The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLIX-B2-2026
XXV ISPRS Congress 2026 “From Imagery to Understanding”, Commission II, 4–11 July 2026, Toronto, Canada

An Approach for deriving Branch Kinematics of Deciduous Trees from hyper-temporal terrestrial Laser Scanner Data

Anne Bienert, Katja Richter, Sophia Böhme, Hans-Gerd Maas

TUD Dresden University of Technology, Institute of Photogrammetry and Remote Sensing, 01062 Dresden, Germany
(anne.bienert, katja.richter1, hans-gerd.maas)@tu-dresden.de, sophia.boehme1@mailbox.tu-dresden.de

Keywords: Hyper-temporal, Branch motion analysis, Terrestrial laser scanning, ICP, Deciduous tree

Abstract

Understanding vegetation dynamics in three-dimensional, high-temporal resolution is essential for advancing ecological research and sustainable forest management. This study introduces a novel methodology for tracking branch kinematics in trees using hyper-temporal terrestrial laser scanning (TLS) data. Focusing on a solitary pedunculate oak (*Quercus robur*) over a one-year period, we employed a geometric feature detection algorithm combined with quantitative structure modeling (QSM) to identify and track distinctive point cloud sections on first- and second-order branches. By leveraging an iterative closest point (ICP) alignment process, branch kinematics were analyzed across multiple epochs, yielding detailed three-dimensional movement trajectories. The results demonstrate that branch movements exhibit screw-shaped patterns. Temporal resolution analysis revealed that a one-week recording interval is sufficient for our study subject to reliably capture kinematic dynamics, whereas longer intervals (e.g., three weeks) result in significant deviations from actual trajectories. The proposed method proved robust against partial occlusions from leaf growth but struggled under extensive occlusions. This research highlights the potential of hyper-temporal TLS for non-contact, high-resolution monitoring of tree canopy dynamics and provides a foundational approach for future studies aimed at modeling vegetation movement and structural changes over time.

1. Introduction

Understanding vegetation dynamics at high temporal and spatial resolution in 3D is crucial for advancing ecological research and informing sustainable forest management. Recent advances in terrestrial laser scanning (TLS) facilitate continuous, fine-scale monitoring of vegetation structure and growth processes. In this context, hyper-temporal TLS has emerged as a powerful tool for quantifying temporal changes in vegetation (Olivier et al., 2017; Campos et al., 2021; Bienert et al., 2024).

Olivier et al. (2017) demonstrated the utility of hyper-temporal TLS by quantifying crown dynamics in deciduous and coniferous trees, revealing for instance patterns of crown space colonization. Bienert et al. (2024) investigated the use of high-frequency hyper-temporal TLS data in combination with 3D tree parameters and point cloud comparisons to quantitatively describe deciduous tree growth. Moreover, TLS acquisitions at hourly intervals have been employed to detect sub-circadian branch movements driven by tree turgor (Puttonen et al., 2016; Zlinszky et al., 2017; Junttila et al., 2022), with Junttila et al. (2022) linking these movements to variations in the trees' water balance. Collectively, these studies highlight the potential of hyper-temporal TLS for capturing fine-scale structural and physiological dynamics in trees.

In Bienert et al. (2024) we presented a first approach for tracking branch movements in hyper-temporal TLS data. The study demonstrated that the examined deciduous tree exhibited distinct branch movements across all scan epochs. These movements were attributed to factors such as leaf growth and the associated increase in branch weight, as well as rainfall events that caused temporary branch displacement shortly after water absorption. However, this approach is constrained to only a limited number of manually selected branch points within the crown.

Figure 1. Pedunculate oak (*Quercus robur*) under leaf-off (left) and leaf-on (right) conditions.

To address these limitations, we propose a novel approach for tracking branch movements at numerous points detected automatically. This study investigates the potential of hyper-temporal TLS data for branch motion analysis over the course of an entire year.

2. Study Site and Data Acquisition

The study site is located in Dresden (51°1'46" N, 13°43'29" E), Germany, and contains a pedunculate oak (*Quercus robur*). This solitary street tree (Figure 1) does not interact with surrounding trees. It attains a height of roughly 12.2 m, a diameter at breast height of 0.22 m, and is situated in an urban environment dominated by buildings.

Hyper-temporal TLS data were acquired with a RIEGL VZ-400i terrestrial laser scanner at a scan resolution of 40 mdeg and comprise between 1.17 million and 2.62 million laser points. The TLS data acquisition was carried out during one year. The dataset includes laser scans from April to January, which were taken daily from the beginning of leaf emergence until the leaves were fully developed (from April to May). After leave development we turned over to a weekly scan acquisition. The scans were performed exclusively during near-calm wind conditions to ensure data quality. As soon as a major gust of wind was registered, the scan was immediately interrupted and subsequently repeated to ensure the accuracy of the recordings. Each scan epoch was captured from four diametrically arranged scan positions. The scan positions were registered to each other using 11 static tie points arranged around the tree on buildings and traffic signs. Each epoch was co-registered to the initial epoch, named epoch E_0, which was recorded under leaf-off conditions on April the 24th.

3. Methods

The objective of this paper is to track and analyze branch kinematics in the hyper-temporal dataset. The basic idea is to automatically identify distinctive point cloud areas, which are then tracked across the scan epochs. For this purpose, prominent tracking points are first detected in the point cloud using a geometric feature detector (Section 3.1). Subsequently, tracking points located on first- and second-order branches are selected using a quantitative structure model (QSM) (Section 3.2). Finally, the point cloud regions around the tracking points are tracked across all scan epochs using an iterative closest point (ICP)-based tracking approach to analyze branch kinematics (Section 3.3). An overview of the general workflow is illustrated in Figure 2. Section 3.4 investigates the required temporal resolution of the data by systematically thinning the available epochs and analyzing the impact of different temporal sampling intervals on the detection of movement patterns.

3.1 Automatic Detection of Tracking Points

As a first step, distinctive points in the point cloud are identified, which will later serve as the basis for detecting prominent point cloud regions for branch kinematic tracking. For this purpose, the geometric feature detector Harris 3D is used (Sipiran and Bustos, 2011). It initially identifies a large number of distinctive points within the tree crown and along the trunk. First, trunk points are efficiently removed by placing a vertical cylinder with a radius r_S around the tree's position, retaining only the points outside the cylinder. Finally, to optimize processing time, the large number of tracking points is reduced by iteratively retaining a point and removing all other points within a predefined distance d. This process is repeated until no points remain to process.

3.2 Filtering of Tracking Points

From all points resulting from the automatic detection of tracking points, only those located on strong branches are used for the analysis of branch kinematics. Points on small, thin twigs are not considered. The reason for this is that finer branches often lack visible branch structures due to leaves. The filtering of the tracking points is realized using a QSM, which provides a hierarchical classification of branches based on their position relative to the main trunk and other branches. For the analysis of branch kinematics, first-order branches that grow directly from

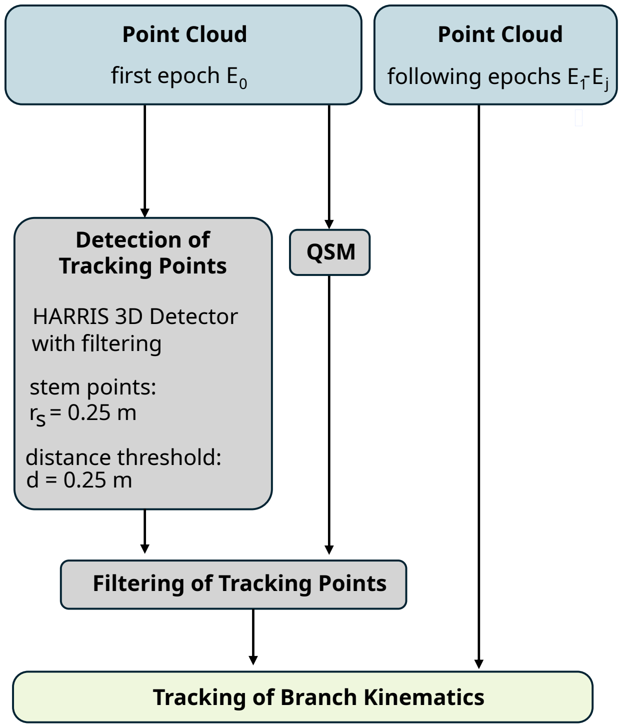

Figure 2. Workflow of the automatic branch motion analysis, with two key parameters: vertical cylinder radius r_S and distance threshold d for filter purpose (Section 3.1).

the trunk of the tree and second-order branches that grow out from the first-order branches are used.

The QSM was generated with the *TREEQSM* software (version 2.30) developed by Raumonen et al. (2013). This software is specifically designed to generate 3D tree models from TLS point clouds by segmenting trees into cylindrical components. By connecting these fitted cylinders, the software enables detailed analysis of branch topology, geometry, and volume. *TREEQSM* was run in *Matlab*® (MathWorks, Natick, MA, USA) version R2023a. The QSM was created based on a downsampled point cloud of epoch E_0, ensuring a manageable dataset for accurate modeling. Several reasonable parameter settings were tested (grid search algorithm) to determine which one worked best. For each setting, it was calculated how well the points fitted the generated cylinders using the select_optimum function in *TREEQSM*. For this purpose, the average distance between each cylinder and its closest points was computed. Then, these average distances were summed across all cylinders. The parameter setting with the smallest total distance was selected as the best one.

Subsequently, the QSM was used to filter out those tracking points from the entire set that lay on first- and second-order branches. For this purpose, a suitable buffer was applied around the first- and second-order branches derived from the QSM. Tracking points falling within this buffer b were then selected for further analysis, resulting in a set of points P^j (where j denotes the current number of tracking point and m the total number of tracking points, with $j = 1, \ldots, m$).

3.3 Tracking of Branch Kinematics

To automatically identify distinctive areas within the point cloud for tracking branch kinematics, we extract point cloud sections around the tracking points P_i^j in the initial epoch $i = 0$ using a predefined distance threshold d_0. Each extracted section is refined using a noise filter, where all points lacking neighbors within a radius r_g are removed. This ensures that the accuracy of branch kinematic tracking is not affected by noise.

Subsequently, the motion of the extracted point cloud sections is tracked across all subsequent epochs. For this purpose, the following steps are carried out iteratively for each point cloud section in epoch E_i (where i denotes the current epoch and n the total number of epochs, with $i = 1, \ldots, n$):

1. extraction of similar point cloud section around tracking point P_{i-1}^j in current epoch *i*,
2. refinement of point cloud section with a noise filter,
3. alignment of both point cloud sections (E_0 and E_i),
4. transformation of tracking point P_0^j into current epoch *i*.

In the first step, we use the tracking points P_{i-1}^j (with $j = 1, \ldots, m$) from epoch E_{i-1} as an approximation to identify and extract corresponding regions of the point cloud in the current epoch E_i. As described above, a distance threshold is used for this purpose, and after cropping, a noise filter is applied to refine the point cloud sections.

Afterwards, we align the corresponding point cloud sections using a generalized ICP (Segal et al., 2009) algorithm from the *PCL* library (version 1.14.1). This algorithm assigns higher weights to scanned points in locally dense regions to favor reliable measurements from densely sampled branches over noisy leaf measurements. The point cloud sections are considered to be successfully aligned if the sigma of the ICP alignment does not exceed the threshold σ_{ICP}.

In the fourth step, each initial tracking point P_0^j is transformed into the current epoch E_i by applying the transformation matrices derived from the ICP alignments, resulting in the updated tracking points P_i^j. Please note, that the algorithm simply projects the initial tracking points P_0^j from epoch E_0 to the aligned location in the current epoch E_i. It is irrelevant whether scan points are present in this area (e.g., due to occlusion caused by leaves). This represents an innovation compared to the algorithm presented in Bienert et al. (2024), as occluded areas at the tracking point are now also included.

The updated tracking points, P_i^j, serve as the input values for the next iteration $(i + 1)$, during which the branch kinematics between epoch E_0 and epoch E_{i+1} are tracked. This ensures that the extracted point cloud section continues to include the same distinctive areas as in epoch E_0, even if the branch undergoes significant overall movement. In this way, steps 1-4 are repeated until the distinctive area is tracked through all epochs, resulting in the updated tracking points P_i^j (with $i = 1, \ldots, 59$ and $j = 1, \ldots, m$).

Finally, the 3D motion trajectory T_j of the distinctive point cloud section across all epochs is derived by fitting a cubic spline to the initial and updated tracking points. At least 5 tracking points must be available in order to fit a cubic spline. For visualization purposes, it is advantageous to choose the coordinate system such that the Z-axis points along the QSM vector direction. This enables the visualization of the branch movements over time in relation to the trunk.

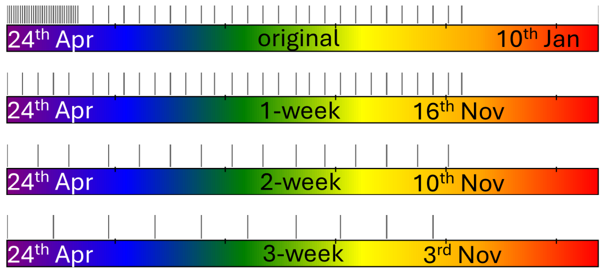

Figure 3. Systematic thinning of the original dataset to simulate lower temporal resolution. Due to the different time frames, the measurement period for the sampling intervals already ends in November.

3.4 Temporal Resolution

Since capturing a high-resolution hyper-temporal dataset is highly labor-intensive, this study aims to determine the minimum temporal resolution required to reliably detect branch kinematics. For this purpose, the original dataset is systematically thinned out on the time axis to simulate datasets with lower temporal resolution. Specifically, a rhythm of 1-week (30 epochs), 2-week (15 epochs), and 3-week (10 epochs) sampling recordings is implemented (Figure 3).

The tracking of branch kinematics is performed for all simulated temporal resolutions, and the trajectory of the branch movement is determined. As the first quality criterion, the standard deviations of the residuals (a measure of the typical size of the deviations of the tracking points from the fitted spline function) are calculated. To evaluate the effect of the sampling time interval on the detection of branch kinematics, the trajectory splines generated at different temporal resolutions are compared both visually and quantitatively. For the quantitative comparison, the distances between the spline functions are analyzed. For this purpose, the spline functions of the original dataset are sampled at discrete points, and the minimal distances between the resulting spline points of the sampling time intervals are calculated. This approach provides a quantitative assessment of the variability between splines across different temporal resolutions.

4. Results

4.1 Automatic Detection of Tracking Points

The Harris 3D detector identified 15,502 distinctive points within the point cloud of epoch E_0. These points were distributed across the entire tree, including both the crown and the trunk. The trunk points were removed by placing a cylindrical filter with a radius r_S of $0.25\,\mathrm{m}$ around the tree's position, eliminating all points located within the cylinder. This step resulted in the removal of $4\,\%$ of the points. Finally, to achieve an even distribution of points across the tree canopy, only those points were retained that had no other candidates within a distance *d* of $0.25\,\mathrm{m}$. This meticulous cleanup process significantly reduced the dataset, leaving 3,053 tracking points that were evenly distributed and well-suited for further analysis.

4.2 Filtering of Tracking Points

Initially, a QSM was generated, serving as the foundation for filtering the tracking points. For the generation of the QSM, the

Hyperparameter	Ranges [m]
PatchDiam1	[0.01, 0.03, 0.05]
PatchDiam2Min	[0.008, 0.013, 0.018, 0.023]
PatchDiam2Max	[0.035, 0.05, 0.065, 0.08]

Table 1. Hyperparameter grid search - hyperparameters and ranges.

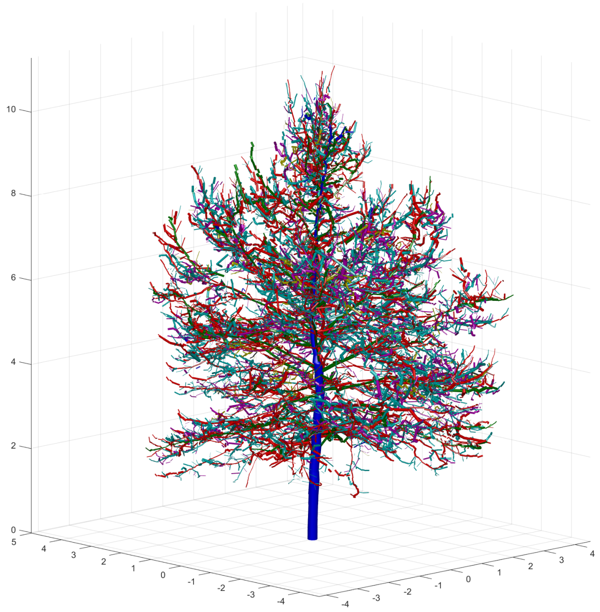

Figure 4. QSM of epoch E_0. The stem is highlighted in blue and the branch orders are distinguished by different colors.

dataset E_0 was downsampled to a voxel size of 0.01 m to ensure a computationally efficient resolution, whereas the subsequent analyzes were performed using the original data. Before starting the QSM generation, a grid search was performed to determine the optimal parameter settings. The parameters and their ranges used for the grid search are presented in Table 1. The optimal parameter set for the three key parameters was determined as follows: PatchDiam1 = 0.05 m, PatchDiam2Min = 0.008 m, and PatchDiam2Max = 0.035 m. These parameters were carefully selected to ensure accurate and reliable modeling results, balancing precision and computational efficiency.

The final QSM comprised 38,760 fitted cylinders, capturing structural details up to a branch order of 8 (Figure 4). When focusing solely on the stem and branches of the first- and second-order, the model yielded 13,396 cylinders. The fitted cylinders exhibited radii ranging from 0.004 m to 0.12 m, with lengths extending up to 0.53 m. Finally, all tracking points located within a buffer b = 0.05 m around the first- and second-order branches from the QSM were filtered out. This resulted in a final point set P_0^j of 883 initial tracking points.

4.3 Tracking of Branch Kinematics

The branch kinematics were analyzed at each of the 883 designated initial tracking points. The extraction of the distinctive point cloud sections around each initial tracking point P_0^j was performed with a distance threshold $d_0 = 0.40$ m and a noise filter radius of $r_g = 0.02$ m. Subsequently, the point cloud sections were tracked through all epochs. Figure 5 illustrates the tracking of a point cloud section between epoch E_0 and epoch E_{41}.

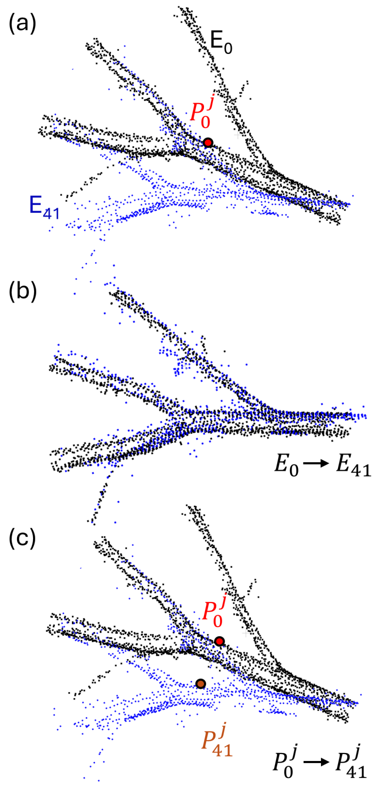

Figure 5. Exemplary tracking of branch kinematics. (a) co-registered point cloud sections of epoch E_0 (black) and E_{41} (blue) with scan shadows on the branch due to occlusion by leaves in E_{41}; (b) point cloud sections after ICP alignment; (c) transformation of P_0^j in epoch E_{41} using the transformation matrix derived from ICP alignment resulting in P_{41}^j.

The σ_{ICP} threshold was set to 0.05 m. If the ICP algorithm fails to converge or if the sigma value exceeds the threshold, the tracking of the point cloud section is interrupted. In this way, a trajectory was determined for each of the 883 tracking points.

The determination of branch movement trajectories at the 883 initial tracking points revealed varying levels of success. To focus on the best results, the trajectories were filtered based on the standard deviation of their splines. Specifically, trajectories with a standard deviation above 0.02 m were excluded, leaving 58 trajectories for further analysis (Figure 6). All branches exhibited varying degrees of movement. To assess the direction of movement, it is helpful to observe from the trunk toward the branch. First-order branches exhibited screw-shaped movements over the course of the year when observed in the direction of the branch. In addition, they often showed a movement away from the stem.

4.4 Temporal Resolution

Enlarging the recording time interval can result in significant changes between epoch i and i+1 particularly during periods of leaf emergence. These changes may prevent the ICP algorithm from converging successfully. As a result, if the number of updated tracking points falls below 5, no spline function is fitted and no trajectory is determined at the initial tracking point. Consequently, the number of trajectories decreases as the temporal resolution is reduced. In the dataset with the original tem-

Figure 6. Point cloud of epoch E_0 showing 883 initial tracking points (black dots) and 48 selected tracking points (red dots) used for temporal resolution analysis.

Interval	σ_{min} [m]	σ_{max} [m]	**Count** [-]
Original	0.001	0.020	58
1-week	0.001	0.018	53
2-week	0.001	0.015	50
3-week	0.001	0.019	48

Table 2. Minimum and maximum standard deviation σ_{min} and σ_{max} of the initial and updated tracking points relative to the spline. The final column specifies the number of initial tracking points.

poral resolution, trajectories could be determined for all 58 initial tracking points. However, with the enlarged temporal resolutions, this was only successful for 48 initial tracking points.

We determined the standard deviation of the initial and updated tracking points based on their fitted spline function. Thus, we obtained 58 standard deviations for the original dataset, 53 for 1-week interval, 50 for 2-week interval, and 48 for 3-week interval. Table 2 summarizes the minimum and maximum standard deviations that occurred per time interval. While the minimum standard deviations for all temporal resolutions remained constant at $0.001\,\mathrm{m}$, the maximum standard deviation values ranged between $0.015\,\mathrm{m}$ and $0.020\,\mathrm{m}$. The smallest standard deviation was observed at the 2-week temporal resolution.

The larger the recording time interval, the fewer points are used for the spline fit. It can therefore be assumed that the 1-week, 2-week, and 3-week splines deviate from the original splines. To investigate this further, the minimum distances of the 1-week, 2-week, and 3-week splines to the original splines were plotted in histograms (Figure 7). The histogram shows that as the recording time interval increases, the minimum distances between the original splines and the splines of the respective

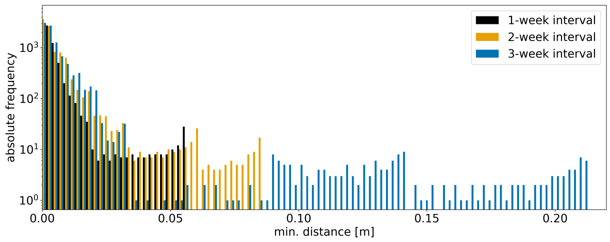

Figure 7. Absolute frequency of minimum distances [m] between splines resulting from simulated and original temporal resolution. The histogram show distance patterns for splines sampled at 1-week (black), 2-week (orange), and 3-week (blue) intervals.

time intervals also increase. The minimum distances for the 1-week interval show values of up to $0.06\,\mathrm{m}$, while the 2-week interval shows values of up to $0.08\,\mathrm{m}$, and the 3-week interval shows values of $0.23\,\mathrm{m}$.

Figure 8 shows the branch kinematics derived from different temporal resolutions. It is evident that the deviation between the trajectory observed at the original temporal resolution and that at a simulated temporal resolution increases systematically with longer temporal sampling interval, a trend that is consistently observable across all three coordinate projections. Notably, a substantial discrepancy already emerged at the first reduction in temporal resolution. The fitted splines yielded standard deviations of $0.003\,\mathrm{m}$ (original data), $0.002\,\mathrm{m}$ (1-week), $0.002\,\mathrm{m}$ (2-week), and $0.001\,\mathrm{m}$ (3-week).

Figure 9 presents a comparison of the trajectories derived at the initial tracking point P_0^{2196} for different temporal resolutions. It is easy to see that the movement pattern was reproduced very well in 1-week and 2-week time intervals. The 3-week interval shows the greatest deviations to the original dataset. The obtained standard deviation only provides information about the correctness of the fit, but not about the position of the spline.

5. Discussion

The evaluation of branch kinematics focused on primary and secondary branch structures, as higher-order branches had insufficient point density due to limited scan resolution. A low number of scan points inhibits the reliable determination of both the positions and orientations of QSM vectors in fine branch structures. The accurate positioning of the QSM vector is critical for filtering and selecting the initial tracking points, ensuring that only relevant points are included in the analysis. Furthermore, the orientation of the QSM vector plays a pivotal role in transforming the coordinate system. Specifically, it enables the rotation of the coordinate system such that the Z-axis aligns with the QSM vector's direction. This alignment is essential for maintaining consistency in the spatial representation of branch kinematics and for accurately projecting movement trajectories within the transformed coordinate space. Incorrect QSM vectors can lead to misaligned trajectories and make interpretation of branch kinematics difficult.

The algorithm for tracking of branch kinematics demonstrated robust performance under partial occlusion conditions, as long as the occlusions remain spatially localized and sufficient branch segment information is available to ensure successful ICP alignment. Under these conditions, distinctive point

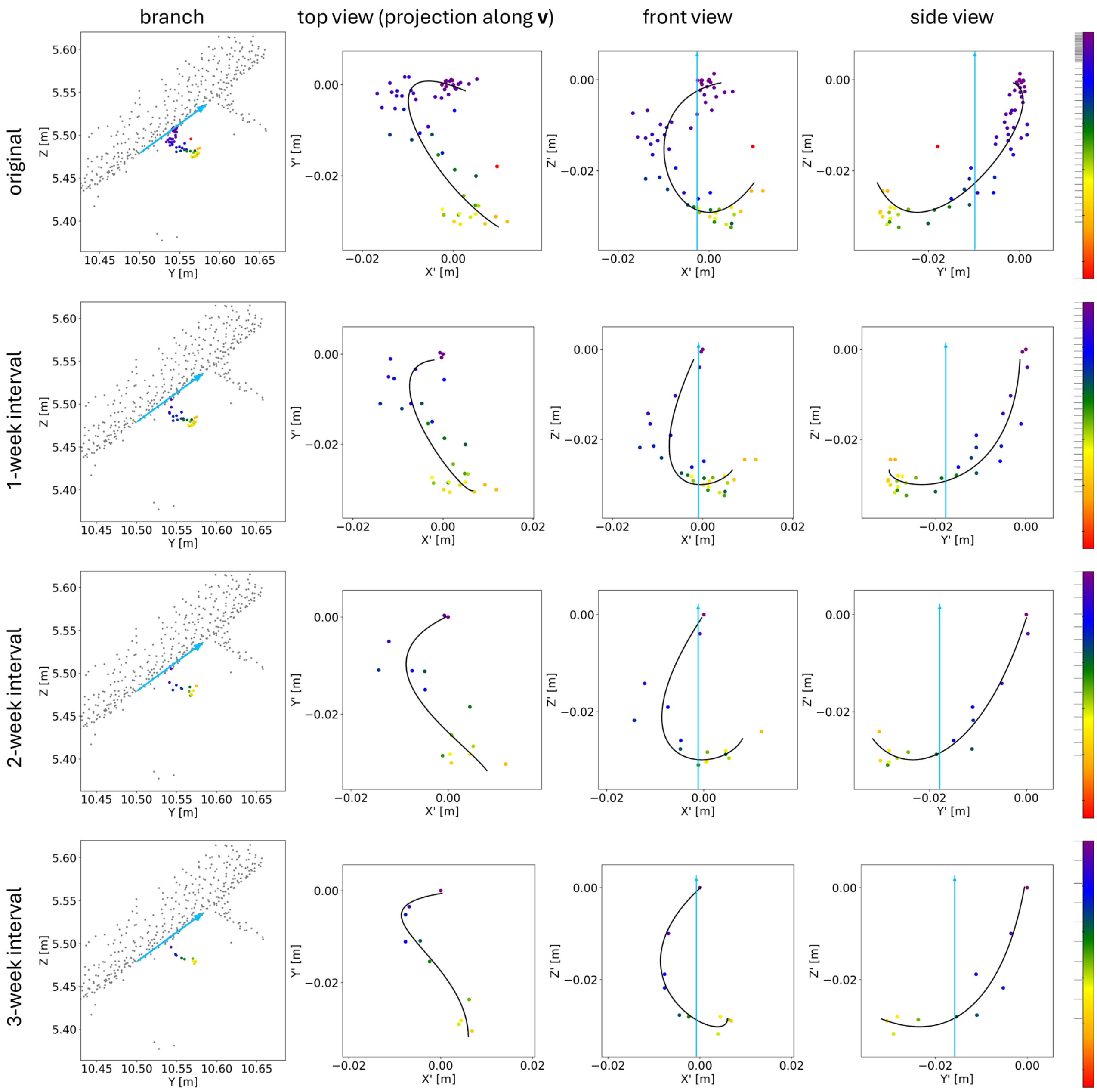

Figure 8. Branch kinematics at the initial tracking point P_0^{2196} at varying temporal resolutions: Column 1: YZ-projection of point cloud section (black) with QSM vector (light blue) and initial and updated tracking points (color coded by recording time), column 2: projection onto the XY'-plane along the QSM vector direction, column 3: front view (XZ' projection), column 4: side view (YZ' projection), each with trajectory spline (black) and QSM vector (light blue). The ticks in the color bar visualize the temporal resolution.

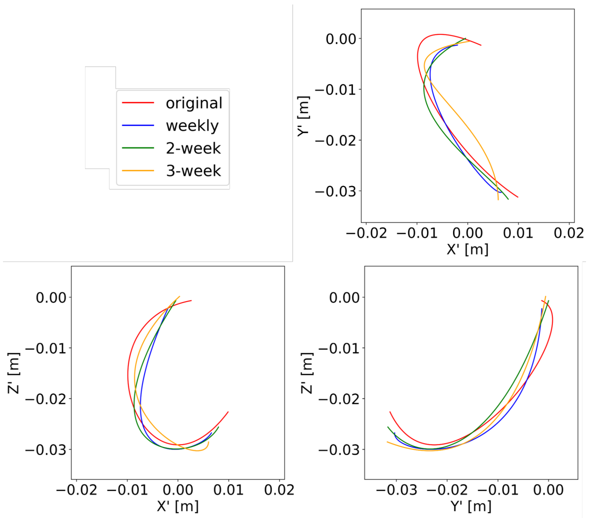

Figure 9. Comparison of the trajectories derived at the initial tracking point P_0^{2196} for different temporal resolutions. red = original interval, blue = 1-week interval, green = 2-week interval, and orange = 3-week interval.

cloud sections could be successfully tracked across the hyper-temporal TLS dataset. However, leaf development poses a challenge to tracking accuracy. As leaves grow, they create extensive occlusions in fine branch structures, reducing the probability of successful ICP alignment in point cloud sections. Over the course of the epochs, the progressive growth of leaves increasingly occluded the point cloud sections.

The simulated 1-week data acquisition preserved sufficient point density to enable accurate tracking of branch kinematics. This was further validated through histogram analysis of minimum distances between the splines, which confirmed the reliability of 1-week recording intervals in capturing dynamic changes over time. Analysis of the temporal resolution showed that extending the measurement intervals to 3-weeks led to a significant reduction in point density in our case (59 vs. 10 epochs). The decline in tracking points below the minimum threshold for reliable tracking hindered the detection of structural changes and compromised the ability to derive meaningful temporal insights. However, the dense temporal spacing at the beginning of the original dataset had a disproportionately large influence on the local curvature and boundary behavior of the spline. The position of the adjusted spline shape changed fundamentally when the intervals were changed, thus underlining the sensitivity of spline interpolation to the spatial and temporal distribution of the underlying initial and updated tracking points.

The recommended temporal resolution of the epochs is influenced by the frequency of data acquisition relative to the dynamics of the observed system. For our tree, a 1-week recording interval was determined to be sufficient, but the temporal resolution required is highly dependent on factors such as the object's movement, size, and the point density of the point cloud.

It is not possible to externally validate our movement trajectories as no conventionally obtained data is available. Individual branches might be equipped with motion sensors to track movement behavior. However, this is limited to a manageable number of points in the crown (economic efficiency) and, on the other hand, the movement is influenced by the weight of a sensor.

6. Conclusions

In this study, we proposed a novel methodology for the automated tracking of branch kinematics using hyper-temporal terrestrial laser scanning data. A particular advantage of the developed algorithm is that movements can be recorded without contact. This makes it possible to analyze a very large number of points in the tree canopy simultaneously. The point clouds obtained via laser scanning are independent of ambient light and can also be used for nighttime hyper-temporal scan evaluations (Puttonen et al., 2016). The developed algorithm is designed to extract any number of points within the tree crown then track the surrounding point cloud sections over multiple epochs. By approximating the trajectories of the initial and updated tracking points, the approach enables the quantification and analysis of branch kinematics. The approach was tested on a street tree, where screw-shaped trajectories were observed on a large number of branches.

The algorithm demonstrates stability against partial occlusions caused by leaves. However, when the occluded areas become too extensive, the algorithm reaches its limits, and the distinctive point cloud sections can no longer be tracked. This issue is mitigated by the dense temporal resolution of the data acquisition process, which ensures that tracking interruptions do not pose a significant problem. The trajectory approximation compensates for missing tracking updates and ensures continuity in the tracking process.

The study showed that the movement trajectory for a simulated recording interval of more than one week deviated significantly from the movement trajectory of the originally recorded epochs. Future work will focus on analyzing branch movement directions and performing validation on additional trees. In addition, the trajectories will be projected into a voxel space to predict occlusions and, if necessary, to model the occluded regions.

Acknowledgements

This study was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project number 519022036. We would like to thank our colleagues at the Institute of Photogrammetry and Remote Sensing (Dresden University of Technology) for their support during the time-consuming data collection.

References

Bienert, A., Richter, K., Böhme, S., Maas, H.-G., 2024. Investigating the potential of hyper-temporal terrestrial laser point clouds for monitoring deciduous tree growth. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48, 33–40.

Campos, M. B., Litkey, P., Wang, Y., Chen, Y., Hyyti, H., Hyyppä, J., Puttonen, E., 2021. A Long-Term Terrestrial Laser Scanning Measurement Station to Continuously Monitor Structural and Phenological Dynamics of Boreal Forest Canopy. *Frontiers in Plant Science*, 11(January), 1–15.

Junttila, S., Campos, M., Hölttä, T., Lindfors, L., El Issaoui, A., Vastaranta, M., Hyyppä, H., Puttonen, E., 2022. Tree Water Status Affects Tree Branch Position. *Forests*, 13(5). https://www.mdpi.com/1999-4907/13/5/728.

Olivier, M.-D., Schneider, R., Fournier, R. A., 2017. A method to quantify canopy changes using multi-temporal terrestrial lidar data: Tree response to surrounding gaps. *Agricultural and Forest Meteorology*, 237, 184–195.

Puttonen, E., Briese, C., Mandlburger, G., Wieser, M., Pfennigbauer, M., Zlinszky, A., Pfeifer, N., 2016. Quantification of Overnight Movement of Birch (Betula pendula) Branches and Foliage with Short Interval Terrestrial Laser Scanning. *Frontiers in Plant Science*, 7(February), 1–13.

Raumonen, P., Kaasalainen, M., Åkerblom, M., Kaasalainen, S., Kaartinen, H., Vastaranta, M., Holopainen, M., Disney, M., Lewis, P., 2013. Fast Automatic Precision Tree Models from Terrestrial Laser Scanner Data. *Remote Sensing*, 5(2), 491–520.

Segal, A., Haehnel, D., Thrun, S., 2009. Generalized-ICP. *Robotics: science and systems*, 2(4), 435.

Sipiran, I., Bustos, B., 2011. Harris 3D: a robust extension of the Harris operator for interest point detection on 3D meshes. *The Visual Computer*, 27(11), 963–976.

Zlinszky, A., Molnár, B., Barfod, A. S., 2017. Not All Trees Sleep the Same—High Temporal Resolution Terrestrial Laser Scanning Shows Differences in Nocturnal Plant Movement. *Frontiers in Plant Science*, 8, 1814.