Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Vine2Graph - An algorithm for defining 2D grapevine structures

Francisco Oliveira

francisco.a.oliveira@inesctec.pt

INESC TEC - Institute for Systems and Computer Engineering, Technology and Science

Daniel Queirós da Silva

INESC TEC - Institute for Systems and Computer Engineering, Technology and Science

Mário Cunha

Faculty of Sciences of the University of Porto

José Boaventura Cunha

University of Trás-os-Montes and Alto Douro

Filipe N. Santos

Faculty of Engineering of the University of Porto

Research Article

Keywords:

Posted Date: November 23rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11689641/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Vine2Graph - An algorithm for defining 2D grapevine structures

Francisco Oliveira[1,2*], Daniel Queirós da Silva[1], Mário Cunha[1,3], José Boaventura Cunha[1,2], Filipe N. Santos[1,4]

[1]INESC TEC - Institute for Systems and Computer Engineering, Technology and Science, Campus da FEUP, S/N, 4200-465, Porto, Portugal.
[2]Department of Engineering, UTAD - University of Trás-os-Montes and Alto Douro, 5000-801, Vila Real, Portugal.
[3]Department of Geosciences, Environment and Spatial Planning, Faculty of Sciences of the University of Porto, Rua do Campo Alegre S/N, 4169-007, Porto, Portugal.
[4]Department of Electrical and Computer Engineering, Faculty of Engineering of the University of Porto, Campus da FEUP, S/N, 4200-465, Porto, Portugal.

*Corresponding author(s). E-mail(s): francisco.a.oliveira@inesctec.pt;
Contributing authors: daniel.q.silva@inesctec.pt; mccunha@fc.up.pt; jboavent@utad.pt; filipe.n.santos@inesctec.pt;

Abstract

A major challenge in precision viticulture consists in translating structural variability between individual grapevines into information that can guide management operations. Grapevine pruning is a complex task that requires an explicit representation of plant structure to support the application of pruning rules. This work proposes Vine2Graph, a graph-based framework that converts grapevine node detections and segmented structural components obtained from 2D images into a structured representation of the visible grapevine. The resulting rooted tree graph preserves the spatial and semantic relationships between grapevine components, including the trunk, canes, cordons and spurs, and provides an intermediate representation for downstream robotic pruning modules. Cane geometry is further represented through parametric curves associated with graph edges, enabling a smooth approximation of cane trajectories. Vine2Graph is independent of the upstream perception method, provided that node detections and

segmentation masks are available in the required format. A structural reconstruction evaluation performed on 15 grapevines resulted in an overall node assignment accuracy of 97.78 %, a segment connection accuracy of 99.13 % and a full cane completeness of 83.80 %. The parametric cane representation achieved an overall mean mask containment rate of 99.50 %. Across the evaluated grapevine examples, mean processing times ranged from 42 to 175 ms, corresponding to graph-generation rates of approximately 6 to 24 Hz.

Keywords: Semantic Representation, Pruning, Automation, Robotics, Precision Agriculture

1 Introduction

There is a growing shortage of labour in vineyards, particularly during grapevine pruning, which is one of the most labour-intensive tasks in viticulture (Mccorkle et al. 2019; Navone et al. 2025). This lack of available workers poses significant challenges to timely and effective vineyard management (Silwal et al. 2022; Klocker et al. 2020).

Grapevine pruning is performed differently around the world, presenting methods that differ not only in grape varieties but also on different terrains, environment, access to the vineyard, and mechanization employed (Strub et al. 2021). This task is performed to manage the growth and yield of the grapevine, as well as to guarantee the health of the plant (Silwal et al. 2022). Several studies have focused on automating pruning, making use of robotic solutions to carry out this branch trimming operation (Tinoco et al. 2021; Oliveira et al. 2025). However, automating this task is a complex procedure, as it depends on several conditions.

Agricultural environments are highly heterogeneous and tend to quickly change due to climatic action, or even by human influence (Gebbers and Adamchuk 2010). When deploying technology, such as robots, to automate agricultural tasks, the variability of the environment inherently increases the probability of failure of the task (Bechar and Vigneault 2016). Furthermore, for grapevine pruning to be conducted with the same criteria as performed manually, several pruning rules need to be implemented depending on the training system of the targeted grapevines. Moreover, to apply pruning rules autonomously, it is required to be able to define the grapevine structure. In the literature, some works have already focused on this matter. Fernandes et al. (2021) presented a method to be implemented in spur pruning. This method consists of detecting the spurs and afterwards performing segmentation on that portion of the grapevine, identifying the elements visible in the frame, such as the spur, canes and nodes. Based on the segmented components, the pruning rules are then applied. Yang et al. (2022) developed a pruning location algorithm based on a thinning algorithm. Firstly, the grapevine structure is segmented, a thinning algorithm is then applied, and in the resulting output the canes are reconnected based on their position. Afterwards, the nodes are extracted and the pruning location is chosen based on the number of buds to be retained. Chen et al. (2024) proposed a similar method which also relies on

segmentation followed by an application of a thinning algorithm to obtain the skeleton of the captured components.

These previous approaches to automated or assisted grapevine pruning have demonstrated the ability to identify structural elements of the grapevine for applying pruning rules. However, these methods typically focus on specific portions of the plant, resulting in a partial representation of the grapevine structure. This may limit the amount of contextual structural information available for the implementation of pruning decision modules. Therefore, the development of grapevine-level representations capable of preserving the spatial and semantic relationships between different plant structures remains relevant for robotic pruning applications. This work hypothesizes that a structured representation of the complete visible grapevine, combined with global spatial analysis of its components, can provide a more comprehensive and consistent semantic representation of the grapevine, which may benefit pruning strategies. Developing such an approach could significantly enhance the viability of automated or semi-automated pruning systems.

Building on this gap, this work proposes Vine2Graph, a framework for converting 2D grapevine perception outputs into a structured semantic graph. Rather than performing perception or pruning decisions directly, Vine2Graph provides an intermediate representation that organises grapevine nodes and structural components while preserving their spatial and semantic relationships. This representation is intended to support downstream robotic modules that require explicit information on grapevine topology and structure for pruning-related decision-making. By providing plant-level structural information, this representation also supports precision viticulture approaches in which management operations can be adapted according to the variability between individual grapevines.

2 Methodology

The Vine2Graph framework comprises three sequential stages, presented in Figure 1: graph generation, in which detected nodes are associated with segmented cane structures and connected into initial graph components; grapevine model optimisation, in which structurally inconsistent components are filtered and semantic information from permanent woody structures is incorporated; and parametric edge construction, in which cane geometry is represented by smooth parametric curves between consecutive graph nodes.

2.1 Graph generation

First, the Vine2Graph algorithm receives node detections and segmentation masks of the grapevine structural components, together with relevant acquisition parameters such as the image dimensions. Therefore, the graph generation stage operates on the resulting detection and segmentation data and is independent of the specific perception models used to obtain them.

The grapevine node detections and structural masks are then passed to the module responsible for constructing the initial graph using NetworkX data structures (Hagberg et al. 2008). The final grapevine representation is organised as a tree rooted at a

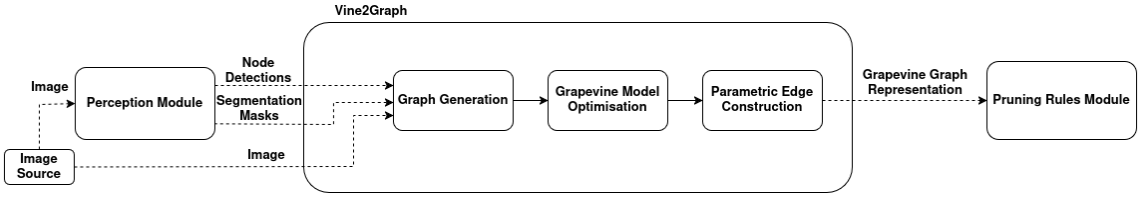


Fig. 1 Vine2Graph Framework

dedicated reference node (Ref), while initially the individual cane components are constructed individually. The algorithm builds the graph according to the following rules:

- Each grapevine node corresponds to a graph node;
- Grapevine nodes are associated with individual canes when their centre coordinates are inside the corresponding cane segmentation mask;
- Nodes within the same cane are ordered and consecutively connected; and
- A dedicated Ref node is created to capture information about the grapevine trunk, serving as the starting point for the grapevine structure by connecting the individual cane components.

When a trunk segmentation is available, the position for the Ref node is obtained from the lower central region of the trunk mask. If the trunk is not segmented, the lower centre of the image is used as a fallback reference position. The current acquisition configuration is intended to observe a single grapevine trunk within the image.

For each cane instance, the algorithm first identifies all grapevine nodes whose centre coordinates are contained in its segmentation mask. A node can only be associated with one cane, and the corresponding cane identifier and segmented area are stored as attributes of the associated graph nodes. Cane instances containing fewer than two nodes are not considered to build a cane component.

To establish the order of the nodes along each cane, the node with the smallest Euclidean distance to Ref node is selected as the starting node. Afterwards, the remaining nodes are iteratively ordered by selecting the node with the smallest Euclidean distance to the previously selected node. Edges are subsequently created only between consecutive nodes in this sequence. This local ordering allows the graph construction to better accommodate canes that develop horizontally or present curved trajectories from their base.

Detected grapevine nodes that cannot be associated with a valid cane component remain disconnected at this stage and are treated during the following graph optimisation process.

2.2 Optimised grapevine model

After constructing the initial cane components, the graph requires tuning to improve the structural consistency of the semantic representation of the grapevine, by removing components that are not structurally consistent with the target grapevine. Grapevine nodes that remain disconnected from any valid cane components are not transferred to the final graph. Similarly, cane instances containing fewer than two nodes have already been discarded during the initial graph generation.

To reduce the inclusion of cane components not belonging to the target grapevine, the framework evaluates the proximity of each cane base to the permanent woody structures of the plant. For this purpose, contour information is considered from the segmented trunk, cordons and spurs. In each connected cane component, the leaf node closest to the Ref node is considered as the base of the cane. The minimum Euclidean distance between this node and the available permanent woody structures contour points is then calculated. If this distance exceeds a predefined proximity threshold, the cane component is considered to not belong to the target grapevine and consequently discarded from the graph.

The retained cane components are then incorporated into the final graph, with the base node of each cane connected directly to the Ref node. As a result, the cane components are combined into a single tree structured representation with root at the Ref node.

In addition to cane topology, semantic information describing the permanent grapevine structures is also stored in the graph. The Vine2Graph framework for every segmented trunk, cordon and spur retains information regarding their centroid, area, bounding box and simplified contour. These data are then associated with the Ref node, providing structural information of the grapevine for further tasks. Furthermore, when spur instances are available in the structure, each cane base is additionally associated with the closest segmented spur available, allowing to establish a semantic relationship between the cane and the permanent structure from which it originates.

2.3 Parametric edges

Grapevine canes are not necessarily linear, and their shape may vary along their length. Therefore, representing the connections between consecutive grapevine nodes only through straight graph edges may result in an oversimplified representation of the cane geometry. To preserve a smoother and more realistic representation of the cane structure, Vine2Graph associates a parametric curve with the graph edges connecting the nodes of each cane.

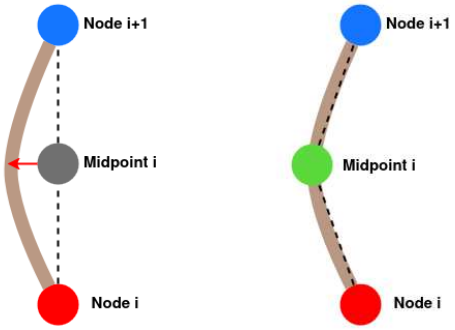
This process is applied independently to each connected cane component. First, for every pair of consecutive grapevine nodes, the geometric middle point (Midpoint) of the line connecting them is calculated. If this Midpoint is contained within the cane segmentation mask associated with both nodes, its coordinates are retained. Otherwise, the algorithm searches for a corrected Midpoint along the direction perpendicular to the original node-to-node segment.

To determine this corrected position, the contour points of the corresponding cane mask are projected onto the perpendicular line passing through the original Midpoint.

Projected points located within the cane mask are considered as candidate positions, and the candidate with the smallest Euclidean distance to the original Midpoint is selected. The Midpoint calculation methods are illustrated in Figure 2.



(a) Midpoint remains unchanged since it is already contained in the cane mask



(b) Midpoint is updated since it is not contained in the cane mask

Fig. 2 Examples of Midpoint calculation and update based on their placement on the respective grapevine cane

After determining the Midpoint coordinates, additional graph nodes are created at these positions, resulting in an ordered sequence containing the original grapevine nodes and the intermediate Midpoints. Performing this operation already allows for the connection between nodes to be more similar to the real world configuration, yet they are still being approximated by straight lines which differ from the real world scenario. A Piecewise Cubic Hermite Interpolating Polynomial (PCHIP) (Virtanen et al. 2020) is then applied independently to the x and y coordinates of the ordered sequence containing both the original grapevine nodes and the intermediate Midpoints.

This interpolator provides a smooth piecewise cubic representation while reducing unnecessary oscillations between closely spaced points.

The 2D position of the cane is represented parametrically by $x(t)$ and $y(t)$, where each interval is described by a cubic polynomial, as shown in Equation 1.

$$\begin{aligned} x_i(t) &= a_{x,i}(t-t_i)^3 + b_{x,i}(t-t_i)^2 + c_{x,i}(t-t_i) + d_{x,i}, \\ y_i(t) &= a_{y,i}(t-t_i)^3 + b_{y,i}(t-t_i)^2 + c_{y,i}(t-t_i) + d_{y,i}, \\ &\text{for} \quad t \in [t_i, t_{i+1}] \end{aligned} \tag{1}$$

The coefficients of the generated curves (a, b, c, d) are then stored as attributes on the edges connecting the grapevine nodes of that respective cane on the graph. This parametric representation provides a smooth approximation of the cane geometry between consecutive points and allows the curve to be easily reconstructed for further analysis and visualisation, all while maintaining low computational cost (Pérez et al. 2017). Finally, the complete graph is published as a Robot Operating System 2 (ROS2) (Macenski et al. 2022) topic, allowing downstream modules to access and use the generated grapevine representation.

3 Data inputs

As previously stated, the Vine2Graph graph generation module receives grapevine node detections and segmentation masks describing the structural components of the plant. In an automated system, these inputs are provided by an upstream perception module operating independently from the Vine2Graph framework. This separation allows the graph-generation stage to remain modular, since different perception methods can be used as long as the required detection and segmentation information is provided in the expected format.

Regarding the grapevine nodes, Vine2Graph requires their 2D position within the image provided as detection coordinates in bounding box format. The centre coordinates of each bounding box are considered as the position of the corresponding grapevine node.

Regarding the structural segmentation masks, four grapevine classes are considered: trunk, cane, spur and cordon, as illustrated in Figure 3. For these structures, Vine2Graph expects the contour coordinates of each segmented instance.

The segmented cane instances are used to associate and organise grapevine nodes into individual cane components, while the trunk, cordon and spur masks provide additional structural information for the graph generation and optimisation processes. Together, these inputs provide the semantic and geometric information required by Vine2Graph to construct the 2D grapevine representation.



Fig. 3 Segmentation classes for grapevine structure segmentation: trunk (green), cordon (blue), spur (red) and cane (pink)

4 Results

4.1 Vine2Graph framework

The Vine2Graph algorithm was tested on three distinct grapevine datasets. Two of these consist of grapevine images from the Portuguese *Dão* and *Douro* regions from previous works (Oliveira et al. 2024). Additionally, the algorithm was evaluated using the 3D2Cut dataset from the work developed by Gentilhomme et al. (2023). The original image resolutions were 4608×3456 px for the *Dão* and *Douro* datasets and 4032×3025 px for the 3D2Cut dataset, but all images were resized to 1280×960 px to standardise the input dimensions for computational performance comparison.

The *Dão* region dataset features grapevines trained using a bilateral cordon system, with medium-length canes and moderate wood density. The distance between nodes is short and there is a higher cane density per plant. In contrast, the grapevines in the *Douro* region dataset present a double Guyot training system, with shorter canes and larger distances between nodes. Meanwhile, the 3D2Cut dataset presents grapevines with low shoot density, increased distances between nodes and a shorter distance from the ground.

Examples of images from these datasets are shown in Figure 4. The grapevines exhibit different structural configurations and do not necessarily contain the same set of segmented structures. These differences allow the behaviour of the developed algorithm to be analysed across grapevines with distinct structural characteristics.

In Figure 5 is possible to observe the full framework applied to a grapevine from the *Dão* region. The annotated nodes and segmentation masks serve as inputs to the Vine2Graph algorithm, from which the graph representation is subsequently generated.

The graph itself contains a reference node, whose position is calculated from the lower central region of the trunk segmentation mask and which serves as the root of the grapevine representation. The grapevine present in this example, from the *Dão* region, apart from the trunk, cane and node structures, also contains cordons and



(a) *Dão*



(b) *Douro*



(c) 3D2Cut (Gentilhomme et al. 2023)

Fig. 4 Grapevines from the three datasets used

spurs. Depending on the task applied in this vineyard, these additional structures can influence the decisions to be conducted. Therefore, the following are calculated for each structure: centroid (used as the reference point for the structure coordinates), area, bounding box and simplified contour. These structures and their respective data are stored on the reference node, allowing easy access by the downstream modules. Furthermore, since spur structures are available, each cane is associated with the closest segmented spur. Therefore, in addition to the identification of the cane to which they belong, the corresponding grapevine nodes also store the identification of the associated spur.

In Figure 6, the grapevine from the *Douro* region presents a different structure, with no spur structures represented. The algorithm generates a graph representation containing information regarding the trunk, cordon, cane and node classes. Unlike the previous example, where canes could be associated with segmented spurs, the nodes in this example are only associated with their respective cane. Although some canes may visually appear to originate from a cordon, this relationship is not explicitly assigned by Vine2Graph, since it depends on the grapevine training system and on the structural configuration of the cordon.



(a) Grapevine node annotations

(b) Grapevine structure annotations

(c) Generated graph

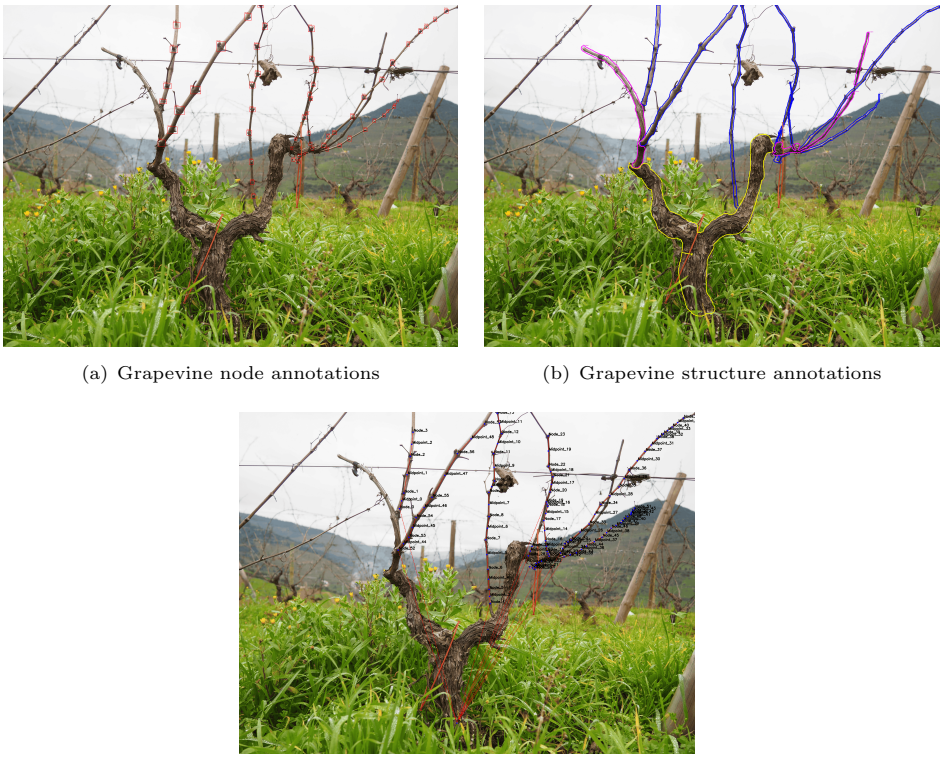
Fig. 5 Stages of the Vine2Graph framework in a grapevine from the *Dão* region

Figure 7 contains the implementation of the framework on a grapevine from the 3D2Cut dataset. Regarding the structures present in the grapevine, this example is similar to the grapevine from the *Douro* dataset, containing nodes, canes, cordon and trunk structures. The main difference consists of the presence of a single horizontal cordon, compared with the two vertical cordon structures present in the *Douro* example. Despite this difference, the Vine2Graph framework does not impose a specific cordon orientation or configuration. Consequently, both grapevine structures are represented without further modification of the framework.

The information stored on the different types of graph structures is illustrated in the diagram of Figure 8.

4.2 Structural and geometric reconstruction evaluation

To quantitatively evaluate the structural reconstruction performed by Vine2Graph, five grapevines from each of the *Dão*, *Douro* and 3D2Cut datasets were analysed, resulting in a total of 15 grapevines. These grapevines were chosen to capture structural variability across the evaluation, since each dataset presented grapevines with distinct structural characteristics. For each example, the expected grapevine nodes and



(a) Grapevine node annotations

(b) Grapevine structure annotations

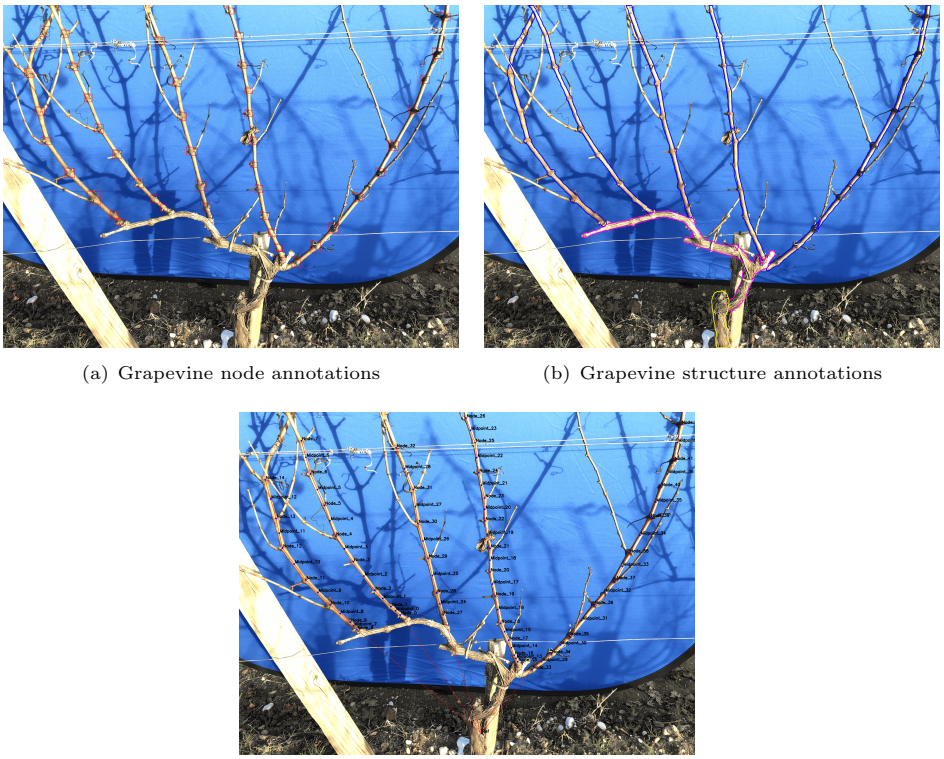
(c) Generated graph

Fig. 6 Stages of the Vine2Graph framework in a grapevine from the *Douro* region

cane structures were compared with the graph generated by Vine2Graph. A node was considered correct when it was assigned to the cane to which it biologically belonged. A generated segment connection was considered correct when it connected two nodes that were consecutive in the expected order of the corresponding cane. Segment connection accuracy was calculated as the percentage of generated connections that were correct. A cane was considered fully complete only when it contained exactly the expected nodes connected in the correct order. The performance of the framework for these examples is presented in Table 1.

Across the 15 analysed grapevines, Vine2Graph correctly assigned 926 of the 947 expected nodes to their respective canes, corresponding to an overall node assignment accuracy of 97.78 %. For the segment connections, 794 of the 801 connections generated by Vine2Graph were correct, corresponding to an overall segment connection accuracy of 99.13 %. Regarding the canes, Vine2Graph generated 129 canes from the 142 expected across the analysed grapevines. Of these expected canes, 119 were reconstructed with the correct set of nodes connected in the correct order, corresponding to an overall full cane completeness of 83.80 %.

The geometric consistency of the parametric cane representation was additionally evaluated by uniformly sampling 100 points along the PCHIP curve generated for



(a) Grapevine node annotations

(b) Grapevine structure annotations

(c) Generated graph

Fig. 7 Stages of the Vine2Graph framework in a grapevine from the 3D2Cut dataset (Gentilhomme et al. 2023)

each cane. The percentage of sampled points located within or on the boundary of the corresponding cane segmentation mask was then calculated. This evaluation was performed for the 129 generated canes across the 15 analysed grapevines. The overall mean mask-containment rate was 99.50 %, with values of 99.79 %, 98.65 % and 99.96 % for the *Dão*, *Douro* and 3D2Cut datasets, respectively, as shown in Table 2.

4.3 Implementation efficiency

The computational performance of the Vine2Graph framework was evaluated on an ASUS (ASUS. https://www.asus.com/ (accessed on 09 November 2026)) laptop computer, containing the following specifications: Intel® Core™ i7-12700H (Intel® Core™ i7-12700H, Intel. https://www.intel.com/content/www/us/en/products/sku/132228/intel-core-i712700h-processor-24m-cache-up-to-4-70-ghz/specifications.html (accessed on 09 November 2026)), 32 GB of RAM, and a NVIDIA GeForce RTX 4060 Graphics Processing Unit (GPU) (NVIDIA GeForce RTX 4060, NVIDIA Corporation. https://www.nvidia.com/en-eu/geforce/graphics-cards/40-series/rtx-4060-4060ti/ (accessed on 09 November 2026)). The reported processing times correspond

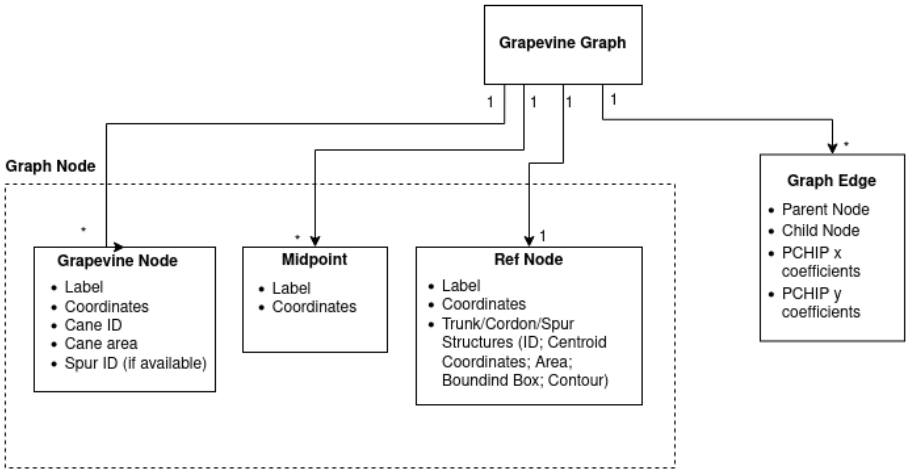


Fig. 8 Structures stored on the grapevine graphs

Table 1 Vine2Graph structural reconstruction performance across three datasets

	Node assignment			Segment connections			Cane reconstruction		
Dataset	**Nodes (GT)**	**Correct nodes**	**Accuracy (%)**	**Generated**	**Correct**	**Accuracy (%)**	**Canes (GT)**	**Correct canes**	**Full cane completeness (%)**
Dão	497	485	97.59	418	416	99.52	77	64	83.12
Douro	256	247	96.48	213	208	97.65	41	31	75.61
3D2Cut	194	194	100.00	170	170	100.00	24	24	100.00
Overall	947	926	97.78	801	794	99.13	142	119	83.80

GT = Ground Truth

Table 2 Geometric evaluation of the parametric cane representation

Dataset	**Generated canes**	**Mean PCHIP mask containment (%)**
Dão	68	99.79
Douro	37	98.65
3D2Cut	24	99.96
Overall	129	99.50

exclusively to the Vine2Graph framework, starting after the required node and segmentation inputs are received by the algorithm and therefore excluding any upstream perception processing. For each dataset, the processing time and memory usage were measured over 10 consecutive executions with the mean value being reported.

To contextualise these measurements, the amount of input information associated with each grapevine example was characterised by the number of grapevine nodes, segmentation masks and pixels occupied by the segmentation masks, as presented in Table 3. The *Dão* example contains the largest amount of structural input information, followed by the *Douro* and 3D2Cut examples. The corresponding computational performance results are presented in Table 4.

The 3D2Cut example presented the lowest mean processing time, followed by the *Douro* and *Dão* examples, respectively. Memory usage showed considerably smaller differences between the three examples. The differences in processing time reflect the different amount and complexity of structural information processed for each grapevine example. Despite these differences, all considered examples were processed in less than approximately 175 ms.

Table 3 Input characteristics of the grapevine examples analysed

Grapevine	Node instances	Segmentation masks	Occupied mask pixels
Dão	107	32	147 385
Douro	55	16	84 229
3D2Cut	43	8	74 428

Table 4 Computational performance of the Vine2Graph algorithm

	Dataset		
	Dão	**Douro**	**3D2Cut**
Mean processing time (ms)	174.86	63.358	42.086
Mean memory usage (MB)	211.78	208.941	208.321

5 Discussion

The main contribution of the developed Vine2Graph framework is the transformation of heterogeneous 2D perception outputs into a unified semantic and topological representation of the grapevine. By organising detected nodes, canes and permanent woody structures within a rooted graph, the framework provides a common representation that can be accessed by downstream robotic modules independently of the perception method used upstream. The examples presented for the *Dão*, *Douro* and 3D2Cut grapevines show that the same framework can represent grapevines with different structural configurations without requiring modifications to the core algorithm. This allows the resulting graph to serve as a semantic representation that can be used

by the downstream modules according to the requirements of the intended application. At the vineyard level, this plant representation also supports management strategies based on the structural variability between individual grapevines. When associated with the localisation of each plant and stored over time, the generated representations can provide useful information for additional tasks regarding vineyard monitoring.

Furthermore, by analysing the performance differences between the three datasets used with Vine2Graph, lower full cane completeness was observed for both the *Dão* and *Douro* datasets when compared with the 3D2Cut dataset. These two datasets presented a higher number of canes and nodes, together with a more pronounced overlap between cane structures. This increased structural complexity and occlusion may reduce the visible continuity of individual canes in the 2D representation, consequently making their reconstruction more challenging for the algorithm. This performance difference in cane completeness can also be partly explained by the criterion used for this metric. Cane components are only generated when at least two grapevine nodes are available, meaning that expected canes with fewer than two represented nodes are not included in the final graph. Furthermore, since full cane completeness requires the complete expected node structure and order to be correct, even a small number of missing or incorrectly associated nodes or connections causes the entire cane to be considered incomplete. Nevertheless, the segment connection accuracy remained high, reaching 99.52 % for the *Dão* dataset, 97.65 % for the *Douro* dataset and 100 % for the 3D2Cut dataset. This indicates that, among the connections generated by Vine2Graph, most consecutive node relationships were correctly established even when the entire cane was not completely reconstructed. Also, for both the *Dão* and *Douro* datasets, Vine2Graph achieved full cane completeness above 75 % and node assignment accuracies above 96 %.

Regarding the parametric cane representation, the PCHIP curves achieved an overall mean mask containment rate of 99.50 %, with the three datasets presenting values above 98 %. These results show that the generated parametric curves remained predominantly within the corresponding segmented cane regions, supporting the geometric consistency of the representation.

The Vine2Graph framework also remains independent from the specific perception modules used to provide the node detections and structural segmentation masks. Therefore, these perception methods may be replaced without modifications to the framework, as long as they provide the data expected by Vine2Graph in the required input format. Nevertheless, the quality of the graph representation will be dependent on the quality of the detections and segmentation masks provided by the models. To reduce the inclusion of structures not belonging to the analysed grapevine, Vine2Graph evaluates the proximity between the base of each cane and the permanent woody structures, represented by the trunk, cordons and spurs. This filtering provides additional consistency to the generated representation, reducing the inclusion of canes from adjacent grapevines.

Regarding computational performance, the evaluated examples presented processing times below 175 ms, and a maximum mean memory usage of approximately

212 MB. These processing times correspond to graph generation rates of approximately 6 to 24 Hz, which may be compatible with perception-level robotic applications depending on the requirements of the system.

6 Conclusion

This work proposes a novel algorithm, Vine2Graph, to define grapevine structures based on 2D images of grapevines and further store them as rooted tree graphs. It is expected that this framework can support the development of decision modules for robotic grapevine applications, particularly by providing the structural information required for pruning decisions. Across the evaluated examples, Vine2Graph was capable of generating structured graph representations of grapevines with different structural configurations, while storing semantic information regarding the different elements of the plant and the parametric geometry of the canes. The quantitative evaluation further showed high node assignment and segment connection accuracy, while the parametric cane representations remained predominantly contained within their corresponding segmentation masks.

Moreover, the implementation of the proposed framework in ROS2 facilitates its integration into robotic systems and supports the modularity of Vine2Graph. The algorithm publishes the constructed graph representations as custom ROS2 messages, which can then be subscribed to by downstream modules responsible for applying decisions regarding grapevine pruning. Furthermore, the generated graph data can be stored and accessed by external modules beyond the pruning operation, providing information regarding the structure of individual grapevines over time. Nonetheless, for long-term use of the generated data, it is relevant to localise the grapevines from which the data were obtained. In the future, the expected next stages of development include the integration of Vine2Graph with localisation and mapping algorithms, enabling the association of the reference node of the 2D representation with a coordinate position in the world. Achieving this could support long-term monitoring of grapevine changes across seasons, as well as the separation of the perception and actuation stages when convenient to do so.

Author contributions. **Francisco Oliveira**: Conceptualisation, methodology, data curation, investigation, software, visualisation, writing - original draft. **Daniel Queirós da Silva**: Data curation, investigation, writing - review and editing. **Mário Cunha**: Supervision, writing - review and editing. **José Boaventura Cunha**: Supervision, writing - review and editing. **Filipe N. Santos**: Funding acquisition, supervision, writing - review and editing.

Acknowledgements. This work is co-funded by the European Regional Development Fund (ERDF) through the Innovation and Digital Transition Programme (COMPETE 2030) under Portugal 2030, within the scope of the project ecoBOTsPT, with reference 25460 (COMPETE2030-FEDER-03032800).

Francisco Oliveira was funded by the Portuguese Foundation for Science and Technology (FCT) through the scholarship 2026.00443.BD.

Declaration of competing interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

Bechar, A., Vigneault, C.: Agricultural robots for field operations: Concepts and components. Biosystems Engineering **149**, 94–111 (2016) https://doi.org/10.1016/j.biosystemseng.2016.06.014

Chen, Z., Wang, Y., Tong, S., Chen, C., Kang, F.: Grapevine branch recognition and pruning point localization technology based on image processing. Applied Sciences **14**(8) (2024) https://doi.org/10.3390/app14083327

Fernandes, M., Scaldaferri, A., Guadagna, P., Fiameni, G., Teng, T., Gatti, M., Poni, S., Semini, C., Caldwell, D., Chen, F.: Towards Precise Pruning Points Detection using Semantic-Instance-Aware Plant Models for Grapevine Winter Pruning Automation (2021). https://doi.org/10.48550/arXiv.2109.07247

Gebbers, R., Adamchuk, V.I.: Precision agriculture and food security. Science **327**(5967), 828–831 (2010) https://doi.org/10.1126/science.1183899

Gentilhomme, T., Villamizar, M., Corre, J., Odobez, J.-M.: Towards smart pruning: Vinet, a deep-learning approach for grapevine structure estimation. Computers and Electronics in Agriculture **207**, 107736 (2023) https://doi.org/10.1016/j.compag.2023.107736

Hagberg, A.A., Schult, D.A., Swart, P.J.: Exploring network structure, dynamics, and function using networkx. In: Varoquaux, G., Vaught, T., Millman, J. (eds.) Proceedings of the 7th Python in Science Conference, Pasadena, CA USA, pp. 11–15 (2008). https://doi.org/10.25080/TCWV9851

Klocker, N., Dun, O., Head, L., Gopal, A.: Exploring migrants' knowledge and skill in seasonal farm work: more than labouring bodies. Agriculture and Human Values **37** (2020) https://doi.org/10.1007/s10460-019-10001-y

Mccorkle, D., Dudensing, R., Hanselka, D., Hellman, E.: The long-term viability of us wine grape vineyards: Assessing vineyard labour costs for future technology development. International Journal of Entrepreneurship and Small Business **36**, 308 (2019) https://doi.org/10.1504/IJESB.2019.097734

Macenski, S., Foote, T., Gerkey, B., Lalancette, C., Woodall, W.: Robot operating system 2: Design, architecture, and uses in the wild. Science Robotics **7**(66), 6074 (2022) https://doi.org/10.1126/scirobotics.abm6074

Navone, A., Martini, M., Chiaberge, M.: Autonomous Robotic Pruning in Orchards and Vineyards: a Review (2025). https://doi.org/10.48550/arXiv.2505.07318

Oliveira, F., Silva, D.Q., Filipe, V., Pinho, T.M., Cunha, M., Cunha, J.B., Santos, F.N.: Enhancing grapevine node detection to support pruning automation: Leveraging state-of-the-art yolo detection models for 2d image analysis. Sensors **24**(21) (2024) https://doi.org/10.3390/s24216774

Oliveira, F., Tinoco, V., Valente, A., Pinho, T., Cunha, J.B., Santos, F.N.: Pruning end-effectors state of the art review. In: Santos, M.F., Machado, J., Novais, P., Cortez, P., Moreira, P.M. (eds.) Progress in Artificial Intelligence, pp. 169–180. Springer, Cham (2025). https://doi.org/10.1007/978-3-031-73497-7_14

Pérez, L.H., Aguilar, M.C.M., Sánchez, N.M., Montesinos, A.F.: Path planning based on parametric curves. In: Róka, R. (ed.) Advanced Path Planning for Mobile Entities. IntechOpen, Rijeka (2017). Chap. 7. https://doi.org/10.5772/intechopen.72574

Strub, L., Stoll, M., Loose, S.M.: The effects of low-input training systems on viticultural costs on flat terrain and steep slope sites. OENO One **55**(2), 415–429 (2021) https://doi.org/10.20870/oeno-one.2021.55.2.4619

Silwal, A., Yandun, F., Nellithimaru, A., Bates, T., Kantor, G.: Bumblebee: A path towards fully autonomous robotic vine pruning. Field Robotics **2**, 1661–1696 (2022) https://doi.org/10.55417/fr.2022051

Tinoco, V., Silva, M.F., Santos, F.N., Rocha, L.F., Magalhães, S., Santos, L.C.: A review of pruning and harvesting manipulators. In: 2021 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), pp. 155–160 (2021). https://doi.org/10.1109/ICARSC52212.2021.9429806

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, İ., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods **17**, 261–272 (2020) https://doi.org/10.1038/s41592-019-0686-2

Yang, Q., Yuan, Y., Chen, Y., Xun, Y.: Method for detecting 2d grapevine winter pruning location based on thinning algorithm and lightweight convolutional neural network. International Journal of Agricultural and Biological Engineering **15**, 177–183 (2022) https://doi.org/10.25165/j.ijabe.20221503.6750