Biomedical Signal Processing and Control 104 (2025) 107537

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

3DCellPol: Joint detection and pairing of cell structures to compute cell polarity

Hemaxi Narotamo [a,b,c], Cláudio A. Franco [b,c,*], Margarida Silveira [a,**]

[a] Institute for Systems and Robotics (ISR/IST), LARSyS, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, 1049-001, Portugal
[b] Instituto de Medicina Molecular - João Lobo Antunes, Faculdade de Medicina, Universidade de Lisboa, Lisbon, 1649-028, Portugal
[c] Universidade Católica Portuguesa, Católica Medical School, Católica Biomedical Research Centre, Lisbon, 2780-156, Portugal

ARTICLE INFO

Keywords:
Deep learning
Generative adversarial networks
3D fluorescence microscopy images
Cell polarity vectors
Endothelial cell front-rear polarity

ABSTRACT

Cell polarity is essential for tissue structure and cell migration, and its dysregulation is linked to diseases such as cancer and vascular disorders. Understanding the associations between molecular mechanisms, such as genetic defects, and abnormal cell polarization can provide clinicians with valuable biomarkers for early disease diagnosis and lead to more targeted therapeutic interventions.

Here, we present a deep-learning framework for cell polarity computation based on the association between pairs of objects. Our approach, named 3DCellPol, is trained to detect and group the centroids of two distinct objects. To demonstrate the potential of 3DCellPol, we use it to compute cell polarity by pairing two cell organelles: nuclei and Golgi. The vectors between nuclei and Golgi define the front-rear polarity axis in endothelial cells. 3DCellPol was evaluated on 3D microscopy images of mouse retinas. It detected 71% of the nucleus–Golgi vectors and outperformed previous methods while requiring much less supervision. Moreover, incorporating synthetic data generated by a generative adversarial network further improved detection to 78%.

We additionally demonstrated our model's adaptability to 2D images by applying it to a public dataset of cervical cytology images, where polarity is defined based on the cytoplasm-nucleus vectors. In this dataset, our model detected over 90% of vectors.

3DCellPol's ability to robustly compute cell polarity is crucial for understanding mechanisms of diseases where abnormal polarity plays a key role, and it may contribute to improved diagnostics and enable targeted therapies. Hence, it is a valuable open-source tool for both biomedical research and clinical practice.

1. Introduction

Various image analysis tasks involve object detection and grouping, where objects are identified individually and grouped if they belong to the same unit. For example, in instance segmentation, the goal is to detect and group foreground pixels representing the same entity [1]. In multi-person pose estimation, human body joints (e.g. heads, elbows, knees, wrists) are identified and grouped to form a skeleton for each person [2]. Similarly, in face detection and recognition, facial landmarks are detected and grouped to identify individuals [3]. These tasks have valuable applications. For example, segmentation is helpful for autonomous driving [1], while multi-person pose detection helps analyzing human activities in monitored areas [2]. However, these tasks become challenging in crowded scenes with overlapping, touching, and occluded objects [1,2].

In biological systems, understanding the interactions between subcellular structures, such as mitochondria–Golgi, mitochondria–vacuole, and endoplasmic reticulum–Golgi, is crucial, as their dysregulation can lead to conditions like neurodegenerative diseases, cancer and obesity [4]. Moreover, cytogeneticists detect and group nuclei and fluorescent in-situ hybridization (FISH) spots in microscopy images to detect chromosomal defects and diagnose genetic diseases [5]. Similar to object detection and segmentation in natural images, analyzing biological images is challenging because overlapping structures make the identification of individual elements and their grouping into distinct units extremely difficult [6].

Within this context, cell polarity analysis is of great clinical value. Cell polarity is the asymmetrical distribution of organelles, proteins, and cytoskeleton, which can establish several axes of polarity in the cell, among which the front-rear and apical-basal axes are the most

---

* Correspondence to: Universidade Católica Portuguesa, Católica Biomedical Research Centre, Rua da Quinta Grande 6, 2780-156 Oeiras, Portugal.
** Corresponding author.
E-mail addresses: hemaxi.narotamo@tecnico.ulisboa.pt (H. Narotamo), cfranco@ucp.pt (C.A. Franco), msilveira@isr.tecnico.ulisboa.pt (M. Silveira).

https://doi.org/10.1016/j.bspc.2025.107537
Received 26 August 2024; Received in revised form 20 December 2024; Accepted 17 January 2025

Abbreviations

CM	Cost Matrix
CNN	Convolutional Neural Network
CS	Cosine Similarity
EC	Endothelial Cell
EP	Endpoint of a Vector
FISH	Fluorescent In-Situ Hybridization
FN	False Negative
FP	False Positive
FDR	False Discovery Rate
GAN	Generative Adversarial Network
GC	Golgi Centroid
GD	Golgi Centroid Distance
GFP	Green Fluorescent Protein
GV	Golgi Volume
HS	Hierarchical Segmentation
IoU	Intersection Over Union
IP	Initial Point of a Vector
MAF	Major Axis Fraction
MSE	Mean Squared Error
NC	Nucleus Centroid
ND	Nucleus Centroid Distance
PM	Pairing Matrix
ReLU	Rectified Linear Unit
SD	Standard Deviation
SSIM	Structural Similarity Index Measure
TP	True Positive
TPR	True Positive Rate
2D	Two-Dimensional
3D	Three-Dimensional

Notation

eg	Golgi embedding map
en	nuclei embedding map
hg	Golgi heatmap
hn	nuclei heatmap
V	3D vector
(Vx, Vy, Vz)	components of vector V
W, H, D	width, height, depth
τ_g	Golgi threshold
τ_L	Threshold level
τ_n	Nucleus threshold (in the nucleus–Golgi datasets)
τ_c	Cytoplasm threshold
τ'_n	Nucleus threshold (in the cytoplasm-nucleus dataset)
$\hat{}$	Denotes a predicted quantity

relevant [7,8]. It plays a crucial role in physiological processes, such as cell migration, proliferation, differentiation, and protein trafficking [7,9–11]. Disruptions in cell polarity are linked to pathological conditions, including cancer [7], cardiovascular disorders [9,11], and chronic diseases [11]. Unraveling the mechanisms underlying cell polarization is fundamental for advancing knowledge of developmental, physiological, and pathological processes, as well as for creating novel diagnostic and therapeutic approaches [7].

Several pairs of biological structures have been used to compute cell polarity: nucleus–centrosome, cytoplasm–nucleus, nucleus–Golgi. The nucleus–centrosome axis orientation has been extensively explored in various works [12]. For instance, in [13], it was used to study epithelial cell polarity under different extracellular matrix geometries to understand how polarity is influenced by the extracellular environment. The displacement of the nucleus relative to the cytoplasm centroid can be used to study cell polarity and migration because cells frequently migrate with the nucleus positioned close to the rear end [14]. Furthermore, the nucleus positioning within the cell is crucial for investigating cell division [14] and pathological disorders, such as cancer [15].

The nucleus–Golgi pairs are particularly useful to study the polarization of endothelial cells (ECs), the cells lining the interior of blood vessels. In this system, front-rear polarity is calculated as a vector from the nucleus centroid to the corresponding Golgi complex centroid [10,16] (Fig. 1A). ECs polarize and migrate towards gradients of vascular endothelial growth factor A (VEGFA) but also against the blood flow direction during vessel remodeling [9,17,18]. Several vascular pathologies are caused by defects in vascular remodeling, such as hypertension, diabetic retinopathy and arteriovenous malformations [9]. Hence, vascular biologists are interested in unraveling the molecular and cellular mechanisms associated to these diseases which will be crucial to develop innovative treatment strategies [9].

The main challenge in understanding these mechanisms is that *in vivo* front-rear cell polarity is usually computed manually by assigning nuclei and Golgi centroids on 2D projections of 3D images [19], a time-consuming process dependent on the user's experience and often leading to fatigue. To address this, automated methods have been proposed, typically following a multi-step strategy that starts with nuclei and Golgi segmentation, followed by 3D centroid computation, and ends with centroid pairing. However, these approaches rely on ground-truth 3D segmentation masks for training, requiring laborious manual voxel labeling [20].

To overcome this issue, we propose 3DCellPol, a deep learning method for cell polarity computation that requires less training supervision compared to previous approaches. Specifically, our approach only requires paired centroids for training and it jointly predicts and groups the centroids of cell organelles. We trained and tested 3DCellPol using 3D microscopy images of mouse retinas that express fluorescent markers for endothelial nuclei and Golgi apparatuses. An example of a subvolume extracted from one of these images is shown in Fig. 1B.

Our approach predicts centroid heatmaps and embedding maps separately for nuclei and Golgi, as illustrated in Fig. 2. First, the centroids are obtained by extracting the local maximizers from the centroid heatmaps. The embedding for each detected centroid is obtained by extracting the value, at the centroid's position, from the embedding heatmap, represented by vertical dashed gray arrows in Fig. 2. Pairing is then performed based on the distance between a nucleus embedding and a Golgi embedding (Fig. 2). Nuclei and Golgi with the closest embeddings are paired, as long as the distances between their centroids and between their embeddings are smaller than certain thresholds. Compared to our previous method [20], 3DCellPol eliminates the need for 3D segmentation ground-truth by relying on centroid annotations, thereby significantly reducing the manual annotation effort.

3DCellPol will be used by vascular biologists to understand the mechanisms controlling EC polarization and migration during blood vessel formation and remodeling in development, healthy and pathological conditions. Hence, our tool will be helpful to develop therapeutic targets for vascular diseases.

Moreover, we train and test a 2D version of 3DCellPol, which we named 2DCellPol, on a dataset of cervical cytology images. In this dataset, we use 2DCellPol to predict cytoplasm-nucleus vectors. In this way, we show that our tool can be useful for cytotechnologists who analyze cervical cytology images for early cervical cancer detection [21]. In this case, our tool has potential for direct clinical application, as the dataset comprises samples from patients with both negative and positive cervical cancer diagnoses [21,22].

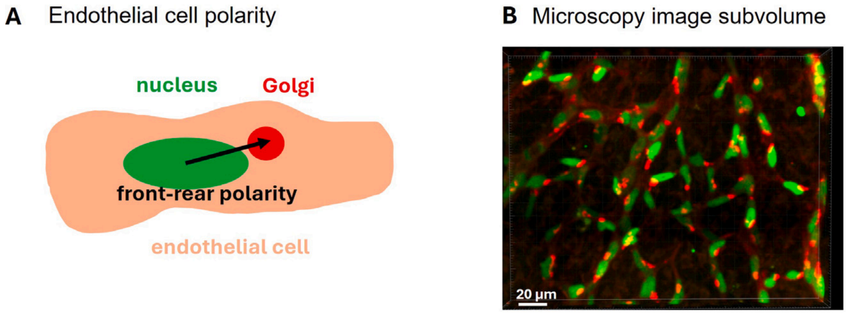

Fig. 1. EC nuclei and Golgi. (A) Schematic representation of an EC (beige), its nucleus (green), its Golgi (red), and the nucleus–Golgi vector (black) representing the front-rear polarity of the cell. (B) Example of a 3D microscopy image subvolume obtained from the dataset used in this work, where the nuclei and Golgi are represented in the green and red channels, respectively.

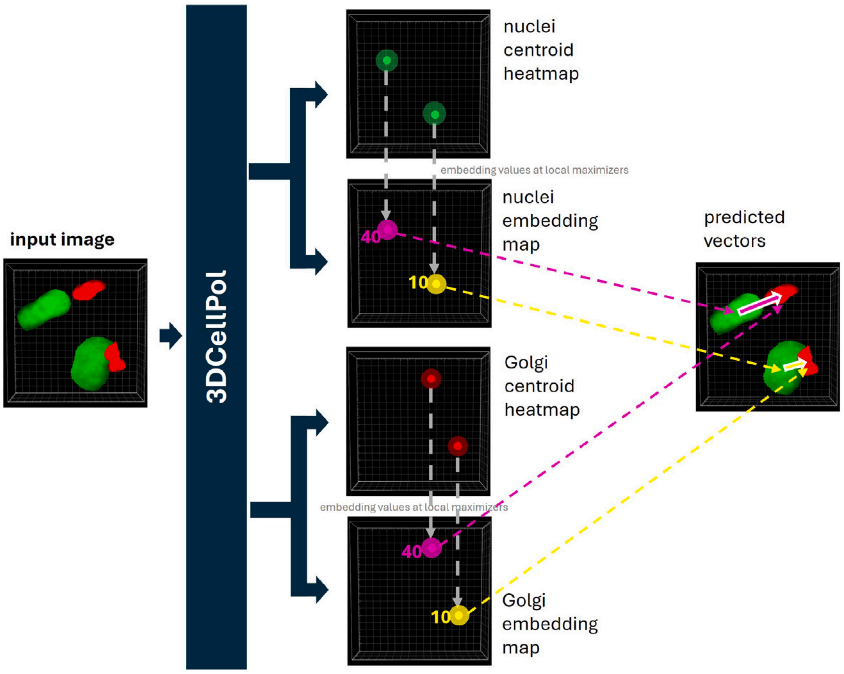

Fig. 2. Schematic representation of 3DCellPol. It receives an image with nuclei and Golgi as input, and outputs the centroid heatmaps and embedding maps of nuclei and Golgi. The centroids are decoded by extracting the local maximizers from the centroid heatmaps. The corresponding embeddings are obtained from the embedding maps at the centroids' positions (vertical dashed gray arrows). A nucleus and a Golgi belonging to the same cell should have similar embeddings. Thus, the distances between the embeddings of nuclei and Golgi are used to compute the vectors.

In conclusion, 3DCellPol can be easily adapted to compute cell polarity in other tissues, because it only requires images and paired centroids for training. We believe that 3DCellPol will mostly be a valuable tool for translational research, for instance to unravel key indicators of diseases caused by abnormal cell polarity, such as cancer. In cancer, metastasis is caused by cells that lose epithelial polarity and develop front-to-back polarity which is essential to invade other tissues [7]. 3DCellPol will allow researchers to study the molecular mechanisms behind this polarity transition, identifying genetic mutations and signaling pathways that lead to incorrect polarization. Understanding these associations between biological processes and abnormal cell polarization can provide clinicians with useful biomarkers for early cancer diagnosis and, potentially, lead to more targeted therapeutic interventions.

2. Related work

2.1. 3D nucleus-Golgi vector computation

The first automated approach for computing nucleus–Golgi vectors, proposed by Barbacena et al. [23], is a multi-step approach. First, segmentation of nuclei and Golgi is performed. Then, 3D centroid is computed for each object and an assignment matrix with the distances between nuclei and Golgi centroids is built. The nucleus–Golgi pairs are obtained by minimizing the cost of this matrix. Barbacena et al. [23] applied a classical thresholding method for segmentation, whereas we used the 3D U-Net deep learning model [20,24]. This model [24] and its variants [25,26] have been widely used for cell nuclei segmentation in 3D microscopy images. In these works [20,24–26], the models are applied for semantic segmentation, where each voxel is classified as either background or nucleus. Hence, these architectures are unable to segment each nucleus individually, a task known as instance segmentation.

StarDist-3D is a widely used model for 3D nuclei instance segmentation [27]. It represents nuclei as star-convex polyhedra, approximating their roundish shape. For each voxel, StarDist-3D predicts distances to the object boundary along K radial directions and estimates the probability that the voxel belongs to a nucleus. Only voxels with an object probability above a threshold are considered, and non-maximum suppression removes redundant shapes representing the same object.

However, whether they are semantic or instance segmentation methods [24–27], these approaches need ground-truth masks for training, which are difficult to obtain since they require a laborious manual annotation step. Moreover, the multi-step approach proposed by Barbacena et al. [23] performs pairing based only on the Euclidean distances between nuclei and Golgi centroids.

In our previous work [20], we proposed a deep learning method for joint segmentation and pairing of nuclei and Golgi, called 3D U-Net HS (Hierarchical Segmentation). In this approach, a first 3D U-Net predicts joint segmentation masks of nuclei and Golgi, representing nucleus–Golgi pairs. The output of this 3D U-Net and the original 3D image are multiplied, and the result is fed into a second 3D U-Net to generate separate segmentation masks for nuclei and Golgi. Finally, for each connected component in the first 3D U-Net's output, the corresponding nucleus and Golgi centroids are computed from the second 3D U-Net's outputs. The polarity vectors are then computed using the nucleus and Golgi centroids as the initial point (IP) and the endpoint (EP), respectively. The first 3D U-Net pairs nuclei and Golgi based on automatically extracted features, not just centroid distances, but still requires 3D ground-truth segmentation masks for training.

2.2. Keypoint detection and grouping

In several imaging tasks, such as object localization, detection and counting, the need for extensive supervision can be reduced by using alternative approaches that rely on less costly annotations, such as object keypoints. These keypoints can be centroids of objects [28] or anatomical features such as landmarks in X-ray images of hands [29] or human body joints [30]. The keypoints based approach has been applied in the microscopy and histopathology imaging fields for the detection of cell/nuclei centroids in 2D [31–33] and 3D [34–36]. In these works [31–36], centroid heatmaps were built based on different functions such as exponential-based, Gaussian, and Gaussian-based functions, and different convolutional neural networks (CNNs) were explored to predict the heatmaps. One common aspect among the generated heatmaps is that they have peaks at the locations of the ground-truth centroids.

Keypoint detection has also been combined with grouping using a single network. For instance, in [37] for detection and pairing of human body joints to estimate multi-person pose, and in [38] for bounding boxes top-left and bottom-right corners joint detection and pairing to detect objects. These approaches consist of training a network to predict keypoint heatmaps and embeddings. The embeddings should be similar for keypoints belonging to the same unit, and dissimilar otherwise. This embedding learning process using CNNs was first proposed by Harley et al. [39] for semantic segmentation. They trained a CNN to predict pixel embeddings, so pixels belonging to the same class should present similar embeddings.

2.3. Generation of synthetic data

Recently, methods based on generative adversarial networks (GANs) have successfully been used to generate synthetic images to overcome the requirement of large manually annotated datasets [40]. There are several works in which the Pix2Pix model [41], one popular extension of GAN, was used for this purpose due to its capacity of conditional translation of one image into another [42]. For instance, the Pix2Pix model was used to generate synthetic datasets of retinal and neuronal images [43] and a dataset of images corresponding to different skin conditions [44]. In these works [43,44], the synthetic datasets generated using the Pix2Pix model were used as a data augmentation method to train a segmentation model [43] and a classifier for skin condition [44].

Table 1
Major and minor axes (Mean ± SD) of nuclei and Golgi computed based on the ground-truth segmentation masks.

	Nuclei	Golgi
Major axis (μm)	15.86 ± 6.55	7.91 ± 3.58
Minor axis (μm)	8.50 ± 3.46	4.85 ± 1.49

3. Contributions

Our proposed 3DCellPol model was inspired by keypoint detection and grouping approaches [37,38]. Additionally, we studied the benefits of using synthetic images for 3DCellPol training. These images were generated using the Vox2Vox model, which is the 3D version of the Pix2Pix model [41]. This work presents three main contributions:

- 3DCellPol, a neural network that simultaneously detects and pairs organelles in 3D microscopy images using a single network. It is simpler than multi-stage approaches (segmentation, centroid computation and centroid pairing) and does not require manually annotated segmentation masks for training. Instead, it only requires pairs of centroid annotations, which can be obtained with significantly less annotation effort.
- A GAN to generate synthetic microscopy images of nuclei and Golgi based on automatically created polarity vectors. We use a combination of real and synthetic images for training 3DCellPol.
- A 2D version of 3DCellPol, denoted as 2DCellPol. We train and evaluate 2DCellPol on a publicly available dataset for cytoplasm-nucleus vector prediction [21], demonstrating its performance on a different polarity prediction task and in 2D.

4. Materials and methods

This section presents the datasets used in this work, the proposed approach, the state-of-the-art methods considered for performance comparison, the evaluation methodology and the experiments performed.

4.1. Datasets

4.1.1. Real nucleus-Golgi dataset

Our primary dataset consists of three 3D fluorescence microscopy images of mouse retinas and corresponding ground-truth nucleus–Golgi vectors (Fig. 3A,C). From these three images, 261 smaller subvolumes of size 128 × 128 × 64 × 2 were obtained. Segmentation ground-truth is required for training the state-of-the-art approaches implemented for performance comparison (Fig. 3B). Both the vectors and the masks were obtained from a manual annotation process performed by an expert.[1] The dataset contains 482 nucleus–Golgi pairs.[2] The resolution of these images across x, y and z is 0.666 μm, 0.666 μm and 0.270 μm, respectively. We computed the mean major and minor axes of nuclei and Golgi based on the ground-truth masks, the values are represented in Table 1.

All the retinas are from the GNrep mice [23]. Stainings were performed according to previously published protocols [45]. This study was conducted in accordance with European Union (EU) regulations and ethical approval was obtained from the Animal Ethics Committee of Instituto de Medicina Molecular (AWB_2015_10_CF_Polaridade).

---

[1] 3D nucleus–Golgi vectors can be annotated using the tool available at https://github.com/HemaxiN/VectorAnnotationTool.

[2] Our real dataset is publicly available at https://huggingface.co/datasets/Hemaxi/3DNGPol.

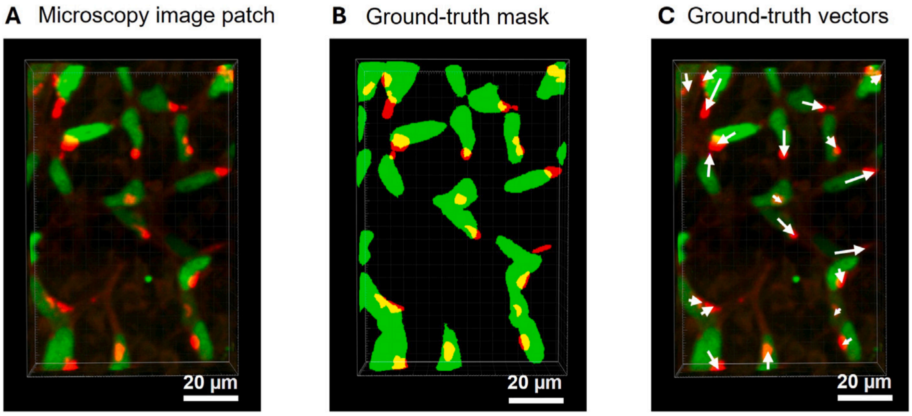

Fig. 3. Samples from the real nucleus–Golgi dataset. (A) Example of a 3D microscopy image subvolume, the green and the red channels represent the nuclei and the Golgi, respectively; (B) corresponding manually annotated segmentation mask; (C) and corresponding manually annotated ground-truth vectors (represented in white).

Table 2
Major and minor axes (Mean ± SD) of cytoplasm and nuclei, in the cytoplasm-nucleus dataset [21], computed based on the ground-truth segmentation masks.

	Cytoplasm	Nuclei
Major Axis (pixels)	95.87 ± 36.36	16.87 ± 6.60
Minor Axis (pixels)	64.83 ± 27.29	12.69 ± 5.22

4.1.2. Synthetic nucleus-Golgi dataset

In this work, we generated a synthetic dataset containing automatically created nucleus–Golgi vectors and corresponding images obtained with a GAN.[3] First, the GAN is trained using the real nucleus–Golgi dataset. Then, we use the trained model to generate synthetic subvolumes that are equal in number to the patches in the real nucleus–Golgi dataset. These synthetic subvolumes and the corresponding vectors are used only for 3DCellPol training, whereas testing is always performed on the real nucleus–Golgi dataset.

4.1.3. Cytoplasm-nucleus dataset

We also considered a 2D dataset to demonstrate our model's performance on a different vector polarity task and in 2D. We used the Cx22 dataset [21], here denoted as cytoplasm-nucleus dataset, which contains cervical cytology images of size 512 × 512 × 3, with constant padding of variable size using a value of 255 (white pixel padding) [21] (Fig. 4A). The dataset also contains the corresponding instance segmentation masks of cytoplasm and nuclei (Fig. 4B). We used these masks to pair cytoplasm and nucleus within the same cellular instance. Therefore, based on the centroids of the instance segmentation masks, we automatically created the ground-truth cytoplasm-nucleus vectors (Fig. 4C). Additionally, we computed the mean major and minor axes of cytoplasm and nuclei using the ground-truth masks (Table 2). These values are expressed in pixels because the image resolution information is not provided. This dataset includes 400 training and 100 test images.

4.2. Proposed approach

Herein, we present the proposed approach to compute cell polarity in 3D microscopy images and a method to generate synthetic training images.

4.2.1. 3DCellPol: Cell polarity vectors prediction

Overview 3DCellPol consists of a 3D U-Net backbone and four output branches (Fig. 5 and Sup. Fig. 1A). It takes as input a 3D microscopy image I (Fig. 5(1)) and outputs four maps, the centroid heatmaps of nuclei and Golgi ($\hat{hn}$ and $\hat{hg}$) and the corresponding embedding maps (en and eg) (Fig. 5(2),(3)). A detailed explanation of 3DCellPol's architecture is provided as Supplementary Information.

3DCellPol's training data consists of images and paired nucleus–Golgi centroid annotations. We generate ground-truth centroid heatmaps by placing a 3D normalized Gaussian kernel at each ground-truth centroid point.

Following Newell et al. [37] and Law et al. [38], the network is trained to predict similar embeddings for keypoints from the same unit, in our case, for a nucleus and a Golgi from the same cell. However, in the loss function, instead of considering the embeddings only at the ground-truth keypoint positions, as done in [37,38], we consider embeddings in a neighborhood around the ground-truth centroid.

During inference, the nuclei/Golgi centroids are obtained from the heatmaps by computing the local maximizers. Then, nuclei/Golgi embeddings at those positions are extracted from their embedding maps (Fig. 5(4,5)). Finally, nuclei and Golgi are paired based on the similarity between their embeddings (Fig. 5(6)).

The 3DCellPol model is trained to minimize the following loss function:

$$L = \alpha \times \left[L_{heatmap}(hg, \hat{hg}) + L_{heatmap}(hn, \hat{hn})\right] + \beta \times \left[L_{pull} + L_{push}\right], \qquad (1)$$

where $L_{heatmap}$ is a variant of the mean squared error (MSE) loss (Eq. (2)), and L_{pull} and L_{push} are the embedding pull and push losses (Eqs. (3) and (4)). α and β are weights assigned to each of the loss terms. The proposed variant of the MSE loss (Eq. (2)) is the MSE weighted by the ground-truth centroid probability map, h, to give more weight to voxels closer to the center rather than assigning equal weights to all voxels. We assign a small constant weight to the background voxels.

$$L_{heatmap}(h, \hat{h}) = \frac{1}{M}\sum_{i=1}^{M}(h_i - \hat{h}_i)^2 \times \begin{cases} w_f \times h_i & \text{if } h_i \neq 0 \\ w_b & \text{if } h_i = 0 \end{cases} \qquad (2)$$

In Eq. (2), h and $\hat{h}$ denote the ground-truth and predicted heatmaps, respectively. M is the total number of voxels in h, i represents the ith voxel, and w_f and w_b are the foreground and background weights, chosen such that $w_b < w_f$. In this way, the smallest weights are applied to voxels with a zero value in the ground-truth heatmaps because there are more zero voxels than non-zero voxels. Moreover, due to the multiplication by h_i, the highest weights are applied at the ground-truth centroid positions, with the weights for other voxels gradually decreasing as their distance from the ground-truth centroid increases.

---

[3] Our synthetic dataset is publicly available at https://huggingface.co/datasets/Hemaxi/3DNGPol.

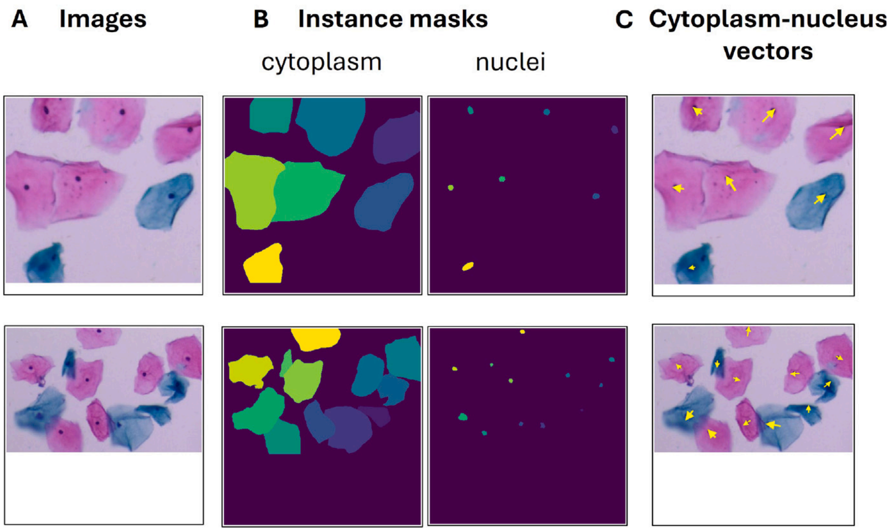

Fig. 4. Training samples from the cytoplasm-nucleus dataset [21]. (A) Examples of training images of size 512 × 512 × 3 with white pixel padding; (B) corresponding instance segmentation masks of cytoplasm (on the left) and nuclei (on the right), the colors are used to represent different instances; (C) corresponding ground-truth cytoplasm-nucleus vectors represented in yellow on top of the training images.

Examples of 2D projections of nuclei and Golgi weight maps are shown in Fig. 6(7),(8).

The embedding loss terms are used to learn similar embeddings for nuclei and Golgi of the same cell and dissimilar for nuclei and Golgi belonging to different cells. Specifically, the network learns to group nuclei and Golgi that belong to the same cell as a result of the pull loss:

$$L_{pull} = \frac{1}{L}\sum_{k=1}^{L}\frac{1}{G+N}\left(\sum_{j=1}^{G}\left[eg(a_j)-e_k\right]^2+\sum_{l=1}^{N}\left[en(b_l)-e_k\right]^2\right) \qquad (3)$$

where L is the total number of ground-truth nucleus–Golgi pairs in a patch, $eg(a_j)$ and $en(b_l)$ are the Golgi embeddings at positions a_j and nuclei embedding at positions b_l. $a_j = \left\{a_1, a_2, \dots, a_G\right\}$ denotes the positions of the voxels in the neighborhood $\mathcal{N}_{\mathcal{G}}$ of the kth ground-truth Golgi keypoint, and $b_l = \left\{b_1, b_2, \dots, b_N\right\}$ denotes the positions of the voxels in the neighborhood $\mathcal{N}_{\mathcal{N}}$ of the kth ground-truth nucleus keypoint. For instance, in the example illustrated in Fig. 6, L = 4, $k = \{1, 2, 3, 4\}$, and a_j and b_l denote the positions of the non-zero voxels in the embedding maps of Golgi and nuclei, respectively. These voxels are shown in different colors in the embedding maps represented in Fig. 6(5),(6), with each color corresponding to a specific k value.

Finally, e_k is the mean predicted embedding value for the kth ground-truth nucleus–Golgi pair, which is computed as: $e_k = \frac{1}{G+N}(\sum_{j=1}^{G} eg(a_j) + \sum_{l=1}^{N} en(b_l))$. This way, the embeddings of paired nucleus and Golgi are pulled towards their mean.

The push loss (Eq. (4)) promotes distinct embeddings for different nucleus–Golgi pairs. Specifically, the loss penalizes embeddings from different pairs with a distance between them smaller than Δ:

$$L_{push} = \frac{1}{L(L-1)}\sum_{k=1}^{L}\sum_{\substack{j=1\\ j\neq k}}^{L} max(0, \Delta - \left|e_k - e_j\right|). \qquad (4)$$

The implementation of 3DCellPol is available at https://github.com/HemaxiN/3DCellPol.

Training For 3DCellPol training, we extracted patches of size X × Y × Z × 2 from the 3D images of the real nucleus–Golgi dataset (Fig. 6(1)). The ground-truth centroid heatmaps (Fig. 6(3),(4)) were automatically created based on the manually annotated nucleus–Golgi vectors (Fig. 6(2)). These heatmaps consist of a 3D Gaussian kernel centered at each ground-truth centroid. In [38], the radius of the Gaussian kernel was defined based on the typical sizes of the objects of interest. Following that idea, here the radius of the Gaussian kernel is defined as $\frac{1}{2}\frac{MajorAxis+MinorAxis}{2}$, where $MajorAxis$ and $MinorAxis$ are defined in Table 1. Thus, nuclei Gaussian kernels are bigger than Golgi Gaussian kernels (Fig. 6(3),(4)). In our preliminary experiments, we also investigated other sizes for the Gaussian kernels, such as $\frac{MajorAxis}{2}$ and $\frac{MinorAxis}{2}$, but these sizes produced poorer results. We selected the element-wise maximum for voxels with overlapping Gaussian kernels.

Testing Overlapped patches of size X × Y × Z × 2 were extracted from each image (Fig. 7(1),(2)). These were fed into the trained 3DCellPol model, which outputs the centroid heatmaps and embedding maps of nuclei and Golgi for each patch.

Firstly, the centroid positions and probabilities were extracted from the centroid heatmaps by identifying local maximizers and maxima (Fig. 7(3)). We only considered local maxima with a value higher than a threshold τ and at least a minimum distance (d_{min}) between their positions. The τ and d_{min} values were defined, for each class (nucleus or Golgi), using the validation set.

Afterward, the values of the predicted embeddings at the previously computed local maximizers were extracted from the predicted embedding maps (Fig. 7(4)). Finally, each nucleus was paired with the Golgi presenting the closest embedding and ensuring a one-to-one correspondence between them. Association between a nucleus and a Golgi with an Euclidean distance between their centroids greater than d_{max} and L1-distance between their embeddings greater than Δ (Eq. (4)) was not allowed.

After the nucleus–Golgi assignment based on their embeddings, we computed a probability for each predicted vector. This value was calculated as the mean of the probabilities associated with the nucleus centroid and the Golgi centroid involved in the assignment (Fig. 7(5)). Finally, for overlapping patches, if there are vectors with similar IPs and/or EPs we keep only the vector with the highest probability and suppress all the others (Fig. 7(6),(7)). Specifically, this suppression is applied to vectors with IPs with a distance smaller than nuclei d_{min}, and EPs with a distance smaller than Golgi d_{min}.

4.2.2. Generation of synthetic images and nucleus-Golgi vectors

We generated synthetic images using a GAN to overcome the limitation of having a small amount of data for 3DCellPol training. We used a Vox2Vox model, an extension to 3D of the Pix2Pix model proposed for image-to-image translation [41]. The Vox2Vox model was derived from the Pix2Pix model by replacing the 2D layers (convolutional and transposed convolutional) with 3D layers. In our previous work [46], we showed that the Vox2Vox model can generate good quality synthetic

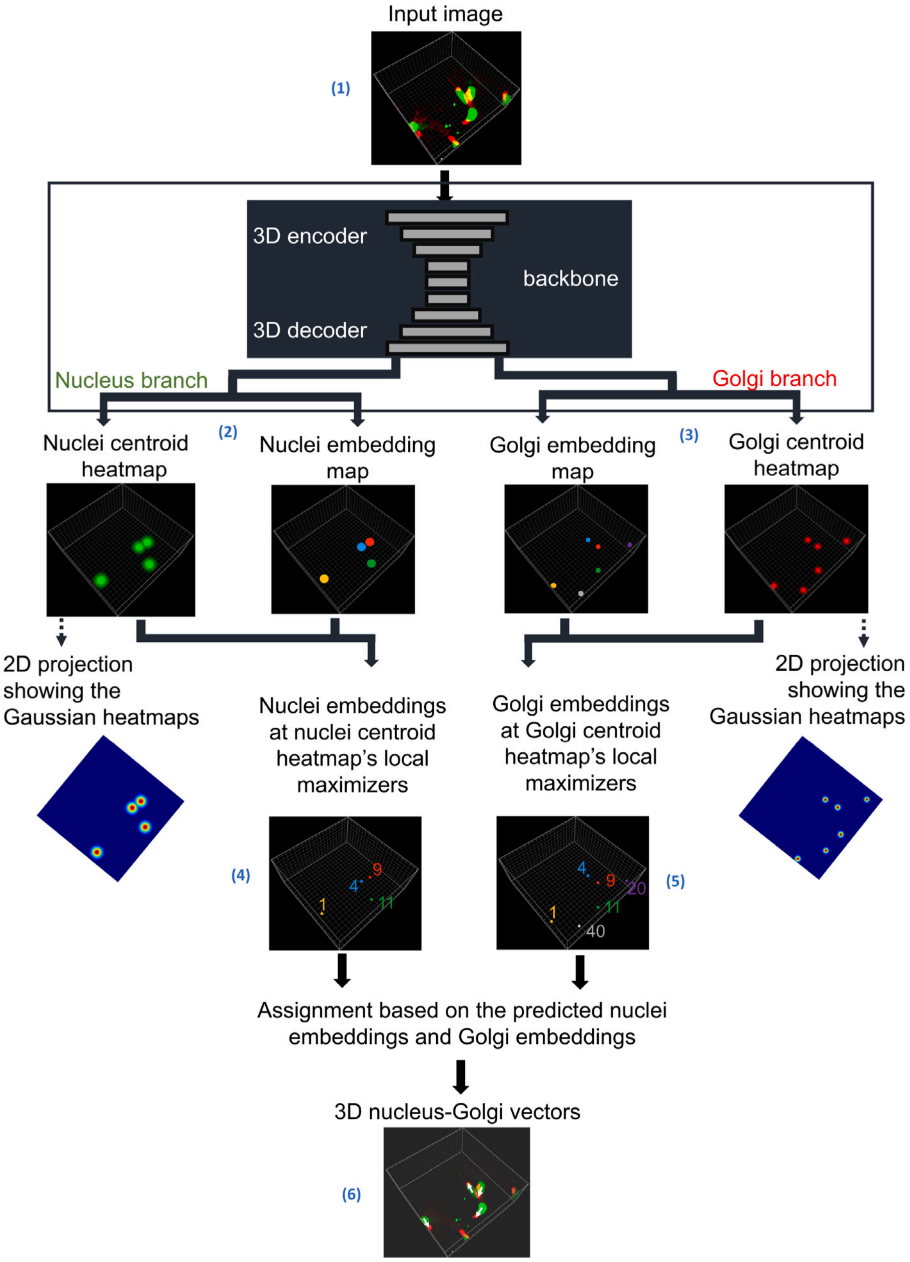

Fig. 5. Overview of the proposed 3DCellPol approach. (1) Example of a 2-channel image subvolume containing nuclei and Golgi; (2)/(3) nucleus/Golgi branch to predict nuclei/Golgi centroid heatmaps and nuclei/Golgi embeddings; (4)/(5) nuclei/Golgi embedding values at the nuclei/Golgi centroid heatmaps local maximizers; (6) predicted polarity vectors.

images. In this work, we use it to translate organelle segmentation masks into images. However, since manual annotation of ground-truth volumes is difficult and time-consuming, we generated the masks using a semi-automated method.

Training To train the Vox2Vox model for synthetic image generation, subvolumes of size X × Y × Z × 2 were extracted from images of the real nucleus–Golgi dataset, and also from segmentation masks. To create the segmentation masks, we extracted image subvolumes from the green channel (16 × 16 × 9) μm around each ground-truth nucleus centroid and from the red channel (8 × 8 × 5) μm around each ground-truth Golgi centroid. The dimensions of the subvolumes were selected based on the typical nuclei and Golgi sizes. Then, Otsu's thresholding [47] method was applied to these subvolumes to obtain the segmentation masks.

Generation of New Synthetic Images After obtaining the trained model, we used the generator to create new synthetic images. The entire process is shown in Fig. 8, and consists of four main steps:

- 1: A dictionary with the organelle segmentation masks was created (Fig. 8(1)). The masks were obtained by applying Otsu's thresholding method [47] locally around each ground-truth centroid.
- 2: Organelle centroids were placed in a subvolume of size X × Y × Z × 2 (Fig. 8(2)). Centroids were generated randomly, with a maximum distance between a nucleus and a Golgi centroid of 14 μm. Additionally, we avoided overlap between centroids, ensuring an Euclidean distance of at least 6 μm between two centroids. These values were selected based on the typical distributions of nuclei and Golgi centroids in the real microscopy images.
- 3: Individual organelle masks were randomly selected from the dictionary for each organelle centroid. Additionally, to add variability to our synthetic dataset, we randomly applied *z*-axis

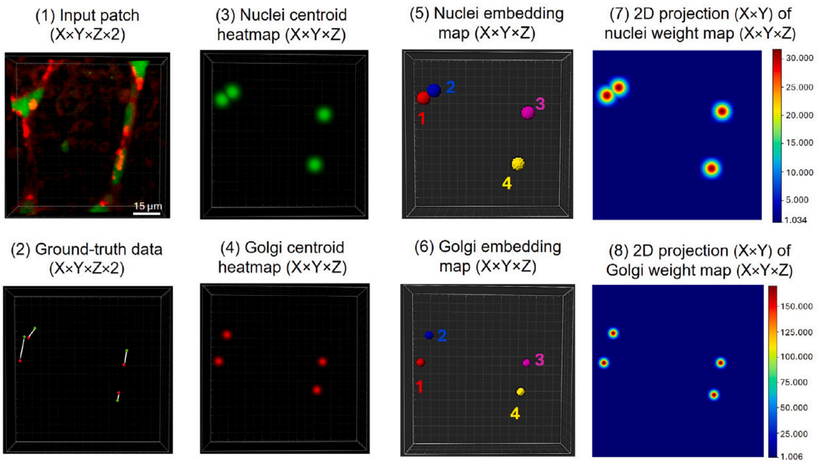

Fig. 6. 3DCellPol's training data. (1) Example of a training sample, the input patch contains red and green channels representing the Golgi and nuclei, respectively. (2) Corresponding hand annotated ground-truth paired nucleus–Golgi centroids represented in a subvolume. The red and green dots indicate the annotated nuclei and Golgi centroids, respectively, while each white line represents a nucleus–Golgi pair. (3),(4) The manually annotated pairs of nucleus–Golgi centroids are converted into centroid ground-truth heatmaps by placing a 3D Gaussian around each centroid. (5),(6) The embedding maps show the neighborhood around each ground-truth keypoint. These maps are used to compute the pull loss (Eq. (3)). The neighborhood around a nucleus and a Golgi from the same cell are represented with the same color. (7),(8) Examples of 2D projections of nuclei and Golgi 3D weights maps. The 3D maps are used to compute the $L_{heatmap}$ loss (Eq. (2)).

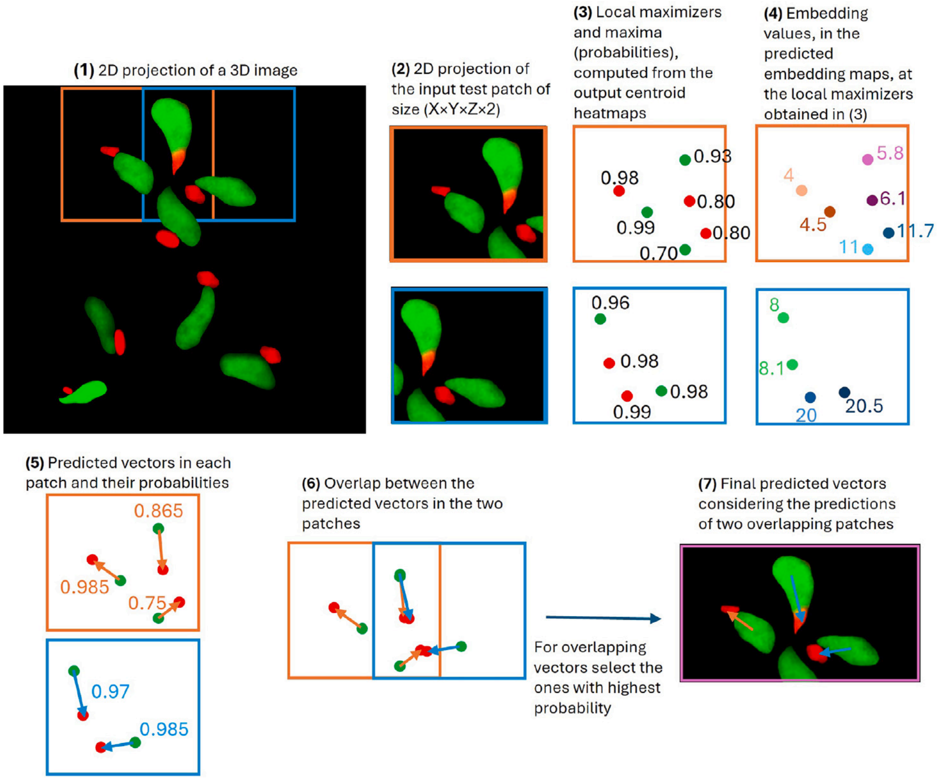

Fig. 7. Schematic representation of 3DCellPol's testing stage. (1) 2D projection of a 3D image containing nuclei and Golgi; (2) two test patches extracted from the image shown in (1) which are fed to 3DCellPol; (3) local maximizers computed from the centroid heatmaps of nuclei and Golgi, and their values (representing probabilities), these local maximizers are the predicted centroids; (4) values of the predicted embeddings at the local maximizers shown in (3); (5) vectors computed for each patch according to the predicted centroids and embeddings, and their probabilities computed as the mean of the connected nucleus and Golgi centroids' probabilities; (6) illustration of the overlap between the predictions of the two patches; (7) the final predicted vectors are obtained by keeping only the vector with the highest probability for vectors that have similar IPs and/or EPs. Note that this example was created particularly to demonstrate the testing procedure, and it is important to note that the scales of nuclei, Golgi and orange and blue patches illustrated in this figure are different from those seen in the real dataset. 2D projections are shown to simplify the explanations, but all analyses were performed in 3D.

aligned rotations, horizontal flips and/or vertical flips to each mask extracted from the dictionary. Finally, the mask was placed at the centroid's position (Fig. 8(3)).

- 4: The segmentation mask was fed to the generator of the Vox2Vox model (Fig. 8(4)) to create the corresponding synthetic image (Fig. 8(5)).

The synthetic images contain organelles in different configurations, which adds more variability to the dataset. Moreover, the synthetic images are created from paired organelle centroids, ensuring that each synthetic image has corresponding ground-truth vectors for training the 3DCellPol model.

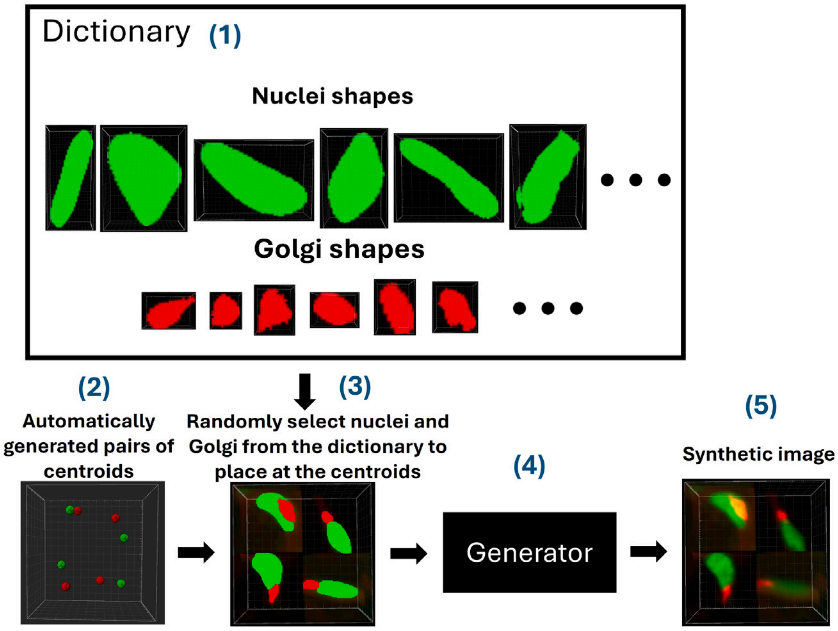

Fig. 8. Synthetic image generation. Schematic representation of the generation of synthetic images using a GAN: (1) creation of a dictionary with individual nuclei and Golgi masks; (2) generation of nuclei/Golgi centroids in a volume (X × Y × Z × 2); (3) random selection of nuclei/Golgi masks which are rotated and/or flipped and placed at the positions of the nuclei/Golgi centroids to obtain a segmentation mask; (4) trained generator of the Vox2Vox GAN model; (5) synthetic microscopy image obtained by feeding the segmentation mask created in (3) to the trained generator (4). This is only an illustration to show the process of synthetic image generation. The real dimensions of these inputs and outputs differ from those shown in the Figure.

Our implementation of the Vox2Vox model to generate synthetic 3D microscopy images is publicly available at https://github.com/HemaxiN/3DVox2Vox.

4.3. Comparison methods

4.3.1. Comparison methods for nucleus-Golgi polarity

We compared the performance of 3DCellPol with multi-stage approaches. For the segmentation step of these methods, we considered the traditional Otsu thresholding method used in [23], as well as state-of-the-art deep learning approaches: 3D U-Net [24] and StarDist-3D [27].

We also considered an approach that only predicts nuclei and Golgi centroid heatmaps (3DCtrDet). It has the same architecture as the 3DCellPol model but does not have the embedding outputs. The centroids are computed by extracting the local maximizers from the predicted heatmaps and pairing is performed based on the distances between predicted centroids, since no embeddings are available.

In [23], the assignment is based on a cost matrix that depends on the Euclidean distances between the centroids of nuclei and Golgi. Similarly, in this work, for each of the three segmentation approaches (Otsu's Thresholding [47], 3D U-Net [24] and StarDist-3D [27]) and for the centroids detection method (3DCtrDet), we compute the cost matrix proposed in [23]. For each test image, a cost matrix (CM) with X rows and Y columns is built, where X and Y denote the number of segmented/detected nuclei and Golgi, respectively. The entry $CM(n,g)$ represents the following quantity computed based on the nth nucleus centroid and gth Golgi centroid:

$$CM(n,g) = \left\| \hat{NC}_n - \hat{GC}_g \right\|_2 \tag{5}$$

where $\|\cdot\|_2$ denotes the L2-norm, $\hat{NC}_n$ and $\hat{GC}_g$ denote the centroids of the nth predicted nucleus and gth predicted Golgi, respectively.

To find the matches between nuclei and Golgi with a minimum total cost, the Hungarian method [48] is applied to CM. Thereafter, a threshold d_{max} is applied to the nucleus–Golgi Euclidean distance to remove incorrect assignments, as done in [23].

On the one hand, for 3D U-Net, the watershed method [49] was applied to its output masks to separate touching objects before computing the quantities shown in Eq. (5). On the other hand, StarDist-3D outputs the instance segmentation masks for nuclei and Golgi which are directly used to extract the nuclei and Golgi centroids for CM computation (Eq. (5)). Moreover, 3DCtrDet outputs the centroid heatmaps for nuclei and Golgi. Local maxima were extracted considering a threshold and the distance between their positions, these values were selected based on the validation set. The detected nuclei and Golgi centroids were used to compute the quantities presented in Eq. (5).

Finally, we also compare the performance of our proposed approach with 3D U-Net HS [20]. However, herein we applied watershed to the joint channel of nuclei and Golgi to split touching pairs, a post-processing step that was not considered in our previous work.

4.3.2. Comparison methods for cytoplasm-nucleus polarity

For the cytoplasm-nucleus dataset [21], we implemented a 2D adaptation of 3DCellPol, which we denote as 2DCellPol. We compared its performance with two multi-stage methods: 2DCtrDet and StarDist-2D, which are the 2D versions of 3DCtrDet and StarDist-3D, respectively. The cytoplasm-nucleus vectors for these multi-stage approaches were computed as follows:

- (1) Prediction of cytoplasm and nuclei centroids
  - StarDist-2D: The trained model was used to infer the instance segmentation masks of cytoplasm and nuclei in the test images. The predicted cytoplasm and nuclei centroids were computed from the segmentation masks.
  - 2DCtrDet: The trained model was used to predict the centroid heatmaps of cytoplasm and nuclei, the centroids of these structures were computed by extracting the local maximizers from the predicted centroid heatmaps.
- (2) A cost matrix with the Euclidean distances between the cytoplasm and nucleus centroids was built, and the Hungarian algorithm was applied to pair cytoplasm and nuclei. Finally, a maximum threshold was applied to the Euclidean distance between the cytoplasm and nuclei centroids.

Table 3
Training and test images, and number of training and validation patches for each fold to perform nested three-fold cross-validation. The training patches were further split into training/validation with approximately 90%/10% ratio. There is no overlap between training and validation patches.

	Fold I	Fold II	Fold III
Training images	A,B	A,C	B,C
# Training patches	179	170	127
# Validation patches	18	17	12
Test image	C	B	A

4.4. Evaluation methodology

4.4.1. Evaluation of nucleus–Golgi polarity

Nested Three-Fold Cross-Validation For evaluation, we performed a nested leave-one-image-out cross-validation, with 3-folds because we have 3 images in the real nucleus–Golgi dataset (A, B, C). Patches of size $128 \times 128 \times 64 \times 2$ were extracted from each image resulting in 197, 187 and 139 training patches for folds I, II and III. Each fold includes patches from different image pairs, that is, folds I, II and III contain patches from images (A+B), (A+C) and (B+C), respectively, and the corresponding held-out test images are C, B and A (Table 3). This cross-validation strategy avoids the bias of testing a model on patches obtained from the same images as the patches used to train the model.

For each approach, except the thresholding method [23], we trained three models and evaluated the performance of each model on the corresponding held-out test set. Briefly, hyperparameter optimization was performed for each fold, and the best parameters were selected based on the model's performance on the validation set. Then, for each fold, the model with the best validation performance was evaluated on the corresponding held-out test set. However, with this division of the dataset into three folds we obtain some folds with more training patches than others (Table 3).

To compensate for the differences in the number of training patches in these three folds, we applied different online data augmentation probabilities to each training fold. We applied z-axis aligned rotations in the range $(0, 360°)$ with steps of size 90°, horizontal flips, vertical flips and/or image brightness and contrast variations, each with predefined probabilities for each fold. That is, for each fold i ($i \in \{I, II, III\}$) the probability was defined as: $\frac{\#\ train\ patches\ \in fold\ I}{\#\ train\ patches\ \in\ fold\ i} \times 0.5$. In this way, for the fold with the highest number of patches (fold I), we considered a data augmentation probability of 0.5. Moreover, we added more variability to folds with fewer training patches (folds II and III) since the probability of data augmentation is inversely proportional to the number of training patches in these two folds.

Furthermore, for each fold, we trained a Vox2Vox model for synthetic image generation, and evaluated each model's performance on its fold's held-out test set. We then used the trained models to create the synthetic nucleus–Golgi dataset by generating 197, 187 and 139 new synthetic patches of size $128 \times 128 \times 64 \times 2$, for training folds I, II and III, respectively, as shown in Fig. 8.

Afterward, we compared three 3DCtrDet models and three 3DCellPol models considering different training sets and strategies: real nucleus–Golgi images, synthetic nucleus–Golgi images, and a combination of real and synthetic nucleus–Golgi images (Table 4). Specifically, 3DCtrDet Real and 3DCellPol Real were trained using only the real nucleus–Golgi dataset. 3DCtrDet Synthetic and 3DCellPol Synthetic were trained from scratch, using only the synthetic nucleus–Golgi dataset. 3DCtrDet Synthetic+Real and 3DCellPol Synthetic+Real were subject to fine-tuning using the real nucleus–Golgi dataset, considering 3DCtrDet Synthetic and 3DCellPol Synthetic as the starting point, respectively. The synthetic images were used only for the training step. The performance of all models was evaluated on the real held-out test images.

Evaluation Metrics for nucleus–Golgi Vectors To evaluate the performance of the nucleus–Golgi vectors prediction methods, it is first necessary to find the match between each ground-truth vector and each predicted vector. We adopted a methodology similar to the matching strategy commonly used in object detection [50,51] and instance segmentation tasks [27].

The evaluation strategy consists of computing metrics at different thresholds. The threshold is applied to the metric defined as the matching criterion. For instance, the matching criterion for bounding boxes is the intersection over union (IoU) between ground-truth and predicted bounding boxes [50], and for centroid detection, it is the Euclidean distance between ground-truth and predicted centroids [51].

We propose a matching criterion considering the distances between the IPs and EPs of the predicted and ground-truth vectors. This is because a predicted vector should be considered a true positive (TP) only if its IP and EP are close to those of a ground-truth vector. Therefore, we build a $G \times P$ pairing matrix (PM), where G denotes the number of ground-truth vectors, and P is the number of predicted vectors. The entry $PM\,(t, p)$ contains the sum of the Euclidean distances between the IPs and EPs of the tth ground-truth vector and pth predicted vector:

$$\left\| IP_t - \hat{IP}_p \right\|_2 + \left\| EP_t - \hat{EP}_p \right\|_2 \tag{6}$$

where IP_t and EP_t are the initial and endpoints of the tth ground-truth vector, $\hat{IP}_p$ and $\hat{EP}_p$ are the initial and endpoints of the pth predicted vector. Thereafter, the Hungarian method [48] is applied to PM to find matches between the predicted and ground-truth vectors.

After matching, to identify the TP vectors, we compute the distances between the IPs and EPs of each ground-truth and its matched predicted vector. That is, we count a predicted vector as TP if:

$$\left\| IP - \hat{IP} \right\|_2 \leq \tau_n \ and \ \left\| EP - \hat{EP} \right\|_2 \leq \tau_g \tag{7}$$

where τ_n and τ_g are thresholds (Sup. Fig. 2). That is, if the distance between the IPs and the distance between the EPs is below τ_n and τ_g, respectively, then the predicted vector is counted as TP.

More specifically, we computed the number of TP at different values of the thresholds τ_n and τ_g. We considered two thresholds (τ_n and τ_g) because the IPs and EPs correspond to nuclei and Golgi centroids, respectively, and these organelles have different sizes (Fig. 3 and Table 1). To select the thresholds (Table 5), we computed the major and minor axes of nuclei and Golgi using their ground-truth segmentation masks (Table 1).

In Table 5, the major axis fraction (MAF) varies between 0.75 and 0.10 with steps of size 0.05, and denotes the fraction of the mean major axis of nuclei (16 μm) and Golgi (8 μm) used to compute nuclei and Golgi thresholds (τ_n and τ_g). Thus, at each τ_L: $\tau_n = MAF_{at\ \tau_L} \times 16$ μm and $\tau_g = MAF_{at\ \tau_L} \times 8$ μm. As observed in Table 5, increasing threshold levels τ_L, from 0 to 13, correspond to more restrictive thresholds. Thus, it is expected that the number of TP decreases with increasing τ_L.

For each τ_L, we also computed the number of false positives (FP), which counts the predicted vectors that are not matched with any of the ground-truth vectors, and the number of false negatives (FN), which are the ground-truth vectors that are not matched with any of the predicted vectors.

Additionally, based on the TP, FN and FP we computed, at each τ_L, the true positive rate (TPR) and false discovery rate (FDR) as shown in Eqs. (8) and (9), respectively.

$$TPR = \frac{TP}{TP + FN} \tag{8}$$

$$FDR = \frac{FP}{FP + TP} \tag{9}$$

Moreover, to analyze the quality of the predicted vectors, we also computed, for the TP obtained at different τ_L, the following metrics:

Table 4
Number of training patches for each fold to perform nested three-fold cross-validation considering different training sets of real and/or synthetic images, and distinct training strategies.

Model	Type of data and model training	Fold I	Fold II	Fold III
3DCellPol Real 3DCtrDet Real	Real (from scratch)	179	170	127
3DCellPol Synthetic 3DCtrDet Synthetic	Synthetic (from scratch)	179	170	127
3DCellPol Synthetic+Real 3DCtrDet Synthetic+Real	Synthetic (from scratch) + Real (fine-tuning)	179+179	170+170	127+127

Table 5
Thresholds considered for the proposed matching criterion to count the number of TP, FP and FN for 3D nucleus–Golgi vector prediction models; τ_n and τ_g denote the thresholds applied to the Euclidean distance between the nuclei and Golgi centroids, respectively, of matched predicted and ground-truth vectors. $\tau_{n(at\ \tau_L)}$ and $\tau_{g\,(at\ \tau_L)}$ are calculated as $MAF_{at\ \tau_L} \times 16$ μm and $MAF_{at\ \tau_L} \times 8$ μm.

Threshold level (τ_L)	0	1	2	3	4	5	6	7	8	9	10	11	12	13
$MAF_{at\ \tau_L}$	0.75	0.70	0.65	0.60	0.55	0.50	0.45	0.40	0.35	0.30	0.25	0.20	0.15	0.10
$\tau_{n(at\ \tau_L)}$ (μm)	12.0	11.2	10.4	9.6	8.8	8.0	7.2	6.4	5.6	4.8	4.0	3.2	2.4	1.6
$\tau_{g\,(at\ \tau_L)}$ (μm)	6.0	5.6	5.2	4.8	4.4	4.0	3.6	3.2	2.8	2.4	2.0	1.6	1.2	0.8

- Cosine Similarity: cosine of the angle between predicted and ground-truth vectors
  $$CS = \frac{V \cdot \hat{V}}{\|V\|_2 \|\hat{V}\|_2} \tag{10}$$
  where V and $\hat{V}$ denote the ground-truth and predicted vectors, respectively, and $\cdot$ denotes the inner product;
- Norm of the Error Vector: L2-norm of the difference between predicted and ground-truth vectors
  $$Error = \left\|V - \hat{V}\right\|_2 \tag{11}$$
- Nucleus Centroid Distance: distance between predicted and ground-truth nuclei centroids
  $$ND = \left\|NC - \hat{NC}\right\|_2 \tag{12}$$
  where NC and $\hat{NC}$ denote the ground-truth and predicted nuclei centroids, respectively;
- Golgi Centroid Distance: distance between predicted and ground-truth Golgi centroids
  $$GD = \left\|GC - \hat{GC}\right\|_2 \tag{13}$$
  where GC and $\hat{GC}$ denote the ground-truth and predicted Golgi centroids, respectively.

Furthermore, we excluded nucleus–Golgi pairs located at image borders, since the information they provide is incomplete. Therefore, we considered 412 nucleus–Golgi pairs for evaluation.

In this work, we want to detect the nucleus–Golgi vectors and it is important to develop a model with a high TPR while achieving a small FDR. That is, we want to detect as many nucleus–Golgi vectors as possible and minimize the false detection of vectors. Our final goal is to compute cell polarity, and incorrect organelles assignments (FP) may lead to erroneous conclusions regarding cell polarization and migration associated to different biological processes. Furthermore, measuring the quality of the predicted vectors is also important. The cosine similarity measures if the ground-truth and predicted vector have similar directions. It is 1 when both vectors are perfectly aligned, so our goal is to obtain values closer to 1 to ensure that the predicted vectors conform with the real vectors in terms of direction. Since the cosine similarity does not depend on the vector lengths, we also compute the L2-norm of the difference between the matched vectors. This metric, which we denote by Error, depends both on the size of the vectors and on their orientation, and our goal is to obtain small values. Moreover, we also computed the distances between the ground-truth and predicted centroids of nuclei and Golgi. Again, small values (closer to zero) are desirable to ensure that the model makes accurate predictions of the centroids of nuclei and Golgi.

Evaluation Metric for Synthetic nucleus–Golgi Images To evaluate the performance of the synthetic image generation approach we considered the structural similarity index measure (SSIM) measure [52], which is computed between a real image (r) and the corresponding synthetic image (s):

$$SSIM(r, s) = \frac{(2\mu_r\mu_s + C_1) + (2\sigma_{rs} + C_2)}{(\mu_r^2 + \mu_s^2 + C_1) + (\sigma_r^2 + \sigma_s^2 + C_2)} \tag{14}$$

where μ_r and μ_s denote the mean voxel intensity of r and s, respectively; σ_r^2 and σ_s^2 correspond to the variance of r and variance of s, respectively; σ_{rs} is the covariance of r and s; and, $C_1 = (K_1 L)^2$ and $C_2 = (K_2 L)^2$. We use the values $L = 255$, $K_1 = 0.01$ and $K_2 = 0.03$, as suggested in [52].

4.4.2. Cytoplasm-nucleus polarity

Train/Validation/Test Split Each model was trained with 400 images and the performance was evaluated on 100 held-out test images, according to the split specified in [21].

Evaluation Metrics for Cytoplasm-Nucleus Vectors The strategy used to evaluate the performance of cytoplasm-nucleus vector prediction models is similar to the one presented for nucleus–Golgi vectors. However, the thresholds at which the metrics are computed are different. Since we have no information about the images' physical dimensions, we defined the thresholds in pixels. First, based on the mean sizes of the major and minor axes of cytoplasm and nuclei (Table 2), we computed the evaluation thresholds as indicated in Table 6. At these thresholds we computed the metrics TPR (Eq. (8)), FDR (Eq. (9)), CS (Eq. (10)), $Error$ (Eq. (11)), ND (Eq. (12)), and Cytoplasm Centroid Distance:

$$CytoD = \left\|CytoC - \hat{CytoC}\right\|_2 \tag{15}$$

where $CytoC$ and $\hat{CytoC}$ denote the ground truth and predicted cytoplasm centroids, respectively.

4.5. Experiments

4.5.1. Nucleus-Golgi polarity

Implementation Details for 3DCellPol For training 3DCellPol on the real nucleus–Golgi dataset, padding was performed before patch extraction to ensure that the image sizes were multiples of the patch sizes. The images were padded with small random values in the interval $[0, 10]$, which represent the typical values of the background voxels in these images. Then, patches of size $128 \times 128 \times 64 \times 2$ were obtained

Table 6
Thresholds considered for the proposed matching criterion to count the number of TP, FP and FN for 2D cytoplasm-nucleus vector prediction models; τ_c and τ'_n denote the thresholds applied to the Euclidean distance between the cytoplasm and nuclei centroids, respectively, of matched predicted and ground-truth vectors. $\tau_{c(at\ \tau_L)}$ and $\tau'_{n(at\ \tau_L)}$ are calculated as $MAF_{at\ \tau_L} \times 96$ pixels and $MAF_{at\ \tau_L} \times 17$ pixels.

Threshold Level (τ_L)	0	1	2	3	4	5	6	7	8	9	10	11	12	13
$MAF_{at\ \tau_L}$	0.75	0.70	0.65	0.60	0.55	0.50	0.45	0.40	0.35	0.30	0.25	0.20	0.15	0.10
$\tau_{c(at\ \tau_L)}$ (pixels)	72.0	67.2	62.4	57.6	52.8	48.0	43.2	38.4	33.6	28.8	24.0	19.2	14.4	9.6
$\tau'_{n(at\ \tau_L)}$ (pixels)	12.75	11.9	11.05	10.2	9.35	8.5	7.65	6.8	5.95	5.1	4.25	3.4	2.55	1.7

from the images, and were normalized to present intensity values ranging between 0 and 1. Each model of the nested three-fold cross-validation was trained from scratch for 400 epochs using the ADAM optimizer [53]. Details about 3DCellPol's hyperparameters are shown in Sup. Fig. 3 A. We monitored both training and validation losses during the training process. After approximately 350 epochs both losses stabilized, thus we chose the final model to evaluate its performance on the test set. Examples of 3DCellPol's training and validation curves for fold I are shown in Sup. Fig. 3B.

The foreground and background weights, w_f and w_b, used in Eq. (2), were computed separately for nuclei and Golgi centroid heatmaps. The weights w_{f_N} (w_{f_G}) were set to be inversely proportional to the number of voxels representing nuclei (Golgi) voxels in the heatmaps, while the weights w_{b_N} (w_{b_G}) were chosen such that they are inversely proportional to the number of background voxels in the heatmaps. Additionally, following Harley et al. [39] we set Δ, defined in Eq. (4), to 2.

During testing, the same padding strategy was applied to the test images. Then, patches of size 128 × 128 × 64 × 2 and an overlap of 32 voxels along the x and y directions were extracted from each image. The threshold d_{max} was set to 14 μm, which is approximately the maximum distance between the ground-truth centroids of paired nucleus and Golgi.

Implementation Details for Vox2Vox We trained a Vox2Vox model using the real nucleus–Golgi dataset for each training fold (I, II and III). Each model was trained for 200 epochs, with a learning rate of 2e−4 and batch size of 2 using the Adam optimizer [53].

Implementation Details for the Comparison Methods

- **3D U-Net:** We used a publicly available implementation of 3D U-Net: https://github.com/ellisdg/3DUnetCNN. It was trained for 200 epochs with a learning rate of 1e−4 and using the Adam optimizer [53].
- **StarDist-3D:** We used a publicly available implementation to train StarDist-3D: https://github.com/stardist/stardist. It was trained for 300 epochs, with 320 rays, and the other parameters set as described in [27].
- **3D U-Net HS:** We trained the 3D U-Net HS model as described in our previous work [20].
- **3DCtrDet:** This approach was trained for 400 epochs, using the Adam optimizer [53], with a learning rate of 1e−3.

For all approaches the d_{max} threshold was set to 14 μm.

4.5.2. *Cytoplasm-nucleus polarity*

Implementation Details for 2DCellPol 2DCellPol was trained from scratch for 400 epochs with batch size of 16 and learning rate of 1e−3 using the Adam optimizer [53]. The weights w_f and w_b (Eq. (2)) were computed based on the cytoplasm and nuclei heatmaps using the strategy explained above for 3DCellPol.

Implementation Details for the Comparison Methods The models were trained from scratch for 400 epochs with the Adam optimizer [53], a batch size of 16, and using a learning rate of 1e−3 and 1e−2, for StarDist-2D and 2DCtrDet, respectively.

For all approaches, the maximum allowed distance between the centroids of paired cytoplasm and nucleus is 40 pixels.

4.5.3. *Computational environment*

All experiments were carried out on a computer with the following specifications: 32 GB RAM, NVIDIA GPU GeForce 1080 Ti and Python 3.5.2. The implementation of 3DCellPol is based on Keras [54] with Tensorflow backend [55]. The 3D images were visualized on ImarisViewer.

5. Results

In this section, we apply our method to predict polarity vectors from 3D and 2D images, analyze their performance, and compare it with the vector prediction approaches based on Otsu, 3D U-Net, StarDist-3D, 3D U-Net HS, and 3DCtrDet, which were described in the previous section. For the 3D task, we also generate synthetic data using the methodology described in Section 4.2.2, and evaluate the impact of adding these data for training.

5.1. *3DCellPol for nucleus-Golgi polarity vector prediction*

First, we analyzed 3DCellPol's qualitative results for nucleus–Golgi vector prediction using the real nucleus–Golgi dataset. Training 3DCellPol took approximately 22 h, 21 h and 15.5 h for folds I, II and III, respectively. Moreover, the test time is approximately 3 min per image of size $768 \times 768 \times 64 \times 2$. Fig. 9 illustrates some 2D projections of 3D microscopy test image patches, with the ground-truth vectors and predicted vectors represented in white. It also shows 2D projections of the predicted nuclei and Golgi centroid heatmaps, the detected local maximizers (black dots) and their embeddings. Some prediction errors are marked with colored asterisks, as described at the bottom of the Figure.

These qualitative results illustrate that 3DCellPol detects the majority of the nucleus–Golgi vectors even for touching pairs (Fig. 9(2),(3),(4)), and most of the predicted embeddings allow for a straightforward nucleus–Golgi pairing. An exception occurs in the patch shown in Fig. 9(5), there are three correctly identified organelles with similar embeddings, two Golgi with embeddings 5.64 and 6.13, and a nucleus with embedding 5.79, making it difficult to identify the real pair, consequently leading to incorrect nucleus–Golgi pairing. Another exception is illustrated in Fig. 9(4), where similar embeddings are predicted for two touching nuclei leading to incorrect pairing of two nuclei and two Golgi. Furthermore, the vectors that are not correctly detected are mostly due to difficult configurations of nucleus–Golgi pairs, specifically nuclei that look like a single nucleus (pink asterisks in Fig. 9(1),(2),(3)), due to nuclei and/or Golgi with small intensity (white asterisks in Fig. 9(1),(2)), and also due to over-detection of nuclei (green asterisk in Fig. 9(3)).

5.2. *Comparison of nucleus-Golgi vector prediction methods*

Next, we compared the performance of 3DCellPol with previously proposed methods for nucleus–Golgi pairing. These experiments were performed using only the real nucleus–Golgi dataset. Fig. 10 shows the nucleus–Golgi vector prediction results obtained with Otsu [23], 3D U-Net [24], 3D U-Net HS [20], StarDist-3D [27], 3DCtrDet, and the proposed approach 3DCellPol.

3DCellPol presents the highest true positive rate (TPR) (Fig. 10A) and the smallest false discovery rate (FDR) (Fig. 10B) for all τ_L,

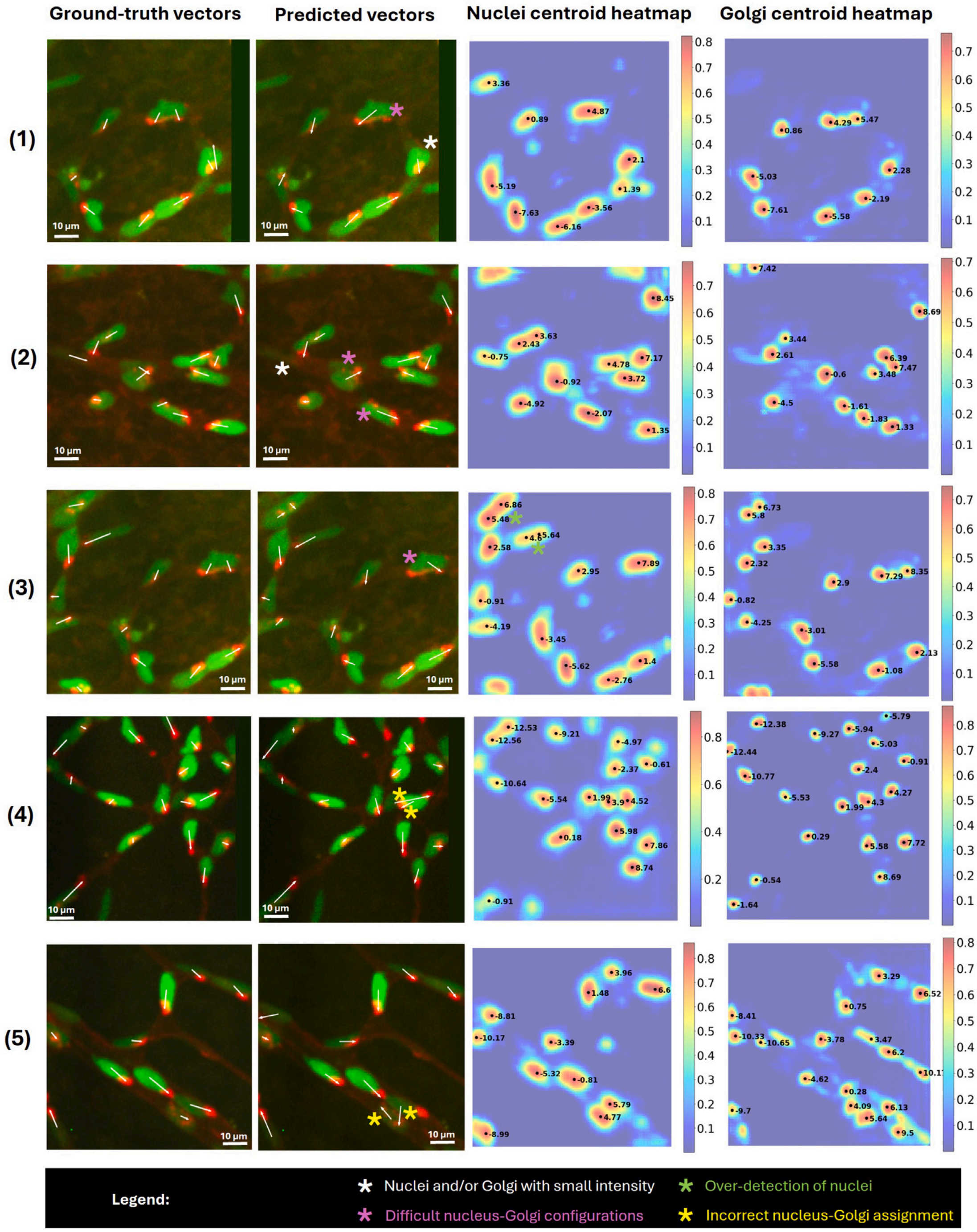

Fig. 9. Performance of 3DCellPol. First and second columns: Examples of ground-truth and predicted vectors, respectively, represented in white on top of 2D projections of input test patches. Third column: 2D projections of the output nuclei centroid heatmaps, and representation of local maximizers (black dots) and nuclei embedding values at local maximizers. Fourth column: 2D projections of the output Golgi centroid heatmaps, and representation of local maximizers (black dots) and Golgi embedding values at local maximizers. Some prediction errors are annotated with colored asterisks as described at the bottom of the Figure.

outperforming all the other methods. The best TPR attained by our method is approximately 71% (at $\tau_L = 0$).

3DCtrDet and StarDist-3D also perform well but their TPR does not reach 61% (Fig. 10A) and their FDR (Fig. 10B) is higher, with 3DCtrDet performing slightly better than StarDist-3D, while 3D U-Net and Otsu's thresholding-based methods perform poorly (Fig. 10A and B).

The cosine similarity (CS) values obtained with 3DCellPol are high, above 0.90 (Fig. 10D). These results indicate that the directions of the ground-truth and 3DCellPol's vectors are very similar, which is essential for estimating EC polarity. Regarding the other vector quality metrics (Fig. 10C, E, F) 3DCellPol presents the smallest Error and ND (nucleus centroid distance) for $\tau_L \leq 8$, and similar values of GD (Golgi centroid distance) compared to StarDist-3D, 3D U-Net HS and 3D U-Net.

For qualitative comparison of the results obtained with the top three performing methods, StarDist-3D, 3DCtrDet and 3DCellPol, we present in Fig. 11 examples of ground-truth vectors (shown in white), and examples of TP, FN and FP vectors (shown in white, pink and yellow, respectively). Furthermore, we annotated the two most common prediction errors, as described at the bottom of Fig. 11.

Overall our qualitative results, in Fig. 11, show that StarDist-3D, 3DCtrDet, and 3DCellPol can detect a large number of nucleus–Golgi pairs even for touching nuclei (Fig. 11(3),(4),(5)), which supports the

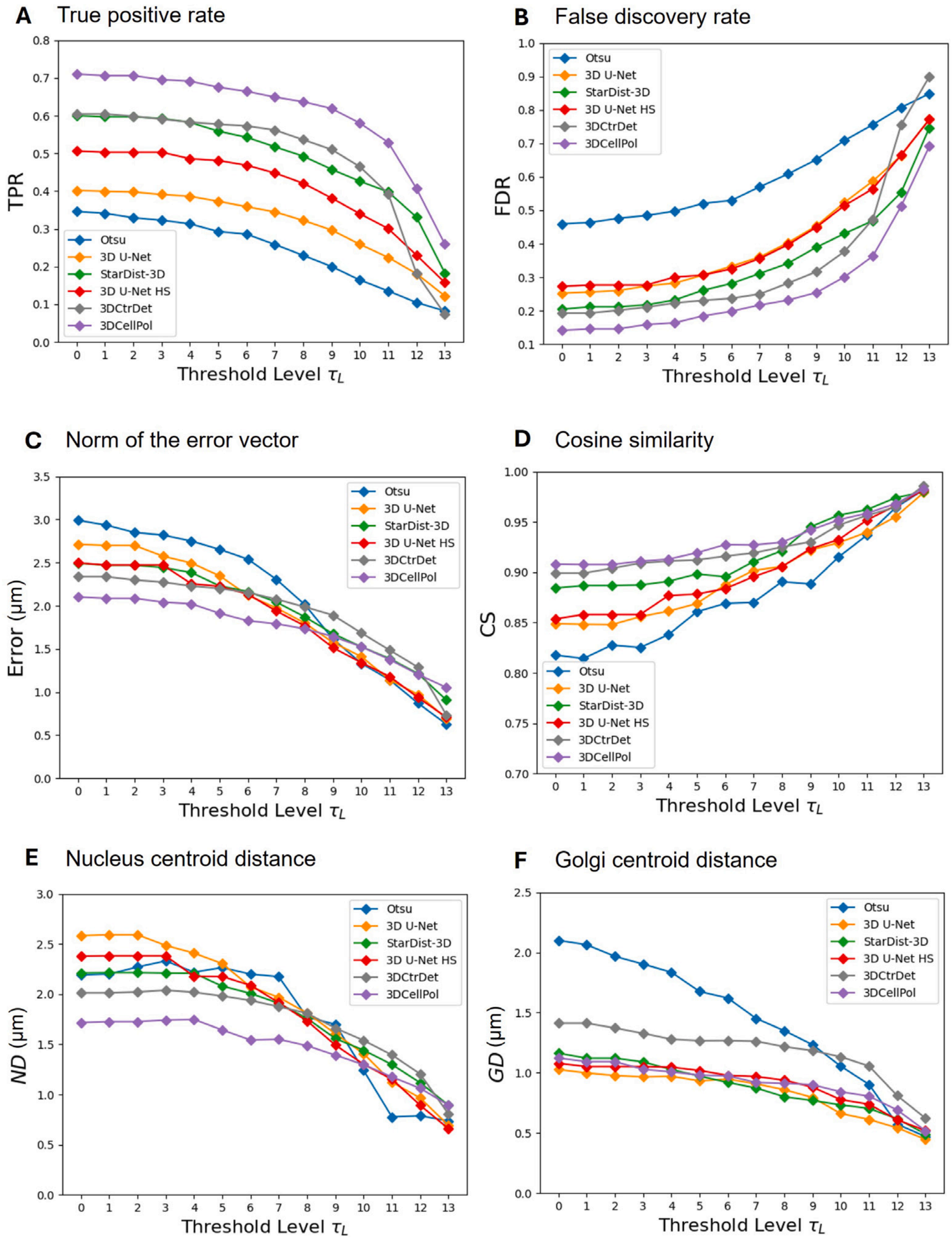

Fig. 10. Quantitative evaluation of the nucleus–Golgi vectors obtained using models trained on the real nucleus–Golgi dataset. The methods compared are: Otsu, 3D U-Net, StarDist-3D, 3D U-Net HS, 3DCtrDet, and 3DCellPol. Performance was evaluated using the following metrics: (A) True Positive Rate (TPR), (B) False Discovery Rate (FDR), (C) Norm of the Error Vector (μm), (D) Cosine Similarity (CS), (E) Nucleus Centroid Distance (ND) (μm), (F) Golgi Centroid Distance (GD) (μm).

quantitative results shown in Fig. 10A. On the one hand, for StarDist-3D and 3DCtrDet, FNs are explained by some nuclei and Golgi that are not segmented/detected (Fig. 11(1),(3)), and other FNs and FPs by incorrect nucleus–Golgi assignment (Fig. 11(1)). On the other hand, all models present several FPs and FNs due to incorrect assignment of nucleus–Golgi pairs in regions of complex configurations of these organelles (Fig. 11(2)) and the presence of noise in the red channel of the microscopy image, which is identified as a Golgi (Fig. 11(4)).

5.3. Synthetic image generation

Next, we trained the Vox2Vox model using the real nucleus–Golgi dataset and then used it to create the synthetic nucleus–Golgi dataset. We evaluated the performance of the Vox2Vox model for synthetic image generation by computing the SSIM [52] between pairs of real images and synthetic images translated from their segmentation masks.

The SSIM values (0.9097 ± 0.686) indicate a high level of agreement between the compared images. This was confirmed by analyzing the qualitative results presented in Fig. 12.

5.4. Impact of training with the real and synthetic nucleus-Golgi datasets

Next, we studied the impact of combining the real and synthetic nucleus–Golgi datasets for 3DCtrDet and 3DCellPol training (Table 4). Examples of synthetic images generated with a Vox2Vox model, using the methodology described in Fig. 8, are presented in Sup. Fig. 4. Although synthetic images are similar to real images (Sup. Fig. 4), they lack some details that are present in the real images, such as

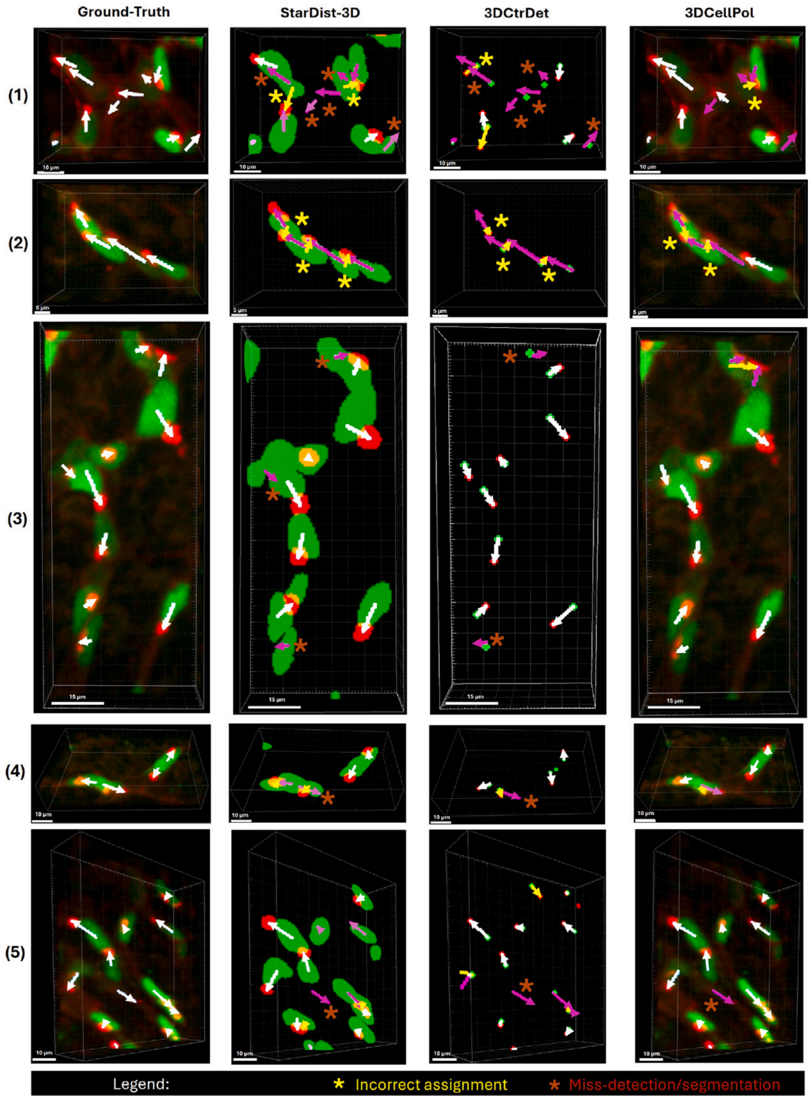

Fig. 11. Qualitative evaluation of the nucleus–Golgi vectors prediction results obtained with StarDist-3D, 3DCtrDet and 3DCellPol. On the left: Examples of fluorescence microscopy image patches (nuclei in green and Golgi in red) and the ground-truth nucleus–Golgi vectors represented in white. Second, third and fourth columns: Corresponding vector prediction evaluation results obtained with StarDist-3D, 3DCtrDet and 3DCellPol, respectively. The white, yellow and pink vectors denote the TP, FP and FN, respectively. These TP, FP and FN correspond to $\tau_L = 5$. The results of StarDist-3D are superimposed on the segmentation masks obtained with the StarDist-3D model. The results of 3DCtrDet are superimposed on top of the detected centroids of nuclei and Golgi, represented in green and red. The results of 3DCellPol are superimposed on the microscopy patch. Two of the most common prediction errors are annotated using different colored asterisks as described at the bottom of the Figure.

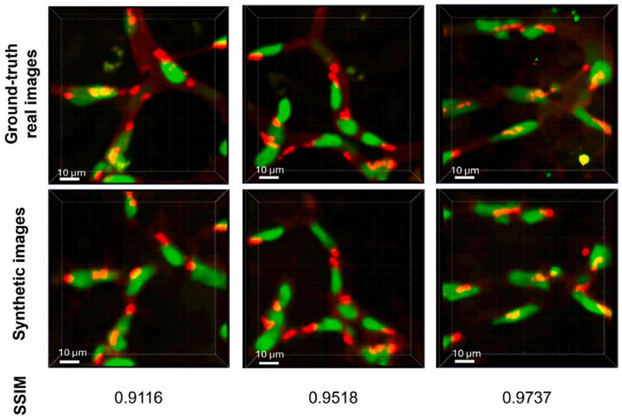

Fig. 12. Comparison between real and synthetic images generated with a Vox2Vox model. Examples of real test images (top row), corresponding synthetic images (middle row), and the SSIM computed between each real and each synthetic image (bottom row).

the background noise, especially in the red channel. Yet, training the model with only synthetic images produced impressive results (Fig. 13). For instance, the TPR and FDR obtained with 3DCtrDet Synthetic and 3DCtrDet Real are close (Fig. 13A,B). Moreover, for 3DCellPol, the results obtained with only synthetic data are also very similar to those obtained with only real data (Fig. 13). The slightly smaller TPR of 3DCellPol Synthetic can be explained by some nuclei that are not detected (Fig. 14(3)).

Regarding the results obtained with the combination of synthetic and real data, we concluded that 3DCtrDet Synthetic+Real outperforms 3DCtrDet Real in terms of TPR (Fig. 13A), Error (Fig. 13C), CS (Fig. 13D), ND (Fig. 13E) and GD (Fig. 13F). Nevertheless, these results are not as good as the ones obtained with 3DCellPol trained with synthetic and real data, demonstrating that the joint detection and pairing of centroids also improves the quality of the detected centroids. Comparing 3DCellPol Real and 3DCellPol Synthetic+Real, we concluded that combining synthetic and real images substantially improves the performance of 3DCellPol, the TPR increases (≈78% at $\tau_L = 0$) and the FDR decreases (Fig. 13A,B), while not compromising

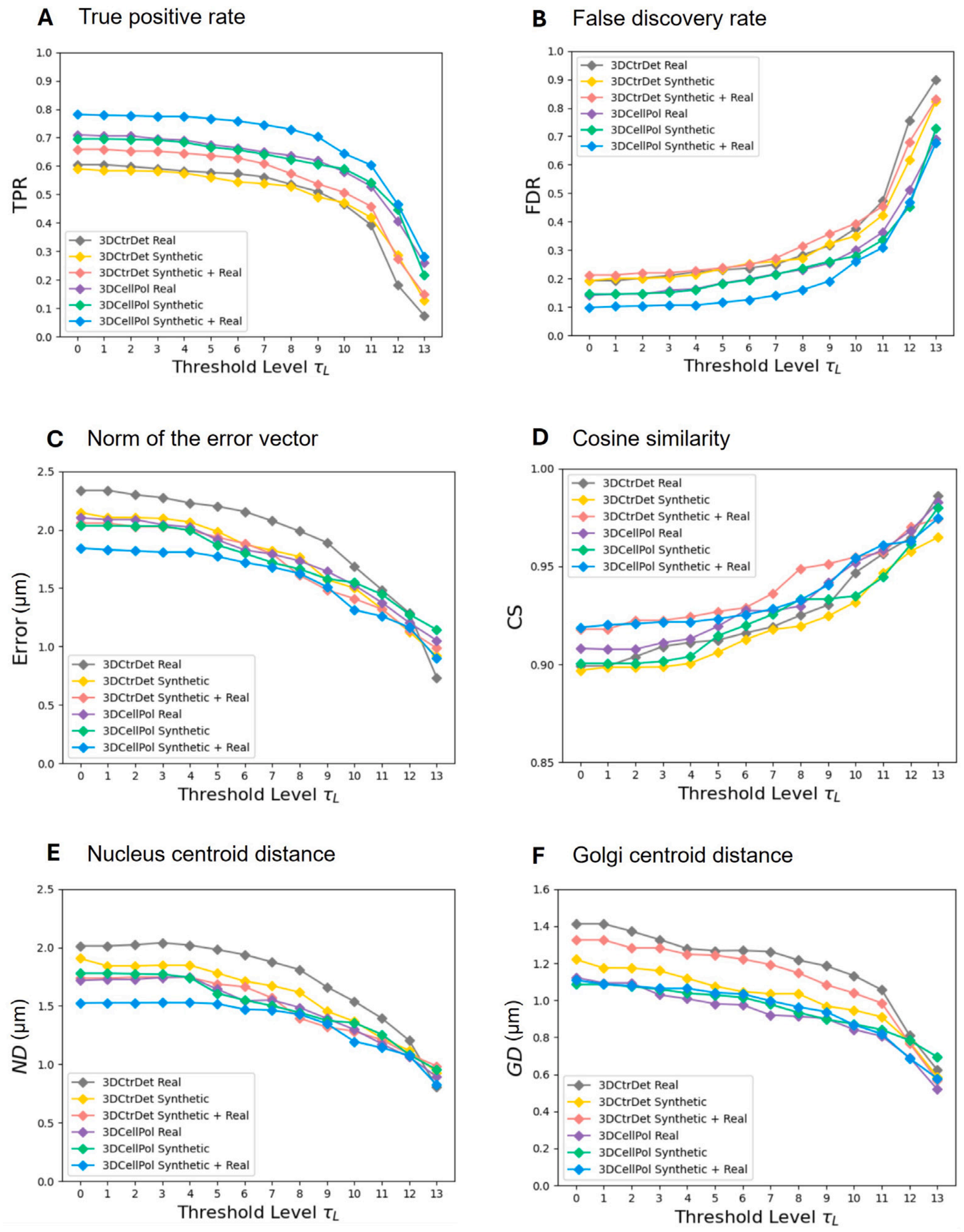

Fig. 13. Quantitative evaluation of the nucleus–Golgi vectors obtained using models trained on the real and/or synthetic nucleus–Golgi datasets. The methods compared are: 3DCtrDet Real, 3DCtrDet Synthetic, 3DCtrDet Synthetic+Real, 3DCellPol Real, 3DCellPol Synthetic, and 3DCellPol Synthetic+Real. Performance was evaluated using the following metrics: (A) True Positive Rate (TPR), (B) False Discovery Rate (FDR), (C) Norm of the Error Vector (μm), (D) Cosine Similarity (CS), (E) Nucleus Centroid Distance (ND) (μm), (F) Golgi Centroid Distance (GD) (μm).

the quality of the predicted vectors (Fig. 13C,D,E,F). Pre-training with synthetic data improves the detection of some nucleus–Golgi pairs, for instance, there is ground-truth pair that was not detected by 3DCellPol Real (Fig. 9(1)), but it is detected by both 3DCellPol Synthetic and 3DCellPol Synthetic+Real (Fig. 14(1)), and the predicted IP of this vector looks better than the IP of the ground-truth vector (Fig. 14(1)). Furthermore, 3DCellPol Real detects a nucleus as two nuclei (Fig. 9(3)) whereas 3DCellPol Synthetic and 3DCellPol Synthetic+Real correctly detect that nucleus as a single nucleus (yellow asterisk in Fig. 14(2)). Additionally, the vectors for the two close nuclei, which 3DCellPol Real did not identify correctly (Fig. 9(4)), are detected by 3DCellPol Synthetic and 3DCellPol Synthetic+Real (Fig. 14(3)).

For a more detailed qualitative evaluation, we present in Fig. 15 examples of ground-truth vectors and vectors predicted by 3DCellPol Real and 3DCellPol Synthetic+Real. In this Figure, we annotated improvements in the vectors predicted by 3DCellPol Synthetic+Real compared to 3DCellPol Real, and also identified some errors in the ground-truth. These improvements and errors are described at the bottom of the Figure.

3DCellPol Synthetic+Real detects more nucleus–Golgi pairs compared to 3DCellPol Real (Fig. 15), which supports the quantitative results shown in Fig. 13A. In this example, both 3DCellPol Real and 3DCellPol Synthetic+Real have a FN and a FP caused by errors in the ground-truth annotation. These errors correspond, respectively, to an incorrect nucleus–Golgi pair (Fig. 15(3)), and a pair not identified manually (Fig. 15(7)), which is correctly detected by both 3DCellPol Real and 3DCellPol Synthetic+Real. This shows how nucleus–Golgi vector assignment is difficult, even for experts.

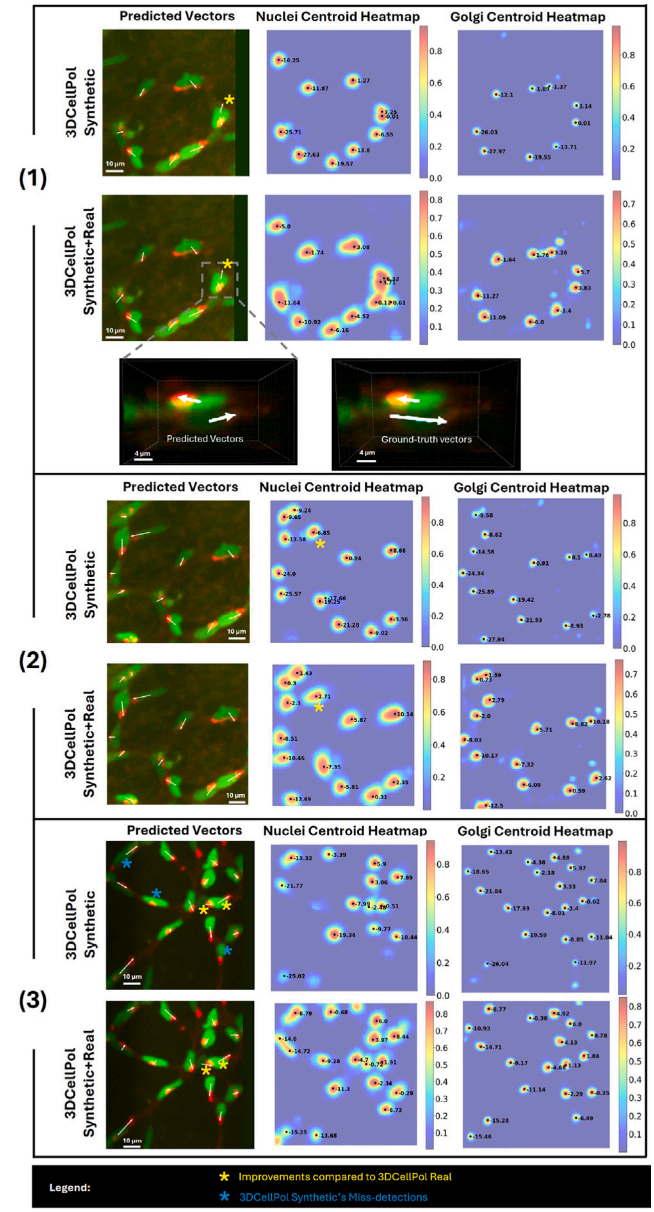

Fig. 14. Performance of 3DCellPol Synthetic and 3DCellPol Synthetic+Real. The first column represents the predicted vectors shown on top of 2D projections of input test patches. The second/third columns represent 2D projections of the output nuclei/Golgi centroid heatmaps, the local maximizers (black dots) and nuclei/Golgi embedding values at local maximizers, obtained with 3DCellPol Synthetic and 3DCellPol Synthetic+Real. Examples (1), (2), (3) can be compared to those obtained with 3DCellPol Real and shown in Fig. 9(1),(3),(4), respectively.

5.5. Cytoplasm-nucleus vectors prediction on the cytoplasm-nucleus dataset

Finally, we trained and tested 2DCellPol, 2DCtrDet and StarDist-2D using the cytoplasm-nucleus dataset [21]. The task of cytoplasm-nucleus assignment seems simpler than nucleus–Golgi pair prediction. In fact, the images shown in Fig. 4 suggest that the distance between nuclei and cytoplasm centroids is the main feature of cytoplasm-nucleus pairing. However, even in this dataset, our proposed approach outperforms multi-stage methods that perform pairing based on centroid distance. Specifically, 2DCellPol outperforms StarDist-2D regarding TPR and FDR (Fig. 16A,B). Additionally, 2DCellPol and 2DCtrDet are competitive. 2DCellPol is slightly better than 2DCtrDet regarding TPR, FDR, Error, CS and CytoD (Fig. 16A,B,C,D,E), whereas 2DCtrDet is slightly better in terms of ND (Fig. 16F).

The qualitative results of the top two methods on this dataset are presented in Fig. 17. These results demonstrate that some FP and FN are due to incorrect cytoplasm-nucleus assignment (green circles in Fig. 17A,B,C,D,E,F). This is mainly due to the existence of nuclei belonging to other cells (green circles in Fig. 17A,C,D,E). Both 2DCellPol and 2DCtrDet struggle in these scenarios, but 2DCtrDet performs worse. Moreover, some vectors detected by both models are not annotated in the ground-truth (blue circles in Fig. 17C). Finally, a slightly smaller TPR and bigger FDR of 2DCtrDet compared to 2DCellPol can be explained by overdetection of cytoplasm centroids (pink circles in Fig. 17C,E).

6. Discussion

We compared 3DCellPol's performance with multi-stage methods, which involve segmentation/detection of cell organelles and cost function minimization for vector computation [23], as well as with our previously proposed vector prediction approach [20]. Our quantitative results (Fig. 10) showed that 3DCellPol outperforms the automated approaches for nucleus–Golgi vectors prediction [20,23] in terms of TPR, FDR, Error, CS and ND. Particularly, 3DCellPol correctly detected 71% of nucleus–Golgi vectors, representing an approximate 10% improvement over the best previous methods (Fig. 10). For GD, 3DCellPol performs similarly to segmentation-based methods (StarDist-3D [27], 3D U-Net HS [20], and 3D U-Net [24]), but it is simpler, requiring only centroid ground-truth rather than labor intensive segmentation masks for training.

Our qualitative and quantitative results (Figs. 9, 10, and 11) demonstrate that 3DCellPol detects most nucleus–Golgi pairs. However, it has difficulty in identifying some pairs, mainly due to low-intensity nuclei or Golgi, overlapping nuclei that look like a single nucleus, dirt or air bubbles appearing as smaller high-intensity red objects, or noise caused by blood vessels in the red channel.

Next, we explored a way to overcome the issue of limited datasets in the biomedical field. We proposed a method to generate synthetic images based on a GAN that translates segmentation masks into synthetic images. However, this method does not require manually annotated segmentation masks since these are created automatically from the centroid annotations.

Analysis of the generated synthetic images revealed a high similarity with ground-truth real images (SSIM = 0.9097), as confirmed by qualitative results (Fig. 12 and Sup. Fig. 4). We then studied the impact of using synthetic images to train 3DCellPol. Specifically, we trained the 3DCellPol model considering only synthetic data, as well as a combination of real and synthetic data. The results, presented in Figs. 13 and 15, indicate that 3DCellPol performs better when trained with a combination of synthetic and real images, identifying 78% of nucleus–Golgi vectors. The model pre-trained on synthetic data learns to extract the main features for nucleus–Golgi pairing. These features are refined by fine-tuning with real data to better match the complexities of the nucleus–Golgi configurations in the real images. This model was evaluated on other microscopy images (Fig. 18), acquired under different experimental conditions. By observing Fig. 18, we can conclude that our approach can accurately predict most polarity vectors.

Finally, we trained and tested a 2D version of 3DCellPol, named 2DCellPol, on a publicly available cervical cytology dataset [21]. We used 2DCellPol to predict cytoplasm-nucleus vectors, which are important to study cell migration, division, and pathological conditions. 2DCellPol detected more than 90% of the vectors and outperformed 2DCtrDet and StarDist-2D.

Overall, these results demonstrated the ability of our approach to predict polarity vectors by associating different biological structures (nucleus–Golgi and cytoplasm-nucleus) in both 2D and 3D. Thus, we believe 3DCellPol and its 2D version can potentially be used to predict pairs of other biological structures, for instance mitochondria–Golgi [4], mitochondria–vacuole [4], endoplasmic reticulum–Golgi [4], and centrosome–nucleus [12], to study numerous physiological and pathological processes.

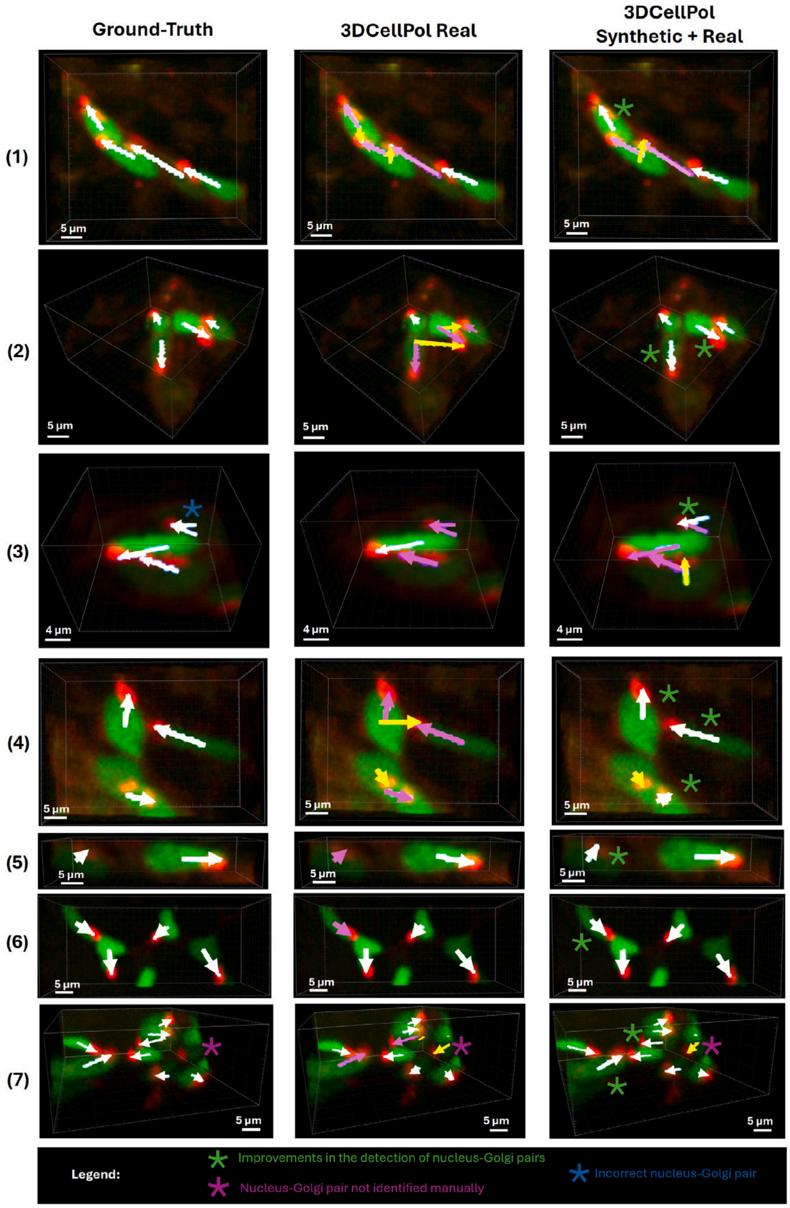

Fig. 15. Qualitative evaluation of the nucleus–Golgi vectors prediction results obtained with 3DCellPol Real and 3DCellPol Synthetic+Real. On the left: Examples of fluorescence microscopy image subvolumes (nuclei in green and Golgi in red) and the ground-truth nucleus–Golgi vectors represented in white. Second and third columns: Corresponding vector prediction evaluation results obtained with 3DCellPol Real and 3DCellPol Synthetic+Real, respectively. The white, yellow and pink vectors denote the TP, FP and FN, respectively. These TP, FP and FN correspond to $\tau_L = 5$. The results are superimposed on the microscopy patch. Some ground-truth errors, and improvements in the prediction of vectors are annotated using different colored asterisks as described at the bottom of the Figure.

7. Conclusions and future work

Cell polarity can be computed in biological images by detecting and pairing cell structures. Accurate cell polarity calculation is essential for understanding several developmental, physiological, and pathological processes [7]. To address this, we proposed 3DCellPol, a model that detects centroids of two distinct cell structures and simultaneously performs their pairing. Our approach exploits the strengths of a CNN, incorporating unique modifications that enable it to estimate 3D polarity vectors. 3DCellPol's robust ability to compute cell polarity makes it a valuable open-source tool for advancing our understanding of diseases like cancer and vascular disorders, where abnormal polarity plays a key role [7,9,10].

To demonstrate 3DCellPol's potential, we focused on 3D EC polarity, an important feature regulating blood vessel development [10]. EC polarity is computed as a vector between the centroids of two organelles: the nucleus and the Golgi complex. Our results showed that 3DCellPol outperforms multi-stage methods while requiring much less supervision, relying only on microscopy images and manually annotated nucleus–Golgi pairs. 3DCellPol's output vectors will be useful for vascular biologists studying EC polarization and migration patterns during blood vessel formation and remodeling, in homeostatic and pathological conditions. Additionally, a 2D version of 3DCellPol was trained and evaluated on a publicly available dataset to compute cytoplasm-nucleus polarity vectors. The results demonstrate superior performance over multi-stage methods. These vectors can assist cytotechnologists in improving cervical cancer diagnosis and proposing innovative treatments [21].

Although our approach performed robustly on both the real nucleus–Golgi and cytoplasm-nucleus datasets, it has limitations. It requires an Nvidia GPU for faster processing of large 3D images, and its performance is affected by background noise, especially in the red channel

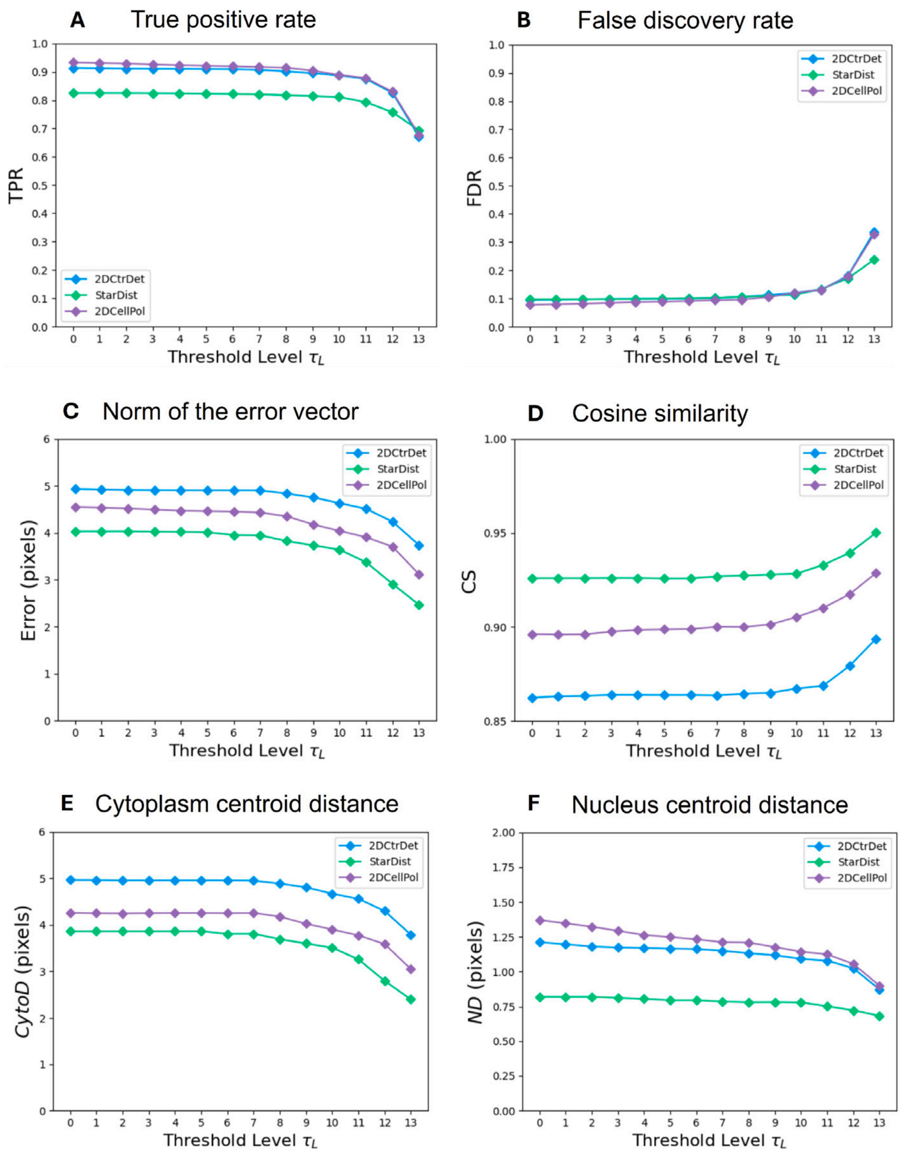

Fig. 16. Quantitative evaluation of the cytoplasm-nucleus vectors obtained using models trained on the cytoplasm-nucleus dataset. The methods compared are: StarDist-2D, 2DCtrDet and 2DCellPol. Performance was evaluated using the following metrics: (A) True Positive Rate (TPR), (B) False Discovery Rate (FDR), (C) Norm of the Error Vector (pixels), (D) Cosine Similarity (CS), (E) Cytoplasm Centroid Distance (CytoD) (pixels), (F) Nucleus Centroid Distance (ND) (pixels).

of the real nucleus–Golgi dataset. Future work could explore advanced computer vision techniques to reduce noise and enhance image quality.

Finally, although 3DCellPol was optimized for the specific dataset used during training, it can be easily fine-tuned for other datasets with minimal supervision, requiring only images and pairs of centroids. We are open to future collaborations to expand our approach by training it on additional nucleus–Golgi datasets from diverse conditions and laboratories. By providing more diverse training data, we believe we can continue to improve 3DCellPol's performance, generalization capability and applicability to new real-world data. Moreover, since 3DCellPol requires little supervision, it can be easily adapted to detect oriented pairs of other biological structures, such as nucleus–centrosome [12] and nuclei-FISH spots [5].

Thus, 3DCellPol will be of great importance to advance our understanding of cell polarization under physiological and pathological conditions. It will be a valuable tool to identify biomarkers of diseases caused by abnormal cell polarity, which can then be used by clinicians to improve diagnosis and therapeutic interventions. We believe 3DCellPol will be useful in both biomedical research and clinical practice.

CRediT authorship contribution statement

Hemaxi Narotamo: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Cláudio A. Franco:** Writing – review & editing, Supervision, Methodology, Data curation, Conceptualization. **Margarida Silveira:** Writing – review & editing, Supervision, Methodology, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT for grammar correction and improving the clarity of the text. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

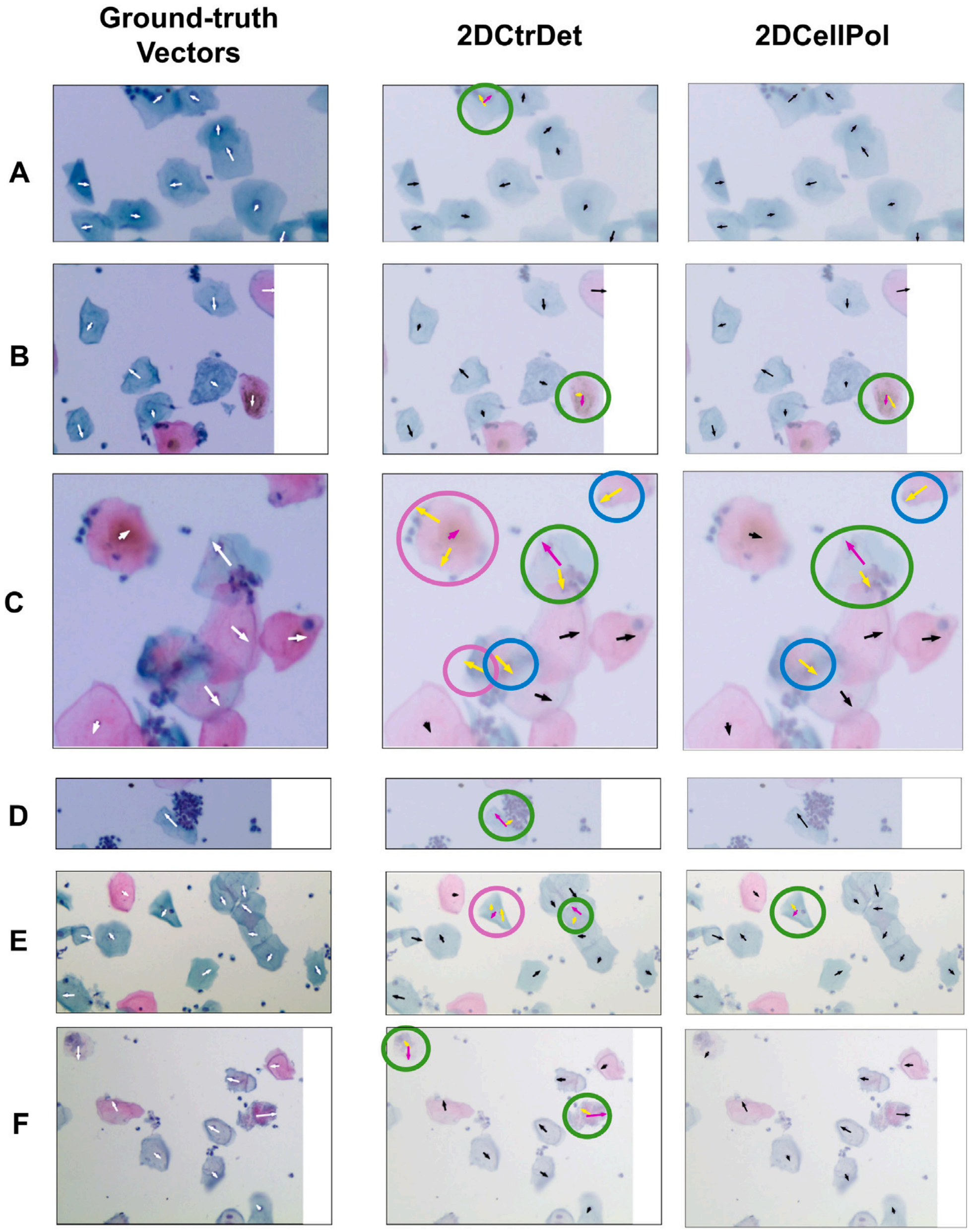

Fig. 17. Qualitative evaluation of the cytoplasm-nucleus vector prediction results obtained with 2DCtrDet and 2DCellPol. On the left are examples of test patches and the ground-truth vectors represented in white. In the middle and on the right: Corresponding vector prediction evaluation results obtained with 2DCtrDet and 2DCellPol, respectively. The black, yellow and pink vectors in these images correspond to TP, FP and FN, respectively, computed at $\tau_L = 5$ (see Table 6). The results are superimposed on the test image patches with a transparency of 50%. Error annotation legend: green circles — incorrect cytoplasm-nucleus assignment; pink circles — overdetection of cytoplasm; blue circles — predicted vectors not annotated in the ground-truth.

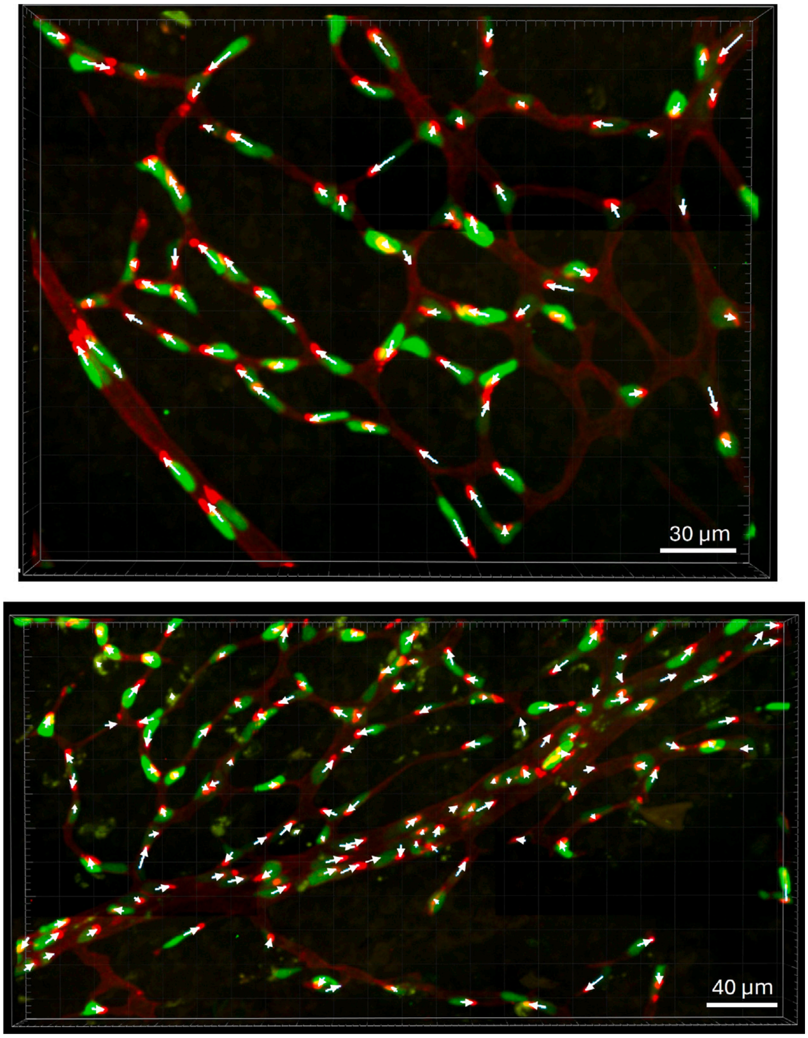

Fig. 18. 3DCellPol's generalization capability. Microscopy image (nuclei in green and Golgi in red) and nucleus–Golgi vectors predicted by the 3DCellPol pre-trained with synthetic data and fine-tuned with all real data (in white).

Acknowledgments

H.N was supported by the Fundação para a Ciência e a Tecnologia (FCT), Portugal Doctoral Grant 2020.04511.BD (DOI 10.54499/2020.04511.BD).

H.N and M.S were supported by LARSyS FCT funding (DOI: 10.54499/LA/P/0083/2020, 10.54499/UIDP/50009/2020, and 10.54499/UIDB/50009/2020).

C.A.F was supported by the FCT, Portugal funding (grants: PTDC/MED-ANM/7695/2020; 2022.09228.PTDC; CEECIND /02589/2018), European Research Council starting grant (679368), a grant from the Fondation LeDucq (17CVD03), and La Caixa Foundation (LCF/PR/HR22/52420027).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.bspc.2025.107537.

Data availability

The data and code are available in the repositories mentioned in the manuscript.

References

[1] A.M. Hafiz, G.M. Bhat, A survey on instance segmentation: state of the art, Int. J. Multimed. Inf. Retr. 9 (3) (2020) 171–189.

[2] C. Wang, F. Zhang, S.S. Ge, A comprehensive survey on 2D multi-person pose estimation methods, Eng. Appl. Artif. Intell. 102 (2021) 104260.

[3] N. Ahmad, J. Yoon, StrongPose: bottom-up and strong keypoint heat map based pose estimation, in: 2020 25th International Conference on Pattern Recognition, ICPR, IEEE, 2021, pp. 8608–8615.

[4] M. Rossini, P. Pizzo, R. Filadi, Better to keep in touch: investigating inter-organelle cross-talk, FEBS J. 288 (3) (2021) 740–755.

[5] C. O'Connor, Fluorescence in situ hybridization (FISH), Nat. Educ. 1 (1) (2008) 171.

[6] F. Xing, Y. Xie, H. Su, F. Liu, L. Yang, Deep learning in microscopy image analysis: A survey, IEEE Trans. Neural Netw. Learn. Syst. 29 (10) (2017) 4550–4568.

[7] M.E. Piroli, J.O. Blanchette, E. Jabbarzadeh, Polarity as a physiological modulator of cell function, Front. Biosci. (Landmark Ed.) 24 (2019) 451.

[8] W.J. Nelson, Remodeling epithelial cell organization: transitions between front–rear and apical–basal polarity, Cold Spring Harb. Perspect. Biol. 1 (1) (2009) a000513.

[9] M. Ouarné, A. Pena, C.A. Franco, From remodeling to quiescence: the transformation of the vascular network, Cells Dev. 168 (2021) 203735.

[10] A. Pena, M. Ouarné, C.A. Franco, Methods to quantify endothelial cell front-rear polarity in vivo and in vitro, Curr. Opin. Hematol. 28 (3) (2021) 208–213.
[11] C. Wang, K. Qu, J. Wang, R. Qin, B. Li, J. Qiu, G. Wang, Biomechanical regulation of planar cell polarity in endothelial cells, Biochim. Biophys. Acta (BBA) Mol. Basis Dis. 1868 (12) (2022) 166495.
[12] G.G. Luxton, G.G. Gundersen, Orientation and function of the nuclear–centrosomal axis during cell migration, Curr. Opin. Cell Biol. 23 (5) (2011) 579–588.
[13] M. Théry, V. Racine, M. Piel, A. Pépin, A. Dimitrov, Y. Chen, J.-B. Sibarita, M. Bornens, Anisotropy of cell adhesive microenvironment governs cell internal organization and orientation of polarity, Proc. Natl. Acad. Sci. 103 (52) (2006) 19771–19776.
[14] F.J. Calero-Cuenca, C.S. Janota, E.R. Gomes, Dealing with the nucleus during cell migration, Curr. Opin. Cell Biol. 50 (2018) 35–41.
[15] T.P. Lele, R.B. Dickinson, G.G. Gundersen, Mechanical principles of nuclear shaping and positioning, J. Cell. Biol. 217 (10) (2018) 3330–3342.
[16] C.A. Franco, M.L. Jones, M.O. Bernabeu, I. Geudens, T. Mathivet, A. Rosa, F.M. Lopes, A.P. Lima, A. Ragab, R.T. Collins, et al., Dynamic endothelial cell rearrangements drive developmental vessel regression, PLoS Biol. 13 (4) (2015) e1002125.
[17] C.A. Franco, M.L. Jones, M.O. Bernabeu, A.-C. Vion, P. Barbacena, J. Fan, T. Mathivet, C.G. Fonseca, A. Ragab, T.P. Yamaguchi, et al., Non-canonical Wnt signalling modulates the endothelial shear stress flow sensor in vascular remodelling, eLife 5 (2016) e07727.
[18] P. Barbacena, M. Dominguez-Cejudo, C.G. Fonseca, M. Gomez-Gonzalez, L.M. Faure, G. Zarkada, A. Pena, A. Pezzarossa, D. Ramalho, Y. Giarratano, et al., Competition for endothelial cell polarity drives vascular morphogenesis in the mouse retina, Dev. Cell 57 (19) (2022) 2321–2333.
[19] M.O. Bernabeu, M.L. Jones, R.W. Nash, A. Pezzarossa, P.V. Coveney, H. Gerhardt, C.A. Franco, PolNet: A tool to quantify network-level cell polarity and blood flow in vascular remodeling, Biophys. J. 114 (9) (2018) 2052–2058.
[20] H. Narotamo, M. Ouarné, C.A. Franco, M. Silveira, Joint segmentation and pairing of nuclei and golgi in 3D microscopy images, in: 43rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE, 2021.
[21] G. Liu, Q. Ding, H. Luo, M. Sha, X. Li, M. Ju, Cx22: A new publicly available dataset for deep learning-based segmentation of cervical cytology images, Comput. Biol. Med. 150 (2022) 106194.
[22] J. Liu, H. Fan, Q. Wang, W. Li, Y. Tang, D. Wang, M. Zhou, L. Chen, Local label point correction for edge detection of overlapping cervical cells, Front. Neuroinform. 16 (2022) 895290.
[23] P. Barbacena, M. Ouarné, J.J. Haigh, F.F. Vasconcelos, A. Pezzarossa, C.A. Franco, GNrep mouse: a reporter mouse for front–rear cell polarity, Genesis 57 (6) (2019) e23299.
[24] Ö. Çiçek, A. Abdulkadir, S.S. Lienkamp, T. Brox, O. Ronneberger, 3D U-net: learning dense volumetric segmentation from sparse annotation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2016, pp. 424–432.
[25] D. Joon Ho, C. Fu, P. Salama, K.W. Dunn, E.J. Delp, Nuclei segmentation of fluorescence microscopy images using three dimensional convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 82–90.
[26] C. Fu, S. Lee, D. Joon Ho, S. Han, P. Salama, K.W. Dunn, E.J. Delp, Three dimensional fluorescence microscopy image synthesis and segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 2221–2229.
[27] M. Weigert, U. Schmidt, R. Haase, K. Sugawara, G. Myers, Star-convex polyhedra for 3D object detection and segmentation in microscopy, in: The IEEE Winter Conference on Applications of Computer Vision, WACV, 2020, http://dx.doi.org/10.1109/WACV45572.2020.9093435.
[28] N. Lakmal Deshapriya, D. Tran, S. Reddy, K. Gunasekara, Centroid-unet: Detecting centroids in aerial images, in: The 42nd Asian Conference on Remote Sensing (ACRS2021), 2021.
[29] C. Payer, D. Štern, H. Bischof, M. Urschler, Regressing heatmaps for multiple landmark localization using CNNs, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2016, pp. 230–238.
[30] A. Newell, K. Yang, J. Deng, Stacked hourglass networks for human pose estimation, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, the Netherlands, October 11-14, 2016, Proceedings, Part VIII 14, Springer, 2016, pp. 483–499.
[31] H. Höfener, A. Homeyer, N. Weiss, J. Molin, C.F. Lundström, H.K. Hahn, Deep learning nuclei detection: A simple approach can deliver state-of-the-art results, Comput. Med. Imaging Graph. 70 (2018) 43–52.
[32] W. Xie, J.A. Noble, A. Zisserman, Microscopy cell counting and detection with fully convolutional regression networks, Comput. Methods Biomech. Biomed. Eng.: Imaging Vis. 6 (3) (2018) 283–292.
[33] Y. Xie, F. Xing, X. Shi, X. Kong, H. Su, L. Yang, Efficient and robust cell detection: A structured regression approach, Med. Image Anal. 44 (2018) 245–254.
[34] M. Lapierre-Landry, Z. Liu, S. Ling, M. Bayat, D.L. Wilson, M.W. Jenkins, Nuclei detection for 3D microscopy with a fully convolutional regression network, IEEE Access 9 (2021) 60396–60408.
[35] Y. Guo, O. Krupa, J. Stein, G. Wu, A. Krishnamurthy, SAU-net: A unified network for cell counting in 2D and 3D microscopy images, IEEE/ACM Trans. Comput. Biol. Bioinform. 19 (4) (2021) 1920–1932.
[36] R. Rojas, C.F. Navarro, G.A. Orellana, V. Castaneda, et al., 3D nuclei segmentation through deep learning, in: 2023 IEEE Conference on Artificial Intelligence, CAI, IEEE, 2023, pp. 309–310.
[37] A. Newell, Z. Huang, J. Deng, Associative embedding: End-to-end learning for joint detection and grouping, Adv. Neural Inf. Process. Syst. 30 (2017).
[38] H. Law, J. Deng, Cornernet: Detecting objects as paired keypoints, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 734–750.
[39] A.W. Harley, K.G. Derpanis, I. Kokkinos, Learning dense convolutional embeddings for semantic segmentation, in: International Conference on Learning Representations (ICLR), 2016.
[40] S. Suganyadevi, V. Seethalakshmi, K. Balasamy, A review on deep learning in medical image analysis, Int. J. Multimed. Inf. Retr. (2021) 1–20.
[41] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1125–1134.
[42] S. Kazeminia, C. Baur, A. Kuijper, B. van Ginneken, N. Navab, S. Albarqouni, A. Mukhopadhyay, GANs for medical image analysis, Artif. Intell. Med. 109 (2020) 101938.
[43] H. Zhao, H. Li, S. Maurer-Stroh, L. Cheng, Synthesizing retinal and neuronal images with generative adversarial nets, Med. Image Anal. 49 (2018) 14–26.
[44] A. Ghorbani, V. Natarajan, D. Coz, Y. Liu, Dermgan: Synthetic generation of clinical skin images with pathology, in: Machine Learning for Health Workshop, PMLR, 2020, pp. 155–170.
[45] C.A. Franco, J. Blanc, A. Parlakian, R. Blanco, I.M. Aspalter, N. Kazakova, N. Diguet, E. Mylonas, J. Gao-Li, A. Vaahtokari, et al., SRF selectively controls tip cell invasive behavior in angiogenesis, Development 140 (11) (2013) 2321–2333.
[46] H. Narotamo, M. Ouarné, C.A. Franco, M. Silveira, Synthetic generation of 3D microscopy images using generative adversarial networks, in: 44th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE, 2022.
[47] N. Otsu, A threshold selection method from gray-level histograms, IEEE Trans. Syst. Man Cybern. 9 (1) (1979) 62–66.
[48] H.W. Kuhn, The hungarian method for the assignment problem, Nav. Res. Logist. Q. 2 (1–2) (1955) 83–97.
[49] J.B. Roerdink, A. Meijster, The watershed transform: Definitions, algorithms and parallelization strategies, Fund. Inform. 41 (1, 2) (2000) 187–228.
[50] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, M. Pietikäinen, Deep learning for generic object detection: A survey, Int. J. Comput. Vis. 128 (2020) 261–318.
[51] L. Wu, S. Han, A. Chen, P. Salama, K.W. Dunn, E.J. Delp, RCNN-slicenet: A slice and cluster approach for nuclei centroid detection in three-dimensional fluorescence microscopy images, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3755–3765.
[52] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.
[53] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[54] F. Chollet, et al., Keras, 2015, https://keras.io.
[55] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-scale machine learning on heterogeneous systems, 2015, URL https://www.tensorflow.org/. Software available from tensorflow.org.