FOREST-GC: A conFOrmable Rendering Engine for Synthetic Tree Generation and Counting

FOREST-GC: A conFOrmable Rendering Engine for Synthetic Tree Generation and Counting

Luciano Prono, *Member, IEEE*, Najmeddine Dhieb, *Member, IEEE*, Philippe Bich, *Member, IEEE*, Chiara Boretti, *Member, IEEE*, Fabio Pareschi, *Senior Member, IEEE*, Marco Brini, Hakim Ghazzai, *Senior Member, IEEE*, Riccardo Rovatti, *Fellow, IEEE*, and Gianluca Setti, *Fellow, IEEE*

***Abstract*—Accurate tree counting from satellite imagery remains a critical challenge for agricultural monitoring, climate analysis, and land-use management. While recent advances in deep learning, particularly large pretrained vision-language models (VLMs), have achieved remarkable results in general visual tasks, we show that they fail to reliably address tree counting in high-resolution satellite imagery. This highlights the need for specialized datasets to support the development and evaluation of models capable of solving this task, especially in plantation environments. To this end, we introduce a configurable synthetic image generator that produces realistic satellite imagery of olive plantations along with precise annotations in terms of the number and positions of trees, providing valuable resources for advancing tree counting research in agricultural settings. The generator simulates terrain textures, spatial distributions of trees, and tree shadows to approximate real-world conditions and exposes over 100 customizable parameters to control appearance and layout. It also includes a simulated near-infrared layer to support vegetation-specific spectral analysis. We compare deep learning methods based on object detection or segmentation models trained on the generated data with pretrained VLMs and an unsupervised computer vision baseline, and show that synthetic training achieves superior accuracy on a manually annotated real-world dataset, demonstrating effective sim-to-real performance transfer specifically in olive plantation imagery. In particular, SAM-ViT-large achieves an average relative error of 4.82% on the real data, which is comparable to the performance of human labelers. The generator source code is publicly available, supporting further research in data-scarce agricultural monitoring with a focus on plantation analysis.**

***Index Terms*—Computer vision, deep learning, olive plantations, precision agriculture, remote sensing, satellite imagery, synthetic data generation, tree counting, vegetation detection.**

Received 11 November 2025; revised 28 January 2026; accepted 24 February 2026. Date of publication 9 March 2026; date of current version 26 March 2026. This work was supported in part by the King Abdullah University of Science and Technology (KAUST) – Center of Excellence for Sustainable Food Security, under Award 5934-10-01. *(Corresponding author: Luciano Prono.)*

Luciano Prono, Philippe Bich, Chiara Boretti, and Gianluca Setti are with the Department of Electronic and Telecommunication, Politecnico di Torino, 10129 Torino, Italy (e-mail: luciano.prono@polito.it; philippe.bich@polito.it; chiara.boretti@polito.it; gianluca.setti@polito.it).

Najmeddine Dhieb and Hakim Ghazzai are with the King Abdullah University of Science and Technology (KAUST), Thuwal 23955, Saudi Arabia (e-mail: najmeddine.dhieb@kaust.edu.sa; hakim.ghazzai@kaust.edu.sa).

Fabio Pareschi is with the Department of Electronic and Telecommunication, Politecnico di Torino, 10129 Torino, Italy, and also with the Advanced Research Center on Electronic Systems (ARCES), University of Bologna, 40125 Bologna, Italy (e-mail: fabio.pareschi@polito.it).

Marco Brini is with SAI Platform, 1204 Geneva, Switzerland (e-mail: mbrini@saiplatform.org).

Riccardo Rovatti is with the Department of Electrical, Electronic, and Information Engineering, University of Bologna, 40136 Bologna, Italy, and also with the Advanced Research Center on Electronic Systems (ARCES), University of Bologna, 40125 Bologna, Italy (e-mail: riccardo.rovatti@unibo.it).

Digital Object Identifier 10.1109/JSTARS.2026.3671468

I. Introduction

MONITORING and counting trees is a fundamental task in environmental management, ecological research, and precision agriculture [1]. Accurate tree inventories are essential for assessing forest health, estimating biomass and carbon sequestration, and guiding reforestation efforts. In agriculture, tree counting plays a critical role in tree-based farming systems, supporting tasks, such as yield estimation, irrigation planning, and disease monitoring [2]. For crops like olive trees, which are often cultivated in semiarid regions and hold both economic and cultural significance, timely and accurate counting is essential to optimize resource allocation and support informed decision-making by farmers. Moreover, large-scale tree counting contributes to global climate monitoring efforts by enabling better land cover classification and tracking deforestation or land degradation over time.

Various methods have been employed to perform tree counting and monitoring, ranging from traditional field-based techniques to advanced remote sensing technologies [3]. Manual counting, while accurate at small scales, is labor-intensive, time-consuming, and impractical for large or remote areas. Ground-based sensors and IoT devices offer automated monitoring capabilities but require substantial infrastructure and maintenance, making them less feasible for widespread deployment [4], [5]. In recent years, uncrewed aerial vehicles (UAVs) have gained popularity due to their high-resolution imaging and flexibility in capturing data over targeted areas [6]. UAV-based approaches have shown considerable promise for tree counting tasks, especially due to their ability to capture high-resolution imagery and operate at low altitudes for precise visual data collection. For example, TasselNetV3 [7] introduced an explainable plant counting framework using UAV-captured images, where local count regression and background suppression significantly improved accuracy and interpretability across various crop types, including maize and wheat. Similarly, TreeFormer [8] proposed a semisupervised transformer-based model for tree counting that leverages high-resolution UAV imagery with limited annotations. It combines pyramid vision transformers with contextual attention modules to produce density maps and global count estimates, demonstrating superior performance over state-of-the-art

baselines in multiple datasets. These works illustrate the growing effectiveness of UAV-based deep learning methods for precision agriculture tasks, such as tree counting, though their scalability remains constrained by flight range, battery limitations, and operational costs for large-scale monitoring.

As a complementary and innovative solution, satellite imagery has emerged as a promising approach showing great potential in many related domains [9], [10], [11], [12]. With increasing spatial and spectral resolution, satellite data can offer consistent, large-scale coverage over diverse terrains and time periods. This opens new possibilities for tree counting across expansive agricultural landscapes, particularly when high-frequency, cost-effective monitoring is required. High-resolution satellite imagery offers several compelling advantages for the task of tree counting, particularly in large and remote agricultural areas. With spatial resolutions reaching submeter levels, as seen in commercial satellites, such as WorldView-3, GeoEye-1, and Pleiades Neo, individual tree crowns can often be distinguished, enabling object-level analysis over vast regions. Unlike UAVs or ground-based methods, satellites can cover thousands of square kilometers in a single pass, allowing for efficient, consistent, and repeatable monitoring at scale. The inclusion of multispectral bands, especially the near-infrared (NIR), further enhances vegetation differentiation through indexes, such as the normalized difference vegetation index (NDVI) and the green atmospherically resistant index, which can highlight the presence and health of trees. In addition, satellites operate on predictable orbits and provide frequent revisit times, making them ideal for temporal analysis and long-term monitoring. These advantages make satellite imagery a valuable and scalable data source for tree counting tasks, especially when paired with robust image processing techniques and artificial intelligence (AI)-driven models [13], [14].

Despite the growing potential of satellite imagery for tree counting, some challenges still limit its effectiveness in practice. Distortions caused by shadows, varying illumination conditions, and topographic irregularities can significantly affect image quality and consistency [15]. Shadows may obscure parts of the canopy, while changes in lighting across a scene can introduce spectral variations that confuse vegetation detection. Grass patches on the terrain further contribute to misclassification. In regions with uneven terrain, topographic distortion can misrepresent the true spatial arrangement of trees, reducing the reliability of spatial analysis. However, one of the most critical obstacles to advancing tree counting methods is the lack of labeled datasets or ground truth annotations. This scarcity of annotated data makes it difficult to develop, train, and validate high-quality supervised models.

To help overcome this limitation, we propose the conformable rendering engine for synthetic tree generation and counting (FOREST-GC), a synthetic satellite image generator for tree counting.[1] The dataset generator can be tuned to closely resemble real satellite images, and its parameters are fine-tuned both through visual inspection and by measuring geometrical properties and the color of the elements in the scene. Unlike existing dataset generators, which typically produce only RGB data, the proposed generator also synthesizes the NIR channel jointly with the RGB channels. Synthetic images come with the positions and size of the generated trees, enabling its use for supervised training and validation. The dataset generator is designed for fast execution on a graphics processing unit (GPU) and easy accessibility, being implemented in Python, and it is publicly available. Once we produce the synthetic dataset, we train multiple different computer vision neural models to evaluate the utility of the generated images for enabling sim-to-real transfer to real satellite imagery. We train deep neural models, ranging from object detectors to segmentation networks, and we analyze their performance to assess the potential of the synthetic data for accurate tree counting and localization. Specifically, the generator has been crafted and evaluated for olive trees plantations. This approach could be adapted to other types of plantations as well, although this would require further analysis across different scenarios.

To provide a baseline comparison for the neural-based approaches, we also evaluate an unsupervised "vegetation-based" method that leverages classical computer vision techniques to isolate vegetated regions and estimate the number of trees based on geometric and spatial analysis. Compared to the proposed neural models trained on our generated images, this method struggles to adapt to complex and irregular tree structures and often fails to distinguish trees from shadows and grass patches. In addition, we evaluate pretrained vision-language models (VLMs), which also struggle to provide accurate counts in this context due to their limited capabilities in regression tasks. Furthermore, to assess the reliability of the FOREST-GC-based approach, we benchmark all models on a manually annotated real-world dataset of high-resolution satellite imagery. Experimental results demonstrate that models trained exclusively on synthetic data achieve competitive performance when applied to real images, highlighting the effectiveness of FOREST-GC for enabling sim-to-real generalization in practical tree counting scenarios.

The novel contributions of this work can be summarized as follows.

1) We present a computationally efficient, PyTorch-based synthetic tree field generator designed for seamless integration into deep learning pipelines. The generator is tailored for tree counting, includes NIR channel generation, and allows precise color control of each generated element.
2) We employ the generated synthetic dataset to train and evaluate different deep learning models for tree counting.
3) We validate the proposed approach on manually annotated real-world satellite images, demonstrating that models trained exclusively on synthetic data can effectively generalize to real scenarios.
4) We show that large pretrained VLMs and nonlearning approaches are insufficient for reliable tree counting, confirming the need for specialized training data.

The rest of this article is organized as follows. Section II introduces closely related works and how our approach differs

[1]The proposed generator. [Online]. Available: https://github.com/SSIGPRO/albero-system

from the state-of-the-art. Then, we define the tree counting problem in Section III. Section IV discusses the generation of the synthetic satellite image dataset and the neural-based methods used for counting. Next, we report and discuss the performance of the proposed approach in Section IV-D. Finally, Section VII concludes this article.

II. Related Works

Tree counting in remote sensing has attracted considerable attention across agricultural and ecological domains, with methods diverging based on sensing platforms (e.g., UAVs, satellites), spectral inputs (e.g., RGB, multispectral), and learning paradigms (supervised versus unsupervised). A significant body of research focuses on satellite imagery for identifying and enumerating olive trees due to their agricultural importance and relatively distinct morphology.

One of the earliest studies in this domain was conducted by Gonzales et al. [16], who exploited the reticular planting pattern of olive trees to facilitate tree counting. Using QUICKBIRD satellite imagery, they extracted geometric features, such as shape, size, and the angles between trees within the grid. Tree detection was based on the joint probability of a candidate tree matching both the expected geometric parameters and the reticular pattern. This approach achieved an average detection rate of 98%, but its effectiveness is limited to regularly planted orchards and performs poorly in irregular planting arrangements. Ioannis et al. introduced the Arbor Crown enumerator method [17], a tree crown detection algorithm for multispectral imagery acquired with the QUICKBIRD satellite over the Keritis watershed in the island of Crete. This algorithm combined the NDVI [18] and red band-based thresholding with blob detection, achieving promising results, with an estimated error of 1.3%.

Around the same years, Yakoub et al. [19] developed an automatic olive tree counting methodology using morphological features extraction followed by classification via a Gaussian process classifier. The output of this approach is a binary classification map that differentiates the olive trees from ground object data. Trained and tested on IkONOS-2 satellite imagery from agricultural regions in Saudi Arabia, the method yielded high overall accuracy. However, the limited size of the training dataset left room for further refinement. Moreno-Garcia et al. in [20] proposed a K-Means clustering algorithm to segment olive trees starting from very high resolution satellite imagery from Spain. While the method showed promising results, it struggled with identifying overlapping tree crowns and lacked diversity in testing data, affecting generalizability. Peters et al. [21] employed an object-based classification approach over a study area in France, integrating data from multiple sensors (ADS40 digital sensor, RAMSES radar, and TerraSAR-X satellite). The decision-based accuracy was obtained as 88%, relatively lower in comparison to the previous techniques.

Several years later, Chemin and Beck [22] presented a method that utilized large-format photogrammetric images with the NIR band to count olive trees in Apulia, Italy. The proposed methodology involves a preprocessing pipeline to convert the images into NDVI maps, followed by a segmentation step based on Niblack's thresholding and Sauvola binarization techniques [23]. The approach showed promising results, achieving an overall mean error of 13%.Khan et al. [24] proposed a multilevel thresholding technique for segmenting olive trees from RGB images obtained via the SIGPAC viewer in Spain, reaching 96% accuracy. In a subsequent study by the same authors [25], the method was further refined. The approach leverages solely the red band extracted from the original imagery. The pipeline involves image sharpening followed by edge detection, with closed edge contours representing tree boundaries subsequently converted into white blobs using morphological reconstruction. These blobs are then filtered based on shape and size, allowing circular blobs within a specified radius range to be identified as olive trees. The improved method, validated against ground truth data from various image sources, demonstrated a remarkably low estimation error of just 1.27%.

More recently, Mezzi et al. [26] developed a hybrid algorithm that combines mean-shift segmentation, spectral thresholding (red band and NDVI), and spatial filtering to detect and count olive trees. Tested across four sites in the Sbikha region of Kairouan, Tunisia, the approach achieved a mean accuracy of approximately 87%. Most notably, Abozeid et al. [27] introduced a deep learning-based approach (SwinTUnet) utilizing only RGB bands for tree detection and counting. Alongside their algorithm, they also released a large-scale olive tree dataset comprising 230 RGB satellite images from the Al-Jouf region in Saudi Arabia. SwinTUnet achieved a state-of-the-art estimation error of just 0.94%, demonstrating the effectiveness of transformer-based deep learning models in remote sensing applications.

Previous studies have explored various techniques for the automatic detection of olive trees, ranging from basic image segmentation and blob detection to more advanced classification approaches. These methods have leveraged both satellite and aerial imagery, incorporating spectral data from grayscale and RGB. While many reported high accuracy, they also shared common limitations, most notably, the lack of publicly available datasets with sufficient image volume and diversity for comprehensive testing. Accuracy often improved with richer spectral information, but some methods were prone to false detections, especially in scenarios involving densely planted trees with overlapping canopies and varied tree sizes. These challenges highlight the need to design automated solutions to generate datasets that closely resemble real-world satellite imagery, which are essential for fairly training and comparing high-quality neural network-based approaches.

Although various studies have followed this line of research [28], [29], [30], most available dataset generators are not tailored to tree counting tasks, do not include the generation of the NIR channel, and are not suited to generate large numbers of annotated images on the fly or need themselves annotated images to train the image generator.

III. Problem Definition and Working Pipeline

The following formulation describes a standard regression problem commonly encountered in remote sensing applications. Let $\mathcal{I}$ denote the space of multispectral satellite images, where

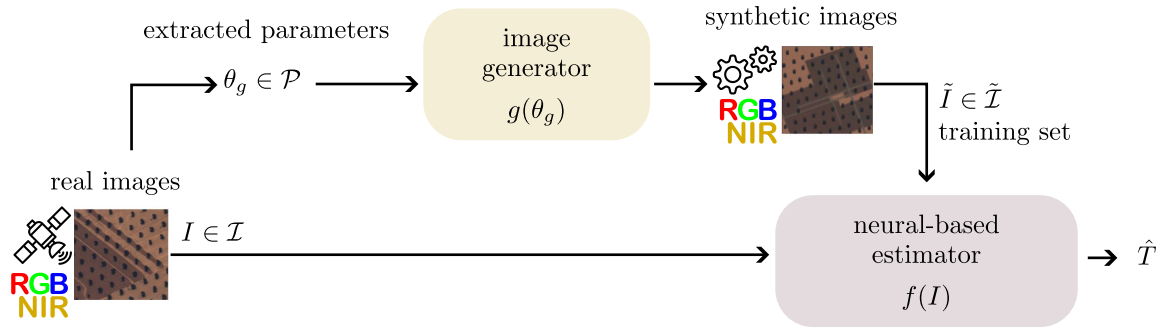

Fig. 1. Schematic overview of the neural-based estimator $f_n(I)$ trained on synthetic images to predict tree numbers $\hat{T}$. Real satellite images (RGB+NIR) are used to extract the generation parameters of the synthetic image generator g. The estimator's accuracy is evaluated on a small dataset of manually annotated real satellite images.

each image $I \in \mathcal{I}$ is defined over four spectral channels: 1) red; 2) green; 3) blue; and 4) NIR. The goal is to design a system f that maps an input image I to the number of olive trees T present in the scene as expressed in the following:

$$f : \mathcal{I} \to \mathbb{N}^+, \quad \hat{T} = f(I) \tag{1}$$

where $\hat{T}$ is the output of $f(I)$, i.e., the estimated number of trees. This type of problem is challenging for pretrained supervised models, primarily because annotated datasets for this specific task are scarce.

Therefore, to enable the training and evaluation of such a system, especially under limited access to annotated real-world imagery, we introduce our synthetic image generation module (FOREST-GC) denoted as g. This generator is parameterized by a certain number of values $\theta_g \in \mathcal{P}_g$, which control aspects, such as tree density, spatial distribution, background texture, illumination conditions, and color palette of the fabricated images. The generator outputs synthetic images $\tilde{I} \in \tilde{\mathcal{I}}$ that approximate the distribution of real-world data

$$g : \mathcal{P} \to \tilde{\mathcal{I}}, \quad \tilde{I} = g(\theta_g). \tag{2}$$

To ensure the realism and utility of the generated data, we calibrate the generator g using a small set of real satellite images $\{I^{(1)}, \ldots, I^{(n)}\} \subset \mathcal{I}$. We observe and measure the geometrical characteristics of the real images and we extract color palettes so that the generated image $g(\theta_g)$ closely resembles the statistical and spectral characteristics of the calibration images. This kind of optimization is validated a-posteriori by using a similarity index to measure the distance between features extracted from real and synthetic images, enabling more effective sim-to-real transfer.

We use this synthetic image generator to produce a training dataset used to fine-tune a neural-based estimator with a supervised learning approach, trained to detect individual trees from which the tree count is obtained. Then, while we perform a first evaluation of the neural models on the synthetically generated dataset, we perform a final evaluation on real satellite images to demonstrate that learning performed on fabricated data can be transferred to realistic scenarios. Fig. 1 visually illustrates the working pipeline.

IV. Synthetic Dataset Generation and Neural-Based Counting

We design FOREST-GC generator framework to meet several key requirements. First, we need to integrate it seamlessly with a Python-based optimization framework. Second, it has to be fast enough to generate large datasets within a short time, allowing to fine-tune its parameters to make it as similar as possible to the real images. Because of this, the generator is implemented with the PyTorch[2] computational framework in order to leverage the GPU computational power to speed-up the generation speed. FOREST-GC is able to render a 640×640 image in half a second on a Nvidia A100 GPU, without relying on a custom backend.

In addition, two assumptions are used to improve the generation pipeline computational efficiency, namely, we hypothesize that trees are disposed in a grid-like pattern and that they are clustered in rectangular fields. This does not impair the generalization capabilities of the generator since 1) the exact position of each tree can be modified by adding a random offset; and 2) the final images are cropped, so the information on the exact shape of the fields is lost. These assumptions are well aligned with agricultural and semistructured landscapes characterized by sparse and spatially separated trees, such as olive groves, which are the primary target of this work. However, they may be less suitable for modeling very dense and highly irregular tree distributions, as found in natural forests. To further enhance realism, the generator parameters are calibrated using statistics extracted from a real satellite dataset, including tree spacing, size distributions, and visual appearance, ensuring that the synthetic images closely reflect real plantation conditions.

A. Tree Field Generation

The tree generation pipeline follows a multistep process. First, an empty canvas is generated and colored with a noisy background. We also add random rectangular patches of different colors to emulate different terrain types. Also, some of these patches are forced not to have trees over them, to emulate roads and field borders.

[2][Online]. Available: https://pytorch.org/

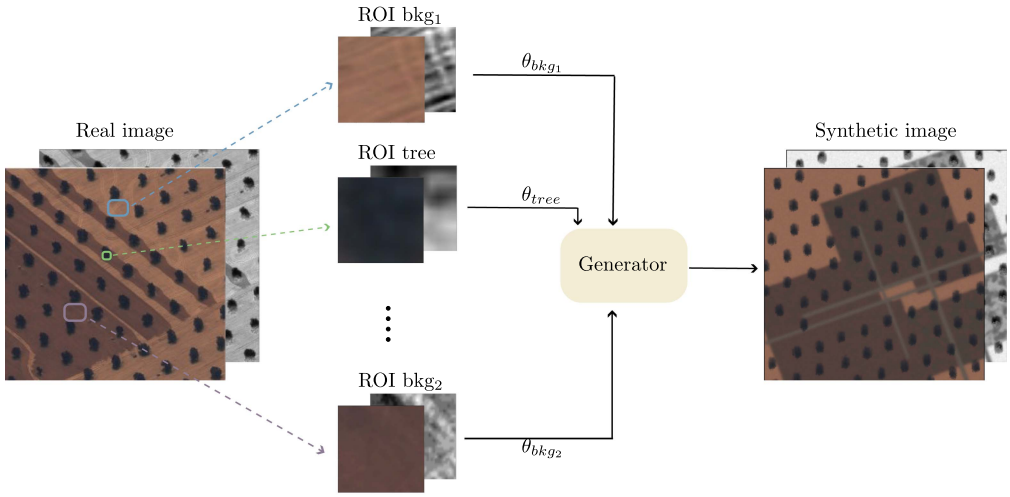

Fig. 2. Color optimization pipeline for aligning synthetic images with real satellite imagery. A patch is extracted from a real image, and each spectral channel value (R, G, B, NIR) is computed as the average of the values in the patch. In this example, a RoI is manually selected for different components in the image, and then all the obtained parameter values are given to the generator to create a synthetic image.

Then, a map of the trees is generated by using a random Gaussian matrix smoothed with a Gaussian blur filter. This results in a matrix where each value corresponds to a tree. If the value is larger than a threshold, the tree is planted, the field position is left empty otherwise. After this, the tree sprites are generated, following the map of the trees. The shape of each tree is generated by using a noisy radius smoothed with a Gaussian blur filter. The larger the value in the map of the trees, the larger the average radius of the tree.

Before applying the sprites on the canvas, sprite-based drop shadow rendering is used to generate the tree shadows, i.e., the tree sprites are shifted and printed multiple times on the canvas by using the shadow color. This step is fundamental since the presence of the shadows of the trees deeply influences the performance of the counting algorithms. Finally, the tree sprites are applied to the canvas. As a postprocessing step, the canvas is rotated and then cropped to avoid border effects. The process produces, along with the synthetic images, the position and size of each tree and the total number of trees.

In order to make the generated images as similar as possible to real satellite imagery, multiple generation parameters are carefully tuned. The generator exposes more than 100 parameters that can be configured to control the appearance and layout of the simulated scenes. Further details about these parameters are provided in Appendix A. To match the geometrical properties of real plantations, we measure the distances and sizes of trees in actual satellite images and adjust the corresponding parameters in the generator accordingly. In addition, we inspect the shape of the trees to tune the noisiness and smoothness of their contours using a trial-and-error approach, aiming to reproduce tree shapes that visually resemble those observed in real imagery.

Since the generation process is fully controlled, all annotations are obtained directly during synthesis. Each tree is handled as an individual object throughout the generation pipeline, which allows the total number of trees in each scene to be directly inferred without additional postprocessing. Moreover, because tree sprites are rendered independently, pixel-level masks can be extracted from the nontransparent pixels associated with each tree, yielding precise spatial annotations for downstream tasks.

B. Color Optimization

Colors play an important role in making the generated images as close as possible to the real ones. To cope with this, we define a pipeline to distill the color of the different elements of the synthetic images, and we determine the optimal values for the red, green, blue, and NIR channels. We begin by selecting several sample images that are representative of the real dataset. For each configurable element in the generator, such as trees, soil, roads, field boundaries, and shadows, we manually define corresponding regions of interest (ROIs). The values for each channel (red, green, blue, and NIR) are then calculated by averaging the respective channel intensities within each selected ROI. By using this approach, multiple color palettes are defined and then randomly picked during dataset generation.

Fig. 2 shows the color optimization pipeline. By using this synthetic pipeline, we produce a dataset composed of 3502 training images ($\sim 70\%$), 503 validation images ($\sim 10\%$), and 1030 test images ($\sim 20\%$).

C. Quality Validation of the Synthetically Generated Images

The similarity between the synthetically generated images and the real ones is validated by computing a widely used image similarity metric, namely structural similarity index measure (SSIM) [31] across all four channels (RGB and NIR). This evaluation was conducted on two sets of image pairs: 1) 50 randomly selected pairs of real images, to establish the inherent similarity level within the real image set; 2) 50 pairs consisting of one real image and one synthetically generated image, to assess how effectively the synthetic images replicate the structural characteristics of the real ones.

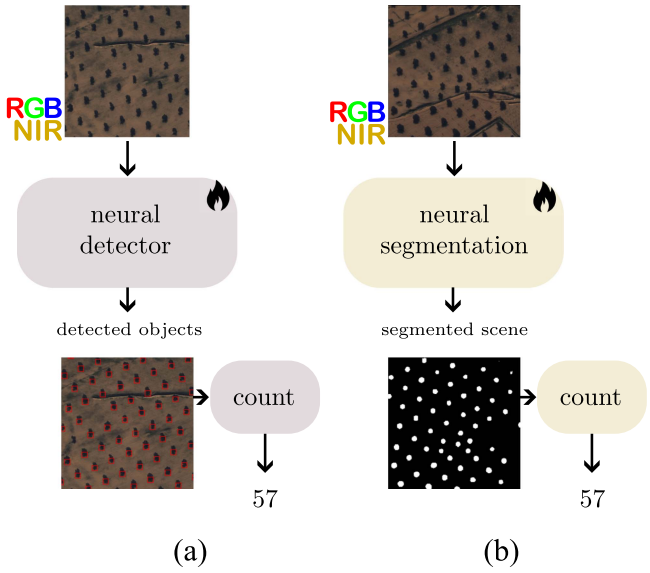

Fig. 3. Overview of the proposed methods for neural-based olive tree counting. The first approach (a) is based on neural object detectors (YOLO) while the second (b) is based on neural segmentation models (SAM-ViT). Both are fine-tuned on the synthetic satellite image dataset.

For the real–real image pairs, the mean SSIM is 0.375 with a variance of 0.124, indicating that the real images set exhibits variability in structure and content, as reflected by an average similarity score below the maximum value that SSIM could assume (i.e., SSIM $\in [-1, 1]$). On the other hand, the real-synthetic image pairs achieve a higher mean SSIM of 0.462 and a lower variance of 0.103. This result suggests that the synthetic images consistently reproduce the structural patterns in the real data, resulting in higher overall similarity scores. At the same time, the fact that the SSIM value is higher for real-synthetic pairs compared to real–real pairs suggests that synthetic images adapt well only to the *average* appearance of real images, while there is still room for improvement to better simulate the variability and noisiness of the real images. Fig. 4 shows some examples of real and synthetically generated images used to perform this analysis.

D. AI-Based Counting Algorithms

Leveraging the availability of annotated data coming from our synthetic satellite image generator, we explore three neural network-based approaches to estimate the number of olive trees in plantation areas.

In this work, we adopt two complementary supervised learning strategies to address the tree counting task. The first relies on object detection neural networks, specifically models from the YOLO family [32], a family of single-shot detectors known for their fast inference and high accuracy in object localization tasks. These networks predict bounding boxes corresponding to individual trees, and the total tree count is obtained by aggregating the number of detected boxes in each tile. The second strategy is based on segmentation models, in particular the UNet [33] and the segment anything model (SAM) [34]. Specifically, the employed UNet architecture uses a ResNet34 backbone [35] encoder for feature extraction, while decoder layers follow the standard UNet design with skip connections. SAM is instead a recent transformer-based architecture designed for segmentation across diverse domains, which has been successfully applied to other remote sensing tasks [36], even when datasets are lacking [37]. These models generate binary masks where each connected component will correspond to a predicted tree. Counting the connected components in the mask provides the final estimate.

Compared to direct regression networks that output a scalar count, these approaches also offer a degree of explainability, as they allow for visual inspection of which regions the models consider to be trees. This interpretability is particularly valuable when assessing performance and understanding potential failure cases. Importantly, these methods are feasible because the synthetic data generator not only provides the total number of trees per image but also precise spatial annotations of each tree's location and size. This enables the generation of training labels suitable for both object detection and segmentation models. Fig. 3 illustrates these strategies.

V. Experiments and Results

In this section, we report the performance of the various tree counting approaches on the data generated by our image simulator. We then conduct an analysis of the outcomes when real-world data is used. To complete the analysis, we also benchmark the proposed approach against a nonlearning approach and a VLM.

A. Baseline Approaches

To further assess the efficacy of the synthetic dataset approach, we perform additional tests on an unsupervised vegetation-based counting method, formulated as a practical computer vision pipeline. This method leverages spectral and spatial analysis to identify vegetated regions and infer the number of trees within them. It is particularly suited for scenarios where annotated datasets are unavailable, offering a solution for vegetation analysis.

The pipeline begins by calculating the NDVI, a widely recognized metric in remote sensing, defined as

$$\text{NDVI} = \frac{\text{NIR} - \text{Red}}{\text{NIR} + \text{Red}} \tag{3}$$

where NIR represents the NIR channel and Red corresponds to the red channel. NDVI highlights vegetated areas based on their spectral properties, with higher values indicating dense vegetation. A threshold is applied to the NDVI values to delineate vegetation regions from other elements in the scene, such as soil or shadows, ensuring a clear separation of vegetated areas for further analysis.

Once the vegetation regions are identified, morphological operations are applied to refine the mask and remove noise, ensuring that only meaningful vegetation clusters are retained. Specifically, a morphological opening operation using a small

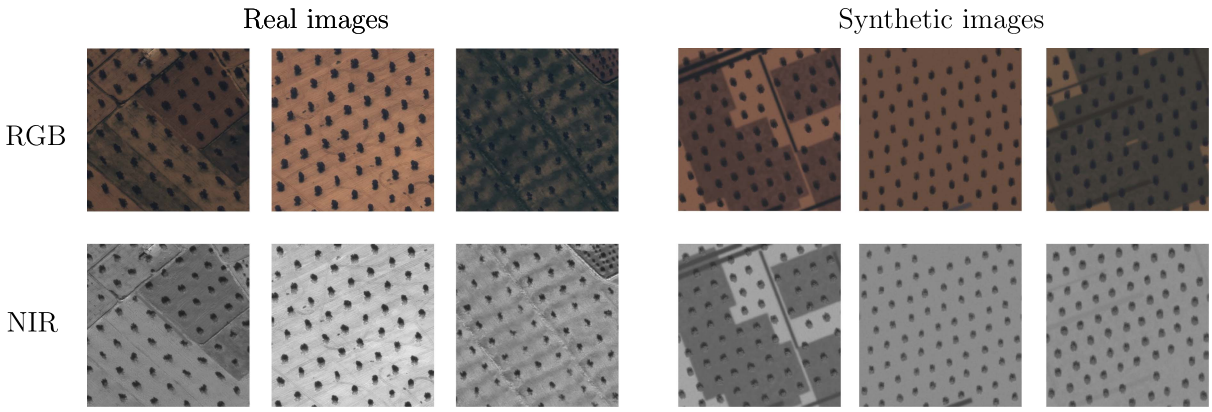

Fig. 4. Examples of real and synthetic satellite image patches across RGB (top row) and NIR (bottom row) channels.

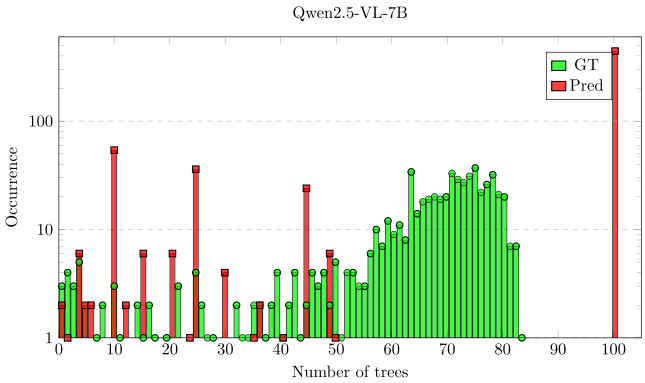

Fig. 5. Comparison between ground-truth tree counts (green) and predicted counts (red) by Qwen2.5VL7B across the test set. The model frequently outputs small, common tokens (1–10) and often defaults to "100," regardless of the actual count.

elliptical structuring element is applied to suppress isolated pixels and smooth region boundaries while preserving the overall shape of vegetation blobs.

The refined mask is then analyzed to segment the vegetation into distinct regions, each corresponding to a potential tree. This is achieved through connected-component analysis, which identifies contiguous vegetation regions and allows each candidate tree to be processed independently. For each connected component, an oriented bounding box is computed to approximate the spatial extent of the vegetation region, enabling robust localization even when trees are not axis-aligned.

Bounding boxes are generated around these regions to estimate the spatial arrangement and count of trees, providing a straightforward representation of individual trees in the scene. Additional filtering and splitting operations are applied to discard spurious detections and to handle elongated regions that may correspond to multiple closely spaced trees.

The pipeline outputs the estimated number of trees, derived from the count of bounding boxes. This approach offers a simple and practical method for tree counting, particularly in monoculture environments, such as olive plantations, where vegetation characteristics are relatively uniform.

Furthermore, to establish an AI-based counting baseline that does not use our generated images, we investigate the application of pretrained VLMs for tree counting tasks. In this approach, an RGB satellite image is provided along with a textual prompt requesting the estimated number of olive trees.

B. *Comparisons on the Synthetic Data*

To compare the different tree counting algorithms, we employ several metrics. The minimum and maximum of the absolute error (Min AE and Max AE, respectively) are calculated to capture the best and worst estimations across all samples. The median of the AE is reported as a robust measure of central tendency that is less affected by outliers, while the standard deviation (Std AE) quantifies the variability and consistency of the estimation errors. Furthermore, the root mean squared error (RMSE) is provided as a comprehensive indicator of the overall magnitude of the errors, giving higher weight to larger discrepancies. Finally, the mean absolute percentage error (MAPE) is computed to express the estimation errors as a percentage of the actual number of trees, providing a metric of accuracy.

Table I reports the performance of all approaches (i.e., the proposed ones, which are trained on the synthetic dataset, and the baselines). It is evident that the unsupervised vegetation-based method yields substantially poorer results, primarily because this technique struggles to cope with the high variability of scenarios intentionally included in the synthetic dataset to simulate complex and diverse conditions. The performance of Qwen2.5VL7B (prompting strategy in Appendix B) on the counting task, reported in Table I is poor as well, highlighting the limitations of VLMs on counting tasks. Fig. 5 further illustrates the behavior of the model by comparing the histograms of predicted versus ground-truth counts. While the model occasionally uses low-value tokens (e.g., 1–10), indicating some comprehension of the task, it disproportionately favors the frequent round number "100," revealing a bias toward commonly seen numeric tokens rather than precise quantitative inference.

This phenomenon can be attributed to the inherent limitations of autoregressive LLMs in regression-like roles. First, the standard categorical cross-entropy objective treats numeric tokens

TABLE I
OLIVE TREE COUNTING PERFORMANCE ON THE SYNTHETICALLY GENERATED SATELLITE IMAGES

Method	Using FOREST-GC	Min AE (↓)	Max AE (↓)	Median AE (↓)	Std AE (↓)	RMSE (↓)	MAPE (↓)
Unsupervised	×	0.00	193.0	9.5	47.92	60.77	146.56%
QwenVL 2.5	×	0.00	72.00	28.00	13.42	32.72	50.87%
YOLOv8-l	✓	0.00	7.00	3.00	1.52	3.49	4.95%
YOLOv11-l	✓	0.00	7.00	3.00	1.47	3.33	4.72%
YOLOv12-l	✓	0.00	8.00	3.00	1.50	3.51	5.01%
UNet-ResNet34	✓	0.00	4.00	0.00	0.61	0.69	0.53%
SAM-ViT-base	✓	0.00	6.00	0.00	0.94	1.08	1.08%
SAM-ViT-large	✓	0.00	6.00	0.00	0.73	0.81	0.61%

as independent classes, ignoring their ordinal relationships, a limitation directly addressed by works introducing regression-style losses like the number token loss [38]. Second, number tokenization and embedding spaces are discontinuous and misaligned: for instance, the token "110" may be embedded closer to "11" than to "111", disrupting arithmetic proximity [39]. These issues collectively hamper the model's fine-grained counting capabilities, forcing it to resort to frequent, round-valued tokens. Experiments on other VLMs support that this regression failure is not unique to Qwen2.5VL7B (see Appendix B). The poor performance of these models further highlights the value of developing a dedicated data generator, such as the one we propose, to train models capable of achieving reliable regression performance on this task. At the same time, training VLMs for regression tasks using language tokens remains an open problem, as these models are primarily designed for classification and captioning. While VLMs show promise for general visual understanding, their performance in precise counting tasks remains very limited. Because of this we selected task-specific models for the evaluation of FOREST-GC.

Table I also shows that the object detection models based on the YOLO family, fine-tuned on the synthetic training set, demonstrate excellent performance on the synthetic test set and show comparable results across generations from YOLOv8 to YOLOv11. However, the best performance is achieved by segmentation methods based on the SAM, which produce binary masks from which the number of connected components is computed to estimate the tree count. In this setting, UNet and SAM models are also fine-tuned on the synthetic dataset using the available pixel-level annotations, rather than being applied in a zero-shot manner. This fine-tuning enables the models to adapt to the visual characteristics of satellite imagery and plantation scenes. As expected, the model using the ViT-Large backbone outperforms the ViT-base variant, with both models achieving a median AE equal to zero, indicating that more than half of the predictions are correct to within a single tree. Details on how these models were trained are provided in Appendix C.

C. Testing on Real-World Dataset

To assess the performance of the proposed models on real-world data, we collect satellite images from the Airbus Pleiades Neo satellite. Specifically, we have data from a region of 25 km^2 from the Chaal region in Tunisia. Images come with four multispectral channels (R, G, B, and NIR) and one high spatial resolution monospectral channel, which is used to improve the spatial resolution of the multispectral channels down to 0.3 m^2 per pixel. Images were acquired from Airbus on June 9th, 2025.

From these regions, we select a subset of 50 satellite image patches of size 640 × 640 pixels, which are manually annotated by four coauthors of this study. Given the inherent difficulty of counting trees in complex scenes, cases where annotators did not reach full agreement were resolved by computing the average of their estimates, which was then used as the ground truth reference. This procedure ensures a more reliable evaluation despite the absence of definitive labels. To quantify annotation consistency, we compute the mean relative error (MRE) across annotators by measuring, for each annotator, the relative deviation of their count from the mean annotation for each image, and then averaging these values over all annotators and samples. The MRE among annotators was 4.86%, a level of variability that should be taken into account when interpreting the results. As with the synthetic dataset, we report the results in Table II, considering the same metrics.

Compared to the evaluation on synthetic data, the unsupervised vegetation-based approach shows a slight improvement in performance, which can be attributed to the lower variability present in the real images relative to the synthetic ones. The pretrained QwenVL-2.5 model again fails to provide acceptable results in this setting. In contrast, all the YOLO models trained on the synthetic dataset demonstrate that it is possible to achieve good performance and confirm the sim-to-real transferability, as the models are able to generalize effectively to real images. Representative examples of these detections are presented in Fig. 7.

The SAM-based models also achieve excellent performance on real data, as illustrated in Fig. 6. In particular, the SAM-ViT-Large model achieves an MAPE of 4.82%, which is close to the level of human performance, given that the relative error in the ground truth annotations is 4.86%. In contrast, UNet performance degrades significantly when tested on the real satellite images, highlighting limited generalization.

Provided that the benchmark relies on human-labeled satellite data rather than absolute ground truth, these results suggest that the synthetic dataset, combined with models with strong generation capabilities, provides a highly effective and customizable foundation for addressing tree counting tasks in real-world scenarios where annotated data are not available.

VI. DISCUSSION AND OPEN DIRECTIONS

Even though we demonstrated the efficacy of the synthetic satellite image generator for training and deploying neural-based

TABLE II
OLIVE TREE COUNTING PERFORMANCE ON THE REAL SATELLITE IMAGES

Method	Using FOREST-GC	Min AE (↓)	Max AE (↓)	Median AE (↓)	Std AE (↓)	RMSE (↓)	MAPE (↓)
Unsupervised	×	0.00	135.00	20.00	31.76	44.13	61.08%
QwenVL 2.5	×	36.75	83.25	42.25	9.64	46.41	86.47%
YOLOv8-l	✓	0.00	99.25	4.00	15.55	18.05	13.92%
YOLOv11-l	✓	0.25	63.25	4.75	10.43	13.20	13.37%
YOLOv12-l	✓	0.00	94.25	5.00	14.65	17.44	15.17%
UNet-ResNet34	✓	0.00	46.00	8.00	11.47	16.54	26.30%
SAM-ViT-base	✓	0.00	18.75	2.25	3.35	4.87	7.70%
SAM-ViT-large	✓	0.00	9.25	2.00	2.12	3,34	4.82%

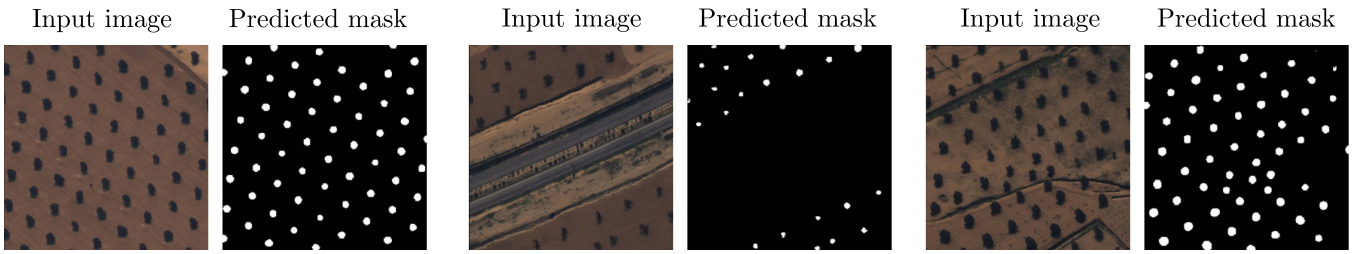

Fig. 6. Examples of segmentation masks predicted by SAM-ViT-Large on real RGB+NIR satellite imagery. For visualization, the input tiles are displayed as RGB composites extracted from the four-channel data. The predicted binary masks highlight the detected tree crowns. The total number of connected components in each mask was used to estimate the tree count per tile.

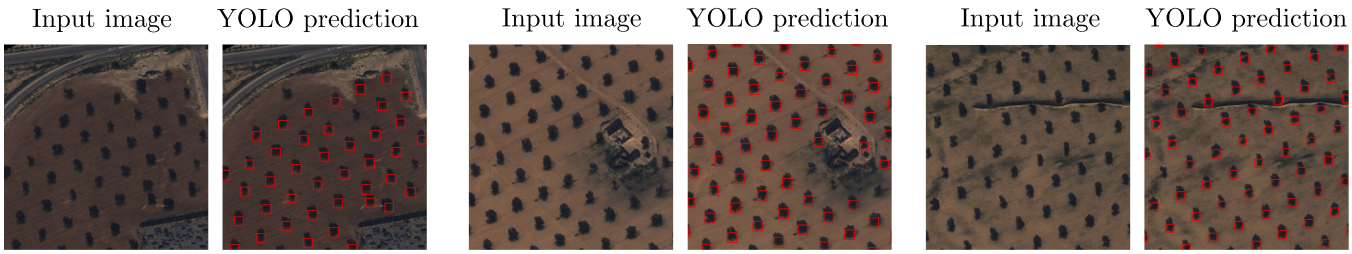

Fig. 7. Examples of detections produced by the YOLO-V8-l model on real RGB+NIR satellite imagery. The input tiles are displayed as RGB composites extracted from the four-channel data, and the red bounding boxes indicate the predicted locations of individual trees. The count of detections in each image was used as the estimated tree count.

methods in real scenarios, this approach comes with some shortcomings that need to be addressed. One limitation is that we did not design the image generator to account for all possible background scenarios encountered globally. In fact, satellite images may include, alongside trees, various obstacles, such as man-made structures or natural variations, such as bodies of water and rocky terrain. While in principle these elements could be incorporated into the image generator, the vast diversity of such features poses a significant challenge.

Another aspect concerns the photometric realism of the generated images. Although FOREST-GC allows fine-grained control over the color of individual elements, it does not explicitly enforce a perfect match to the full color distribution observed in real satellite imagery. Addressing this would likely require additional data-driven calibration or optimization strategies, which we consider a promising direction for future work.

In addition, tree variability itself may become a concern when tree canopies are heavily overlapped, their sizes vary greatly, or multiplneural model's ability to discern individual trees may reach a limit, requiring the intervention of additional tools, model to discern individual trees could reach a limit, requiring the intervention of additional tools, such as UAV inspection.

Finally, the lack of high-confidence ground truth labeling in real images introduces a critical issue: it is often difficult to visually distinguish trees that are overlapped, partially occluded, or too small to be easily recognized. As a result, the tuning process and performance evaluation of the algorithms might suffer because of an arbitrary definition of what counts as a "tree." For example, in this study, we define a tree as "detectable" if its size exceeds a certain threshold and it can be visually confirmed in the imagery. Fig. 8 illustrates this variability, where trees in the same tile can appear with markedly different sizes and densities. Generally speaking, the identification of out-of-distribution trees (such as small, sparse, or unhealthy specimens) may require closer inspection and tailored synthetic datasets that explicitly model these challenging cases to improve robustness. This means that we can currently only claim human-level annotation performance, while lacking ground-truth information to fully characterize performance in more complex scenarios. Moreover, future work could explore extending the generator to

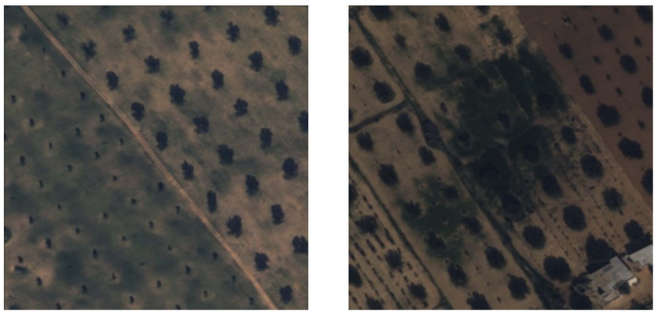

Fig. 8. Examples of a real satellite tile containing trees with significant variability in size, spacing, and visual appearance. This heterogeneity complicates consistent labeling and impacts algorithm training and evaluation.

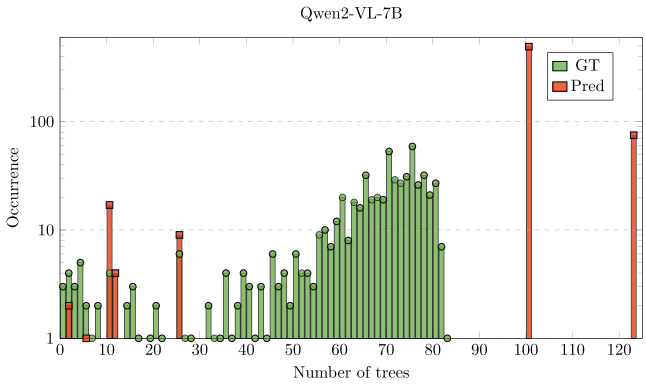

Fig. 9. Histogram comparison between ground-truth tree counts (green) and predictions (red) by Qwen2-VL-7B. The model frequently defaults to small numbers, the round number "100", or "123" (which simply represents the first three digits in sequence), reflecting extremely poor quantitative generalization.

simulate additional tree species, supporting broader applications to forestry monitoring and diverse agricultural contexts.

VII. Conclusion

In this work, we introduce FOREST-GC, a Python-based satellite image generator tailored for neural network-driven tree counting. Designed for speed and ease of integration into GPU-accelerated PyTorch environments, the framework enables the crafting of labeled, large-scale synthetic datasets. By mimicking the characteristics of real satellite imagery, these synthetic datasets provide a powerful foundation for training and evaluating modern neural models to detect and count trees in olive plantations. Our experiments demonstrate that models trained on this data can outperform a traditional unsupervised method, highlighting the potential of this approach, especially in scenarios where real labeled datasets are scarce or unavailable.

Appendix A
Tree Field Generation Parameters

The synthetic satellite image generator, FOREST-GC, presented in this work exposes over 100 configurable parameters to control the appearance, spatial distribution, and rendering properties of the generated tree fields. These parameters enable fine-grained control over aspects, such as background variability, tree placement, shape, shadows, and overall scene composition. We summarize the main categories of parameters and their default configurations as follows.

1) *Field Size and Layout:* Parameters including the output resolution (`field_size`), zoom factor, batch size, and total number of fields generated per batch.
2) *Tree Map Generation:* Controls for the spatial probability map, including Gaussian filter size and sigma (`treemap_filter_*`), noise strength, and pruning rates to emulate irregular planting patterns.
3) *Tree Positioning and Size:* Settings for the spacing between trees (`tree_xspace`, `tree_yspace`), jitter in placement, and thresholds determining whether a tree is planted.
4) *Tree Shape Modeling:* Parameters for sprite radius ranges (e.g., `treeshape_min_radius`), noisiness, and Gaussian smoothing filters applied to each tree silhouette.
5) *Background Synthesis:* Controls for generating complex terrain textures, including multiple passes of rectangular patches, background noise, and overlays to emulate soil variability and roads.
6) *Shadow Rendering:* Parameters defining shadow direction, length, blending strength, and the number of iterations used to create realistic shadow effects.
7) *Color Settings:* A library of predefined color sets specifying RGB values for each scene component (background, patches, trees, shadows) along with a randomization factor to introduce variability.
8) *Postprocessing:* Rotation angles, tiling configuration, and tile dimensions to compose the final image and mitigate border artifacts.

The full set of parameters and their default values are available in the source code repository for reference and customization, along with the specific configuration used to generate the synthetic dataset employed in this study.

Appendix B
Experiments With VLMs

In this Appendix, we report the prompting strategy and supplementary results evaluating additional pretrained VLMs on the tree counting task. Tests were performed using the following prompt: *"You are looking at a high-resolution satellite image of an olive tree plantation. Your task is to count all the individual olive trees visible in the image. Return only the number of olive trees as a single integer. Do not include any other text, symbols, or explanation"*. The result is the number generated by the VLM. Specifically, we tested Qwen2-VL-7B (Fig. 9) and Aya-Vision-8B (Fig. 10). The results consistently confirm the inability of these models to perform precise numeric regression from satellite imagery.

Overall, both models exhibit similar or worse performance than the example shown in the main text (Fig. 5). Despite occasionally producing low-value tokens consistent with small

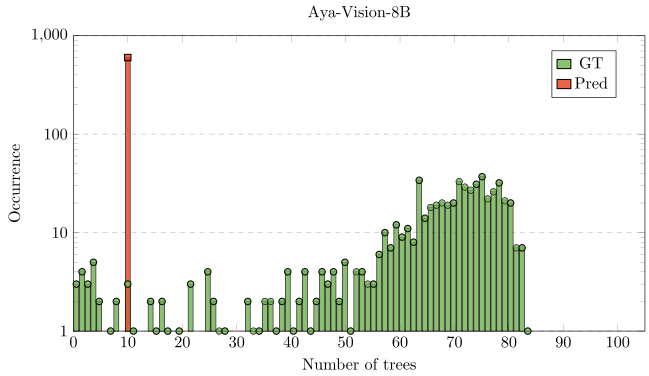

Fig. 10. Histogram comparison between ground-truth tree counts (green) and predictions (red) by Aya-Vision-8B. Predictions are totally biased toward the common numeric tokens "10".

tree counts, predictions overwhelmingly cluster around a few frequent integers, confirming the regression failure of current VLMs on this task. These results reinforce the motivation to develop specialized synthetic datasets as proposed in this work.

APPENDIX C
NEURAL METHODS SETTINGS

All the pretrained VLMs used in this study were evaluated with a temperature parameter set to zero to produce deterministic outputs.

The YOLO-based architectures presented in this work were initialized using the corresponding pretrained models provided by Ultralytics.[3] The architectures were modified to accept input images with dimensions $(H \times W \times 4)$, where $H = W = 640$. All YOLO-based networks were trained for 20 epochs with an initial learning rate of 0.01 and a batch size of 16.

The SAM-based models were initialized from the weights available on HuggingFace,[4] and adapted to accept four-channel inputs by adding a convolutional adapter at the beginning of the model. This adapter consists of three convolutional filters with four input channels each and a kernel size of 1×1. The models were trained for 10 epochs using a binary cross-entropy loss, a learning rate of 10^{-5}, and a batch size of 4. All models were trained on a single NVIDIA A100 GPU with 40 GB of memory.

ACKNOWLEDGMENT

During the realization of this work G. Setti was also a member of the Center of Excellence for Sustainable Food Security with with King Abdullah University of Science and Technology (KAUST), Saudi Arabia.

REFERENCES

[1] M. Brandt et al., "High-resolution sensors and deep learning models for tree resource monitoring," *Nature Rev. Elect. Eng.*, vol. 2, no. 1, pp. 13–26, Jan. 2025.

[2] B. T. W. Putra, P. Soni, B. Marhaenanto, S. P. S. Harsono, and S. Fountas, "Using information from images for plantation monitoring: A review of solutions for smallholders," *Inf. Process. Agriculture*, vol. 7, no. 1, pp. 109–119, 2020.

[3] X. Liang et al., "Close-range remote sensing of forests: The state of the art, challenges, and opportunities for systems and data acquisitions," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 3, pp. 32–71, Sep. 2022.

[4] H. Teng, Y. Wang, D. Chatziparaschis, and K. Karydis, "Adaptive lidar odometry and mapping for autonomous agricultural mobile robots in unmanned farms," *Comput. Electron. Agriculture*, vol. 232, 2025, Art. no. 110023.

[5] D. Chatziparaschis, H. Teng, Y. Wang, P. Peiris, E. Seudiero, and K. Karydis, "On-the-go tree detection and geometric traits estimation with ground mobile robots in fruit tree groves," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2024, pp. 15840–15846.

[6] H. Pathak, C. Igathinathane, Z. Zhang, D. Archer, and J. Hendrickson, "A review of unmanned aerial vehicle-based methods for plant stand count evaluation in row crops," *Comput. Electron. Agriculture*, vol. 198, 2022, Art. no. 107064.

[7] H. Lu, L. Liu, Y.-N. Li, X.-M. Zhao, X.-Q. Wang, and Z.-G. Cao, "TasselNetV3: Explainable plant counting with guided upsampling and background suppression," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4700515.

[8] H. A. Amirkolaee, M. Shi, and M. Mulligan, "TreeFormer: A semi-supervised transformer-based framework for tree counting from a single high-resolution image," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4406215.

[9] L. Yao, T. Liu, J. Qin, N. Lu, and C. Zhou, "Tree counting with high spatial-resolution satellite imagery based on deep neural networks," *Ecological Indicators*, vol. 125, 2021, Art. no. 107591.

[10] S. Zhang et al., "A mapping approach for eucalyptus plantations canopy and single tree using high-resolution satellite images in Liuzhou, China," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4410413.

[11] H. He, F. Zhou, Y. Xia, M. Chen, and T. Chen, "Parallel fusion neural network considering local and global semantic information for citrus tree canopy segmentation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 1535–1549, 2024.

[12] B. Victor, A. Nibali, and Z. He, "A systematic review of the use of deep learning in satellite imagery for agriculture," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 18, pp. 2297–2316, 2025.

[13] C. Persello et al., "Deep learning and Earth observation to support the sustainable development goals: Current approaches, open challenges, and future opportunities," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 2, pp. 172–200, Jun. 2022.

[14] P. Kovačovič, R. Pirnık, J. Kafková, M. Michálik, A. Kanáliková, and P. Kuchár, "Satellite-based forest stand detection using artificial intelligence," *IEEE Access*, vol. 13, pp. 10898–10917, 2025.

[15] A. Shahtahmassebi, N. Yang, K. Wang, N. Moore, and Z. Shen, "Review of shadow detection and de-shadowing methods in remote sensing," *Chin. Geographical Sci.*, vol. 23, no. 4, pp. 403–420, Aug. 2013.

[16] P. González, P. Montesinos, and L. Gómez, "Tree detection using probabilistic reticule scanning in quickbird images," *Int. J. Remote Sens.*, vol. 28, no. 3/4, pp. 793–807, 2007.

[17] I. N. Daliakopoulos, E. G. Grillakis, A. G. Koutroulis, and I. K. Tsanis, "Tree crown detection on multispectral VHR satellite imagery," *Photogrammetric Eng. Remote Sens.*, vol. 75, no. 10, pp. 1201–1211, 2009.

[18] G. Filippa et al., "NDVI derived from near-infrared-enabled digital cameras: Applicability across different plant functional types," *Agricultural Forest Meteorol.*, vol. 249, pp. 275–285, 2018.

[19] Y. Bazi, H. Al-Sharari, and F. Melgani, "An automatic method for counting olive trees in very high spatial remote sensing images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2009, pp. II125–II128.

[20] J. Moreno-Garcia, L. J. Linares, L. Rodriguez-Benitez, and C. Solana-Cipres, "Olive trees detection in very high resolution images," in *Proc. 13th Int. Conf. Inf. Process. Manage. Uncertainty Knowl.-Based Syst.*, 2010, pp. 21–29.

[21] J. Peters, F. Van Coillie, T. Westra, and R. Wulf, "Synergy of very high resolution optical and radar data for object-based olive grove mapping," *Int. J. Geographical Inf. Sci.*, vol. 25, pp. 971–989, 06 2011.

[22] Y. Chemin and P. Beck, "A method to count olive trees in heterogenous plantations from aerial photographs," 2017.

[23] L. Zhao, S. Zheng, W. Yang, H. Wei, and X. Huang, "An image thresholding approach based on Gaussian mixture model," *Pattern Anal. Appl.*, vol. 22, pp. 75–88, 2019.

[3] Pretrained models. [Online]. Available: https://docs.ultralytics.com

[4] Specifically, the models were initialized starting from the weights of the `facebook/sam-vit-base` and `facebook/sam-vit-large` models.

[24] A. Khan et al., "Remote sensing: An automated methodology for olive tree detection and counting in satellite images," *IEEE Access*, vol. 6, pp. 77816–77 828, 2018.
[25] M. Waleed, T.-W. Um, A. Khan, and Z. Ahmad, "An automated method for detection and enumeration of olive trees through remote sensing," *IEEE Access*, vol. 8, pp. 108592–108601, 2020.
[26] R. Mezzi et al., "Olive tree classification and inventory with medium resolution multi-spectral satellite imagery," in *Space Fostering African Societies: Developing the African Continent Through Space*. Berlin, Germany: Springer, 2020, pp. 13–30.
[27] A. Abozeid, R. Alanazi, A. Elhadad, A. I. Taloba, R. M. Abd El-Aziz, and A. M. Khalil, "A large-scale dataset and deep learning model for detecting and counting olive trees in satellite imagery," *Intell. Neurosci.*, vol. 2022, Jan. 2022, Art. no. 1549842.
[28] F. Kong, B. Huang, K. Bradbury, and J. M. Malof, "The Synthinel-1 dataset: A collection of high resolution synthetic overhead imagery for building segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2020, pp. 1803–1812.
[29] M. Alibani, N. Acito, and G. Corsini, "Multispectral satellite image generation using StyleGAN3," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 4379–4391, 2024.
[30] V. A. Le et al., "Mask conditional synthetic satellite imagery," in *Proc. Int. Conf. Learn. Representations Workshop Mach. Learn. Remote Sens.*, 2023, pp. 1–6.
[31] R. Dosselmann and X. D. Yang, "Existing and emerging image quality metrics," in *Proc. Can. Conf. Elect. Comput. Eng.*, 2005, pp. 1906–1913.
[32] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788.
[33] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241.
[34] A. Kirillov et al., "Segment anything," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 3992–4003.
[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[36] D. Zhang, F. Wang, L. Ning, Z. Zhao, J. Gao, and X. Li, "Integrating SAM with feature interaction for remote sensing change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 4513011.
[37] J. Gao, D. Zhang, F. Wang, L. Ning, Z. Zhao, and X. Li, "Combining SAM with limited data for change detection in remote sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, 2025, Art. no. 5614311.
[38] J. Zausinger et al., "Regress, don't guess: A regression-like loss on number tokens for language models," in *Proc. Int. Conf. Mach. Learn.*, 2025, pp. 73995–74017.
[39] A. K. Singh and D. Strouse, "Tokenization counts: The impact of tokenization on arithmetic in frontier LLMs," 2024, *arXiv:2402.14903*.

Luciano Prono (Member, IEEE) received the M.Sc. degree in electronics engineering and the Ph.D. degree "cum laude" in electrical, electronics, and communication engineering from Politecnico di Torino, Turin, Italy, in 2023 and 2019, respectively.

He was a Visiting Ph.D. Student with the Institute of Neuroinformatics, University of Zurich and ETH, Zürich, Switzerland, in 2022, and a Visiting Researcher with the King Abdullah University of Science and Technology, Thuwal, Saudi Arabia, in 2025. He is currently an Assistant Professor with the Department of Electronics and Telecommunications, Politecnico di Torino. His research interests include in low-power systems and applications of artificial intelligence, mainly in the fields of Internet of Things, tiny Machine Learning (tinyML), edge computing, compressed sensing and neuromorphic computing.

Najmeddine Dhieb (Member, IEEE) received the Diplôme d'Ingénieur (with honors) in telecommunication engineering from the École Supérieure des Communications de Tunis (SUP'COM), Aryanah, Tunisia, in 2019.

He was a Research Assistant with the Smart City Lab within the School of Systems and Enterprises, Stevens Institute of Technology, Hoboken, NJ, USA, in 2019. He currently contributes his expertise as an AI Research Engineer with King Abdullah University of Science and Technology, Thuwal, Saudi Arabia, after serving as a Senior Data Scientist with Bdeo Technologies S.L., Madrid, Spain. His research interests include the intersection of artificial intelligence, computer vision, Internet of Things, edge computing, and blockchain.

Philippe Bich (Member, IEEE) received the B.Sc. degree in computer engineering, the M.Sc. degree (with honors) in mechatronic engineering, and the Ph.D. degree in electronics and telecommunications from the Department of Electronics and Telecommunications, all from Politecnico di Torino, Torino, Italy, in 2018, 2021, and 2026, respectively.

In 2021, he was a Visiting Student with the Boston University Robotics Lab, Boston University, Brookline, MA, USA, under the supervision of Prof. John Baillieul. His research interests include robotics and artificial intelligence, specifically in the areas of quantization and pruning of large deep neural networks to optimize both their training and inference processes.

Chiara Boretti (Member, IEEE) received the M.Sc. degree (with honors) in mechatronic engineering from Politecnico di Torino, Torino, Italy, in 2021, where she is currently working toward the Ph.D. degree in electrical, electronics, and communications engineering.

She is affiliated with the Interdepartmental Center for Service Robotics (PIC4SeR), Politecnico di Torino. Her research interests include robotic visual perception, utilizing both conventional frame-based cameras and cutting-edge neuromorphic sensors, such as event-based cameras. Her interests extend also to neuromorphic computing and to the development of algorithms for resource-constrained systems.

Fabio Pareschi (Senior Member, IEEE) received the Dr. Eng. degree (with honors) in electronic engineering from the University of Ferrara, Ferrara, Italy, in 2001, and the Ph.D. degree in information technology from the University of Bologna, Bologna, Italy, in 2007, under the European Doctorate Project.

He is currently an Associate Professor with the Department of Electronic and Telecommunication, Politecnico di Torino, Turin, Italy. He is also a Faculty Member with ARCES, University of Bologna. His research interests include analog and mixed-mode electronic circuit design, statistical signal processing, compressed sensing, DC–DC converters, random number generation and testing, and electromagnetic compatibility.

Dr. Pareschi was the recipient of the 2019 IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS Best Paper Award, the Best Paper Award at European Conference on Circuit Theory and Design 2005, and the Best Student Paper Award at EMC Zurich 2005 and IEEE International Workshop on the Electromagnetic Compatibility of Integrated Circuits 2019. From 2010 to 2013, he was an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: EXPRESS BRIEFS and for the IEEE OPEN JOURNAL OF CIRCUITS AND SYSTEMS from 2020 to 2022.

Marco Brini was born in Torino, Italy, on October 14, 1972. He received the M.Sc. degree in mathematics with a focus on cryptography and the MBA degree in international business from SAA, both from the University of Torino, Torino, Italy, in 2000 and 2002, respectively.

From 1992 to 2008, he held roles in network engineering, cybersecurity, and digital strategy, including with FIAT and as CEO of Minteos and EnvEve, where he pioneered Internet-of-Things platforms for environmental monitoring. From 2008 to 2022, he led digital agriculture projects across Africa, Asia, and Latin America, collaborating with Nestlé, Syngenta, and USAID. From 2022 to 2023, he was with ETH Zurich, Zürich, Switzerland, and the International Fund for Agricultural Development (IFAD). In 2024, he served as Senior Director of Agricultural Innovation and R&D, NADEC, Riyadh, Saudi Arabia. He is currently with the SAI Platform, Geneva, Switzerland, leading work on sustainable agriculture and digital innovation. He advises International Telecommunication Union, Food and Agriculture Organization of the United Nations (FAO), and the World Agriculture Forum. He coauthored books with Springer and FAO–IFAD–the Islamic Development Bank.

Hakim Ghazzai (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia, in 2015, and the diplome d'Ingenieur (honors) in telecommunication engineering and the master's degree in high-rate transmission systems from the Ecole Superieure des Communications de Tunis, Tunis, Tunisia, in 2010 and 2011, respectively.

He was a Researcher Scholar with the Qatar Mobility Innovations Center, Doha, Qatar; Karlstad University, Karlstad, Sweden; and Stevens Institute of Technology, Hoboken, NJ, USA. He is currently a Senior Research Scientist with KAUST. He has authored and coauthored more than 200 publications. His research interests include artificial intelligence-enabled applications, the Internet of Things, intelligent transportation systems, and mobile and wireless networks.

Dr. Ghazzai was the recipient of appreciation for being an exemplary reviewer for IEEE Wireless Communications Letters in 2016 and IEEE Communications Letters in 2017. He has been on the Editorial Board of the IEEE COMMUNICATIONS LETTERS and the IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY. He also joined the Board of IoT and Sensor Networks (specialty section of *Frontiers in Communications and Networks*) as an Associate Editor.

Riccardo Rovatti (Fellow, IEEE) received the M.S. degree in electronic engineering and the Ph.D. degree in electronics, computer science, and telecommunications from the University of Bologna, Bologna, Italy, in 1992 and 1996, respectively.

He is currently a Full Professor of Electronics with the University of Bologna. He has authored more than 300 technical contributions to international conferences and journals and two volumes. His research interests include mathematical and applicative aspects of statistical signal processing, on machine learning for signal processing, and on the application of statistics to nonlinear dynamical systems.

Dr. Rovatti was the recipient of the 2004 IEEE Circuits and Systems (CAS) Society Darlington Award, the 2013 IEEE CAS Society Guillemin-Cauer Award, and the 2019 IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS Best Paper Award. He was also the recipient of the Best Paper Award at European Conference on Circuit Theory and Design 2005 and the Best Student Paper Award at the EMC Zurich 2005 and IEEE International Symposium on Circuits and Systems 2011. He was Distinguished Lecturer of the IEEE CAS Society for the years 2017 to 2018. His contributions include nonlinear and statistical signal processing applied to electronic systems.

Gianluca Setti (Fellow, IEEE) received the Dr. Eng. (honors) and Ph.D. degrees in electronic engineering from the University of Bologna, Bologna, Italy, in 1992 and 1997, respectively.

From 1997 to 2017, he was with the Department of Engineering, University of Ferrara, Ferrara, Italy, as an Assistant Professor, from 1998 to 2000, Associate Professor, from 2001 to 2008, and as a Professor, from 2009 to 2017, of Circuit Theory and Analog Electronics. Since 2017, he has been a Professor of Electronics, Signal, and Data Processing with the Department of Electronics and Telecommunications, Politecnico di Torino, Torino, Italy. From 2022 to 2026, he was on leave to serve as Dean of the Computer, Electrical, Mathematical Sciences, and Engineering Division and Professor od the Electrical and Computer Engineering Program, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia. He also held several positions as a Visiting Professor/Scientist with Swiss Federal Technology Institute of Lausanne, Lausanne, Switzerland, in 2002 and 2005; University of California San Diego, La Jolla, CA, USA, in 2004; IBM, in 2004 and 2007; University of Washington, Seattle, WA, USA, in 2008 and 2010; and University of Maryland, College Park, MD, USA, from 2020 to 2022. He has coedited the books *Chaotic Electronics in Telecommunications* (CRC Press, 2000), *Circuits and Systems for Future Generation of Wireless Communications* (Springer, 2009), and *Design and Analysis of Biomolecular Circuits* (Springer, 2011), has coauthored the book *Adapted Compressed Sensing for Effective Hardware Implementations* (2018) as well as guest edited the May 2002 special issue of the IEEE Proceedings on "Applications of Non-linear Dynamics to Electronic and Information Engineering." His research interests include nonlinear circuits, recurrent neural networks, electromagnetic compatibility, compressive sensing and statistical signal processing, biomedical circuits and systems, power electronics, design and implementation of IoT nodes, circuits and systems for machine learning, and applications of AI techniques for anomaly detection and predictive maintenance.

Dr. Setti was the recipient of the 2025 IEEE Circuits and Systems (CAS) Society Mac Van Valkenburg Award, the 2013 IEEE CAS Society Meritorious Service Award, the 2013 IEEE CAS Society Guillemin-Cauer Award, the 2004 IEEE CAS Society Darlington Award, as well as the 2019 IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS Best Paper Award. He was also the recipient of several other conference best (student) paper awards at European Conference on Circuit Theory and Design 2005, EMC Zurich 2005, IEEE International Symposium on Circuits and Systems 2011, IEEE PRIME 2019, IEEE International Workshop on the Electromagnetic Compatibility of Integrated Circuits 2019 and at I2MTS2024, and the recipient of the 1998 Caianiello prize for the best Italian Ph.D. thesis on Neural Networks. He served as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS - PART I from 1999 to 2002 and 2002 to 2004 and for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS - PART II from 2004 to 2007, as the Deputy-Editor-in-Chief, for the *IEEE Circuits and Systems Magazine* from 2004 to 2007, as well as the Editor-in-Chief for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS - PART II from 2006 to 2007 and of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS - PART I from 2008 to 2009. He also served on the editorial Board of IEEE ACCESS from 2013 to 2015, and since 2025, and of the PROCEEDINGS OF THE IEEE from 2015 to 2018. From 2019 to 2024, he served for two terms as the first non US Editor-in-Chief of the PROCEEDINGS OF THE IEEE, the flagship journal of the Institute. He was a Distinguished Lecturer of the IEEE CAS Society from 2004 to 2005 and 2013 to 2014, a member of the CAS Society Board of Governors from 2005 to 2008, and served as the 2010 CAS Society President. In 2012, he was the Chair of the IEEE Strategic Planning Committee of the Publication Services and Products Board, and, in 2013 to 2014, he was the first non North-American Vice President of the IEEE for Publication Services and Products. He has served in the program committee of many conferences and was, in particular, the Special Sessions Co-Chair of the International Symposium on Circuits and Systems (ISCAS)2005 (Kobe) and ISCAS2006 (Kos); the Technical Program Co-Chair of IEEE Workshop in Nonlinear Dynamics of Electronic Systems 2000 (Catania), ISCAS2007 (New Orleans), ISCAS2008 (Seattle), IEEE International Conference on Electronics, Circuits and Systems 2012 (Seville), IEEE Biomedical Circuits and Systems Conference 2013 (Rotterdam), IEEE International Midwest Symposium On Circuits and Systems 2023 (Phoenix), as well as the General Co-Chair of International Symposium on Nonlinear Theory and Its Applications 2006 (Bologna), ISCAS2018 (Florence), ISICA2026 (Thuwal), and ISCAS2026 (Shanghai).