder with deep supervision shows best performance.

num_queries	AP	AP_{50}	AP_{75}	FLOPs
100	45.3	75.3	49.4	49G
300	45.9	76.5	50.4	61G
500	**46.1**	**76.8**	**50.8**	73G
1000	45.3	76.3	50.0	104G

Table 5. **Num. queries**. Scaling number of object queries benefits the model.

5. Conclusions

We introduce IAUNet, a novel query-based U-Net architecture with a lightweight convolutional Pixel decoder and a Transformer decoder that supervises object-specific queries for instance segmentation in biomedical imaging. Our model outperforms leading methods, particularly for medium and large objects, and sets a strong baseline for cell segmentation tasks, as demonstrated on our Revvity-25 Dataset. While IAUNet performs well in most tasks, it struggles with small object segmentation and could benefit from optimization for high-instance images. Future work will focus on improving performance in these areas.

Acknowledgments

This work was supported by Revvity and funded by the TEM-TA101 grant "Artificial Intelligence for Smart Automation." Computational resources were provided by the High-Performance Computing Cluster at the University of Tartu. We thank the Biomedical Computer Vision Lab for their invaluable support. We express gratitude to the Armed Forces of Ukraine and the bravery of the Ukrainian people for enabling a secure working environment, without which this work would not have been possible.

References

[1] Mohammed A S Ali, Kaspar Hollo, Tõnis Laasfeld, Jane Torp, Maris-Johanna Tahk, Ago Rinken, Kaupo Palo, Leopold Parts, and Dmytro Fishman. ArtSeg—Artifact segmentation and removal in brightfield cell microscopy images without manual pixel-level annotations. *Scientific Reports*, 12(1):11404, 2022. 1

[2] Min Bai and Raquel Urtasun. Deep watershed transform for instance segmentation, 2017. 2

[3] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: High quality object detection and instance segmentation, 2019. 2

[4] Hu Cao, Yueyue Wang, Joy Chen, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, and Manning Wang. Swin-unet: Unet-like pure transformer for medical image segmentation, 2021. 3

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers, 2020. 1, 2, 3, 4

[6] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L. Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation, 2021. 3

[7] Qi Chen, Wei Huang, Xiaoyu Liu, Jiacheng Li, and Zhiwei Xiong. Pctrans: Position-guided transformer with query contrast for biological instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 3903–3912, 2023. 1, 2, 3

[8] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation, 2021. 4

[9] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[10] Tianheng Cheng, Xinggang Wang, Shaoyu Chen, Wenqiang Zhang, Qian Zhang, Chang Huang, Zhaoxiang Zhang, and Wenyu Liu. Sparse instance activation for real-time instance segmentation, 2022. 2

[11] Kevin J. Cutler, Carsen Stringer, Teresa W. Lo, Luca Rappez, Nicholas Stroustrup, S. Brook Peterson, Paul A. Wiggins, and Joseph D. Mougous. Omnipose: a high-precision morphology-independent solution for bacterial cell segmentation. *Nature Methods*, 19(11):1438–1448, 2022. 3

[12] Bin Dong, Fangao Zeng, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Solq: Segmenting objects by learning queries, 2021. 4

[13] Christoffer Edlund, Timothy R. Jackson, Nabeel Khalid, Nicola Bevan, Timothy Dale, Andreas Dengel, Sheraz Ahmed, Johan Trygg, and Rickard Sjögren. Livecell—a large-scale dataset for label-free live cell segmentation. *Nature Methods*, 18(9):1038–1045, 2021. 5

[14] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries, 2021. 1, 4

[15] Patrick Follmann and Rebecca König. Oriented boxes for accurate instance segmentation, 2020. 2

[16] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D. Cubuk, Quoc V. Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation, 2021. 7

[17] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation, 2014. 2

[18] Simon Graham, Quoc Dang Vu, Shan E Ahmed Raza, Ayesha Azam, Yee Wah Tsang, Jin Tae Kwak, and Nasir Rajpoot. Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images, 2019. 2

[19] Ali Hatamizadeh, Yucheng Tang, Vishwesh Nath, Dong Yang, Andriy Myronenko, Bennett Landman, Holger Roth, and Daguang Xu. Unetr: Transformers for 3d medical image segmentation, 2021. 3

[20] Junjie He, Pengyu Li, Yifeng Geng, and Xuansong Xie. Fastinst: A simple query-based model for real-time instance segmentation, 2023. 1, 2, 4

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, 2015. 1

[22] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2018. 1, 2, 6, 7

[23] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks, 2019. 4, 8

[24] Armen R Kherlopian, Ting Song, Qi Duan, Mathew A Neimark, Ming J Po, John K Gohagan, and Andrew F Laine. A review of imaging techniques for systems biology. *BMC systems biology*, 2:1–18, 2008. 1

[25] Alexander Kirillov, Evgeny Levinkov, Bjoern Andres, Bogdan Savchynskyy, and Carsten Rother. Instancecut: from edges to instances with multicut, 2016. 2

[26] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering, 2020. 2, 6, 7

[27] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023. 6

[28] Feng Li, Hao Zhang, Huaizhe xu, Shilong Liu, Lei Zhang, Lionel M. Ni, and Heung-Yeung Shum. Mask dino: Towards a unified transformer-based framework for object detection and segmentation, 2022. 1, 2, 3, 6, 7

[29] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection, 2017. 8

[30] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution, 2018. 8

[31] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts, 2017. 7

[32] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. 7

[33] Zhi Lu, Gustavo Carneiro, and Andrew P. Bradley. An improved joint optimization of multiple level set functions for

the segmentation of overlapping cervical cells. *IEEE Transactions on Image Processing*, 24(4):1261–1272, 2015. 5, 6

[34] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation, 2016. 5

[35] Erick Moen, Dylan Bannon, Takamasa Kudo, William Graf, Markus Covert, and David Van Valen. Deep learning for cellular image analysis. *Nature Methods*, 16(12):1233–1246, 2019. 1

[36] Larry E. Morrison, Mark R. Lefever, Lauren J. Behman, Torsten Leibold, Esteban A. Roberts, Uwe B. Horchner, and Daniel R. Bauer. Brightfield multiplex immunohistochemistry with multispectral imaging. *Laboratory Investigation*, 100(8):1124–1136, 2020. 1

[37] Marius Pachitariu and Carsen Stringer. Cellpose 2.0: how to train your own model. *Nature Methods*, 19(12):1634–1641, 2022. 2, 6

[38] Tim Prangemeier, Christoph Reich, and Heinz Koeppl. Attention-based transformers for instance segmentation of cells in microstructures. In *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2020. 2, 6

[39] Shan E Ahmed Raza, Linda Cheung, Muhammad Shaban, Simon Graham, David Epstein, Stella Pelengaris, Michael Khan, and Nasir M. Rajpoot. Micro-net: A unified model for segmentation of various objects in microscopy images. *Medical Image Analysis*, 52:160–173, 2019. 2

[40] Dillon Reis, Jordan Kupec, Jacqueline Hong, and Ahmad Daoudi. Real-time flying object detection with yolov8, 2024. 1, 6

[41] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2016. 2

[42] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 1, 3

[43] Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. *Cell Detection with Star-Convex Polygons*, page 265–273. Springer International Publishing, 2018. 2

[44] Mischa Schwendy, Ronald E Unger, and Sapun H Parekh. EVICAN—a balanced dataset for algorithm development in cell and nucleus segmentation. *Bioinformatics*, 36(12): 3863–3870, 2020. 5

[45] Jyrki Selinummi, Pekka Ruusuvuori, Irina Podolsky, Adrian Ozinsky, Elizabeth Gold, Olli Yli-Harja, Alan Aderem, and Ilya Shmulevich. Bright field microscopy as an alternative to whole cell fluorescence in automated analysis of macrophage images. *PLoS One*, 4(10):e7497, 2009. 1

[46] Prem Shrestha, Nicholas Kuang, and Ji Yu. Efficient end-to-end learning for cell segmentation with machine generated weak annotations. *Communications Biology*, 6(1):232, 2023. 1

[47] Russell Stewart and Mykhaylo Andriluka. End-to-end people detection in crowded scenes, 2015. 5

[48] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature Methods*, 18(1):100–106, 2021. 2, 6, 7

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. 1

[50] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao. Yolov9: Learning what you want to learn using programmable gradient information, 2024. 1, 6

[51] Gufeng Wang and Ning Fang. Detecting and tracking nonfluorescent nanoparticle probes in live cells. *Methods Enzymol*, 504:83–108, 2012. 1

[52] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation, 2018. 1, 3

[53] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection, 2021. 4

[54] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection, 2021. 2