An Approach for RGB-Guided Dense 3D Displacement Estimation in TLS-Based Geomonitoring

Zhaoyi Wang[1], Jemil Avers Butt[1,2], Shengyu Huang[1], Nicholas Meyer[1], Tomislav Medić[1], Andreas Wieser[1]

[1] Institute of Geodesy and Photogrammetry, ETH Zürich, Switzerland - zhaoyi.wang@geod.baug.ethz.ch
[2] Atlas optimization GmbH, Switzerland

Keywords: Terrestrial laser scanning (TLS), built-in camera, RGB image, landslide, displacement vector field (DVF).

Abstract

Estimating 3D deformation with high spatial resolution from TLS point clouds is beneficial for geomonitoring. Existing methods for this task primarily rely on geometric data. They do not use radiometric information although it is often available as well. This leaves potential for improvement. To address this, we propose an approach that utilizes RGB images—captured by built-in cameras of TLS scanners and co-registered with TLS point clouds—to generate dense 3D displacement vector fields for deformation analysis. Our method comprises three main steps: (1) applying the Efficient-LoFTR algorithm to establish dense 2D pixel correspondences on RGB images across two epochs; (2) projecting 3D points from both epochs onto RGB images and establishing 3D point correspondences by matching the projected pixels with the established 2D correspondences; (3) clustering the point cloud of one epoch and refining the 3D point correspondences within each cluster to produce the final displacement vector fields. Experiments on real measurements obtained from a rockfall simulator and from a real-world landslide demonstrate that our method achieves comparable accuracy to state-of-the-art geometry-based methods, with improved density and computational efficiency. By using radiometric features, our approach complements geometry-based methods, suggesting that combining both will enhance coverage and/or accuracy for geomonitoring applications.

1. Introduction

In geomonitoring applications, estimating 3D deformation can enhance our understanding of Earth’s physical processes, potentially contributing to reducing risks associated with geological hazards such as landslides, rockfalls, and debris flows (Jaboyedoff et al., 2012; Mukupa et al., 2017; Albanwan et al., 2024). One way to generate such 3D data is through the processing 3D point clouds acquired via LiDAR scanners or photogrammetry. Among LiDAR-based sensor systems, terrestrial laser scanning (TLS) scanners have become a promising tool in geomonitoring, favored for their high measurement quality and spatio-temporal resolution.

To obtain dense displacement maps or 3D models of deformation, TLS point clouds from at least two measurement epochs undergo a typical data processing chain, including data preprocessing (*e.g.*, outlier removal and point cloud registration), per point (or per group of points) correspondence establishment, and estimation of displacements between these corresponding points. All three steps affect the information content and quality of the estimated displacements. However, in this study we focus on the latter stages: establishing correspondences and estimating displacements, often performed within a single algorithm.

Common approaches, such as C2C (Girardeau-Montaut et al., 2005), C2M (Cignoni et al., 1998) and traditional M3C2 (Lague et al., 2013), establish point-to-(point; approximated surface; averaged points) correspondences based on computing distances in Euclidean space, often directed along local surface normals, and provide per-point signed or unsigned displacement magnitudes (Euclidean distances between matches). While effective in many applications, these methods have notable limitations. Namely, establishing correspondences directly in Euclidean space results in incorrect matches when displacements occur along the surface of the sampled terrain, a common case in geomonitoring. Additionally, the “directed” nature of establishing correspondences makes these methods sensitive only to displacements in predefined directions. Hence, these approaches inevitably capture only a projection of the full 3D displacement, resulting in underestimation or omission of significant displacements. Even some state-of-the-art M3C2 variants (James et al., 2017; Zahs et al., 2022; Yang and Schwieger, 2023) do not fully resolve these issues, as they primarily focus on refining the quality, *e.g.*, by local patch or plane-based filtering, rather than addressing the limitations in dense 3D displacement estimation. Consequently, these methods offer insight into the distribution of 1D displacements within 3D space but fail to capture the full extent of 3D displacement fields.

To address these challenges, alternative approaches emerged, capable of providing 3D displacement vector fields (DVFs). Some of these approaches entail “Piecewise ICP” and related alternatives (Teza et al., 2007; Friedli and Wieser, 2016; Wujanz et al., 2018), which partition point clouds into tiles and estimate per-tile displacements as translation vectors obtained through the ICP algorithm (Besl and McKay, 1992; Chen and Medioni, 1992; Bergevin et al., 1996) or some of its variants. In this case, the per-tile correspondences are also established directly in the Euclidean space. Despite being effective solutions for some cases, this class of algorithms usually generates sparser 3D DVFs (per tile and not per point) and produces biased estimates if per-tile rigid body motion is violated.

Some approaches use image representations of the acquired point clouds, establishing correspondences in 2D image space. These methods, often based on hillshade representations of terrain geometry and techniques like image correlation, optical flow or feature tracking (Fey et al., 2015; Holst et al., 2021; Teo et al., 2023), are more effective when motion occurs along terrain surfaces. However, they also result in sparse DVFs due to the algorithms used. Additionally, multi-directional

https://doi.org/10.5194/isprs-annals-X-G-2025-953-2025 | © Author(s) 2025. CC BY 4.0 License.

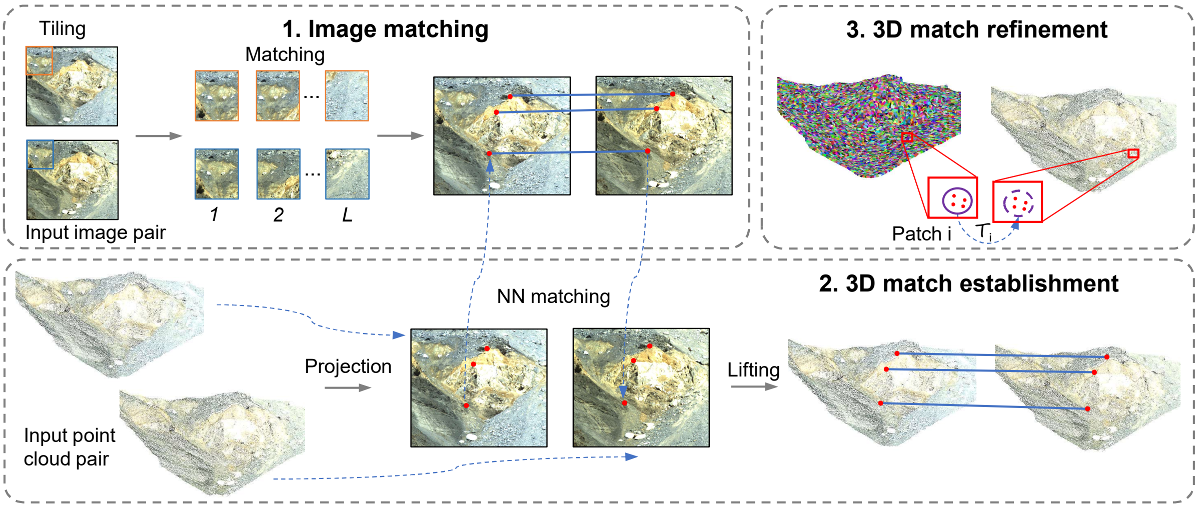

Figure 1. Method overview. **(1)** Source and target images, corresponding to point clouds before and after the deformation epoch, are tiled into L sub-images. A deep learning-based algorithm is used to perform image matching between source and target sub-images, generating 2D pixel correspondences. **(2)** Source and target point clouds are projected onto their respective images. For each projected pixel from the source point, its closest 2D pixel correspondence is found via NN matching. This pixel correspondence is used to find the projected pixel from the target point via NN matching. The projected source and target pixels are lifted back to 3D space to establish 3D point correspondences. **(3)** The source point cloud is clustered into small patches using a deep learning-based segmentation algorithm. Within each patch, point correspondences are used to estimate an initial rigid transformation, followed by a refined estimation of the transformation. This refined transformation is then applied to all matched source points within the patch.

M3C2 (Williams et al., 2021), a variant of M3C2, adaptively selects the direction of meaningful estimations. In contrast, F2S3 (Gojcic et al., 2020) estimates 3D DVFs by establishing per-point correspondences using deep learning based 3D point feature descriptors, followed by deep learning-based outlier removal. F2S3 has shown strong performance in multiple case studies (Gojcic et al., 2021; Kenner et al., 2022).

All these above approaches share a fundamental limitation: they rely on geometric information to establish point correspondences, requiring sufficient geometric structure (or variability) to ensure discrimination and correct matches. In regions with poor geometric structure, *e.g.*, extended planar areas or areas with repetitive patterns, their effectiveness diminishes. In this study, we propose an alternative algorithm for establishing point correspondences and estimating displacements from TLS data, which addresses this limitation and expands the available algorithm toolbox. The algorithm leverages built-in RGB cameras, now standard in TLS systems, which are aligned with the scanning unit through manufacturer calibration. This allows for establishing direct 3D point-to-2D pixel correspondences. Hence, it enables the use of a different data modality (RGB radiometric instead of geometric data) to indirectly establish 3D point correspondences from 2D pixel correspondences across epochs, potentially improving results in geometrically poor regions. Furthermore, to capitalize on recent advancements in image processing and mitigate some aforementioned limitations of traditional methods, we employ one of the state-of-the-art deep learning-based algorithms Efficient-LoFTR (Wang et al., 2024) for establishing epoch-wise pixel correspondences.

To validate our method, we compare it with an early 3D DVF estimation approach: Piecewise ICP (Friedli and Wieser, 2016), and a state-of-the-art geometry-based approach for 3D DVF estimation: F2S3 (Gojcic et al., 2021). The latter method also provides the highest density of 3D DVFs among all the aforementioned algorithms. We conduct comparisons using both a physically emulated rockfall dataset and a real-world landslide monitoring dataset. The implemented method is described in the following section (Sec. 2), with experiments presented in Sec. 3, results and discussion in Sec. 4, and conclusion in Sec. 5.

2. Methodology

Given the source (first epoch) and target (second epoch) point clouds along with their associated RGB images (with known camera intrinsic and extrinsic information), our objective is to estimate the 3D DVFs between the two epoch-wise point clouds. This is achieved through a three-step process: first, we establish 2D pixel correspondences by matching source and target images (*cf.* Sec. 2.1); next, we derive 3D point correspondences from these 2D pixel correspondences (*cf.* Sec. 2.2); and finally, we refine the 3D point correspondences to produce the final 3D DVFs (*cf.* Sec. 2.3). An overview of our method is depicted in Fig. 1. Although the method is designed for two epochs, it is readily generalizable to multi-epoch scenarios by sequentially processing pairs of epochs.

2.1 Image matching

Image matching. Unlike traditional and detector-based methods (Lowe, 2004; Dusmanu et al., 2019; Sarlin et al., 2020), detector-free methods bypass keypoint detection, making them more robust in regions with poor texture or under extreme changes in viewpoints or illumination. A representative of detector-free methods is LoFTR (Sun et al., 2021), which is widely used in various applications, including remote sensing (Ioli et al., 2023). LoFTR operates on a coarse-to-fine matching mechanism: it first establishes dense pixel correspondences at a coarse resolution, incorporating with self- and cross-attention layers within a Transformer (Vaswani et al., 2017), and then refines the matches within local patches at a fine resolution.

We employ Efficient-LoFTR (Wang et al., 2024), an enhanced version of LoFTR, to establish epoch-wise pixel correspondences. Efficient-LoFTR improves upon its predecessor in both efficiency and matching accuracy. Specifically, it optimizes the matching process by first aggregating similar features in local regions, then applying attention layers on selected tokens. It further enhances accuracy by refining matches after the coarse-to-fine matching through correlation and expectation operations.

Image tiling. Instead of performing image matching directly on the original RGB images, we first tile the images to improve matching efficiency and reduce memory consumption. The tiling strategy is designed to account for the maximum displacement by incorporating an appropriate overlap between tiled images.

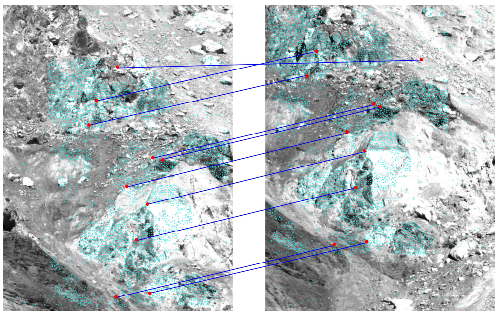

Figure 2. 2D matching results using Efficient-LoFTR (Wang et al., 2024). Tiled RGB images are shown in grayscale for better visualization. Initial matched pixels are highlighted in cyan color, with 10 randomly selected correspondence lines visualized in blue color. These initial correspondences may include errors, thus further refinement needs to be implemented.

2D match establishment. Once the images from source and target epochs are tiled, we perform matching between each pair of tiled images using the Efficient-LoFTR, as illustrated in Fig. 2. This process yields 2D pixel correspondences between the tiled images, with pixel coordinates determined based on the tiled image dimensions. These pixel coordinates are then converted to the original image coordinates by tiled the position of the upper-left corner of each tiled image. The pixel coordinates between each pair of tiled images are collected to form the final set of 2D correspondences, denoted as $\mathbf{C}^{\mathrm{2D}} = \{(\mathbf{u}^{\mathrm{s}}, \mathbf{v}^{\mathrm{s}}, \mathbf{u}^{\mathrm{t}}, \mathbf{v}^{\mathrm{t}})\}$, where $(\mathbf{u}^{\mathrm{s}}, \mathbf{v}^{\mathrm{s}})$ and $(\mathbf{u}^{\mathrm{t}}, \mathbf{v}^{\mathrm{t}})$ represent the matched pixel coordinates in the source and target images, respectively.

2.2 3D match establishment

Point cloud to image projection. Once the image matching on RGB images is complete, the next step is to establish 3D point correspondences from the 2D pixel correspondences. However, not all 2D pixels have corresponding 3D points due to different sensor setup and resolution mismatches between the scans and the images. To address this, we first project the 3D points of both epochs onto 2D pixels and then find closest 2D pixel correspondences for these projected pixels. The projection is performed using the provided camera intrinsic and extrinsic parameters, as depicted in Eq. (1), which are often available from scanner-specific software during data pre-processing.

$$\mathbf{p} = \mathbf{K} \cdot \mathbf{M} \cdot \mathbf{P}, \tag{1}$$

where $\mathbf{p} = [\mathbf{u}_p, \mathbf{v}_p, \mathbf{1}]^{\mathrm{T}}$ and $\mathbf{P} = [\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{1}]^{\mathrm{T}}$ denote homogeneous image coordinates and homogeneous point coordinates, respectively. Here, $\mathbf{K}$ and $\mathbf{M}$ denote 3×3 camera intrinsic and 3×4 camera extrinsic, respectively. The projected source and target pixel coordinates are denoted as $(\mathbf{u}_p^{\mathrm{s}}, \mathbf{v}_p^{\mathrm{s}})$ and $(\mathbf{u}_p^{\mathrm{t}}, \mathbf{v}_p^{\mathrm{t}})$, respectively.

Lift 2D matches to 3D matches. There may exist misalignment between the TLS scanner and its built-in cameras, as investigated in Wang et al. (2023). To accommodate this, for each projected 2D pixel from the source point, we find its nearest neighbor 2D match established in Sec. 2.1 within its k-pixel radius. If a match is found, the corresponding target 2D pixel is used to determine the corresponding 3D point, if one exists. We repeat this process for all projected pixels to establish 3D correspondences. Finally, we filter these correspondences based on a predefined maximum displacement threshold. We denote these 3D point correspondences as $\mathbf{C}^{\mathrm{3D}}$:

$$\mathbf{C}^{\mathrm{3D}} = \{(\mathbf{X}^{\mathrm{s}}, \mathbf{Y}^{\mathrm{s}}, \mathbf{Z}^{\mathrm{s}}, \mathbf{X}^{\mathrm{t}}, \mathbf{Y}^{\mathrm{t}}, \mathbf{Z}^{\mathrm{t}})\}. \tag{2}$$

2.3 3D match refinement

Patch clustering. Inspired by the well-established as-rigid-as-possible assumption (Sorkine and Alexa, 2007), we assume that the movement within a small area can be approximated as rigid. This assumption is practical in landslides, where small stones likely move always as rigid bodies. We define such a small area as a single patch, which can be identified by clustering (or segmentation) algorithms.

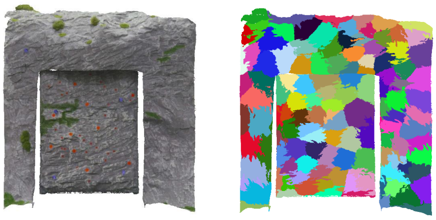

Figure 3. The clustering result on the Rockfall Simulator dataset. *Left*: the source point cloud; *Right*: the clustering result of the source point cloud, with different clusters color-coded for visualization. The chosen clustering algorithm well preserves object boundaries under appropriate settings.

Many clustering algorithms have been proposed in previous studies. HDBSCAN (Campello et al., 2013), for example, is often used for background removal, *e.g.*, ground in scene flow estimation for autonomous driving (Lin and Caesar, 2024). However, in our case, the primary goal of clustering is to well preserve the boundaries of objects, *e.g.*, stones. Therefore, we utilize a supervoxel segmentation algorithm (Lin et al., 2018), which has been verified to effectively preserve object boundaries in LiDAR point clouds, as shown in Fig. 3.

Patch match refinement. Based on the clustering, 3D points within each patch are assumed to undergo rigid deformation. For each patch, we first estimate an initial transformation using point correspondences within current patch through the Kabsch algorithm (Kabsch, 1976). This initial transformation is then refined through a point-to-point ICP (Besl and McKay, 1992), and we denote the refined transformation as $\mathcal{T}$. Once refined, this transformation is applied to all matched source points within the patch to form the local displacement vectors, as depicted in Eq. (3). Finally, the local displacement vectors from all patches are collected to generate the final DVFs, denoted as $\hat{\mathbf{V}}^{\mathrm{3D}} = \cup_{i=1}^{N} \hat{\mathbf{V}}_i^{\mathrm{3D}}$, where N represents the number of patches.

$$\hat{\mathbf{V}}_i^{\mathrm{3D}} = \{\mathcal{T}(\mathbf{X}_i^{\mathrm{s}}, \mathbf{Y}_i^{\mathrm{s}}, \mathbf{Z}_i^{\mathrm{s}}) - (\mathbf{X}_i^{\mathrm{s}}, \mathbf{Y}_i^{\mathrm{s}}, \mathbf{Z}_i^{\mathrm{s}})\}, \tag{3}$$

3. Experiments

To evaluate the proposed method, we conduct experiments on datasets collected from both a physically emulated rockfall event and a real-world landslide. We describe the datasets in Sec. 3.1 and illustrate preliminary observations in Sec. 3.2. In Sec. 3.3, we detail a manual process to generate dense reference DVFs that enable performance evaluation. In Sec. 3.4, we illustrate the baseline methods selected for comparison.

3.1 Datasets

Rockfall Simulator. The Rockfall Simulator is a computer-controlled mechanical apparatus whose central part can be moved vertically and rotated around a horizontal axis (Gojcic et al., 2020), as illustrated in Fig. 4. The surfaces of the simulator are textured to resemble rocks while the controlled changes in geometry allow for acquisition of data that emulate measurements of rockfall events with known ground truth. We use the simulator to perform vertical translations with a magnitude of approximately 0.035 m. To capture the movement, we employ a Leica RTC360 scanner to obtain scans before and after the movement, achieving an average spatial resolution of around 0.0006 m at a distance of approximately 1.7 m. Associated RGB images are captured using the built-in cameras with a resolution of 5120 x 5120 pixels . They provide a ground sampling distance (GSD) of around 0.0006 m/pixel. Additionally, four mini prisms are installed in the moved part to provide further evaluation. We measured the 3D coordinates of these prisms using a Leica TS60 total station.

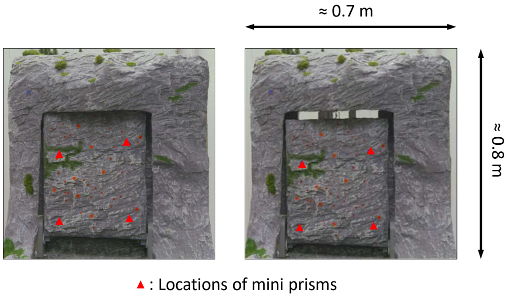

Figure 4. Status of the Rockfall Simulator before (*left*) and after (*right*) the movement.

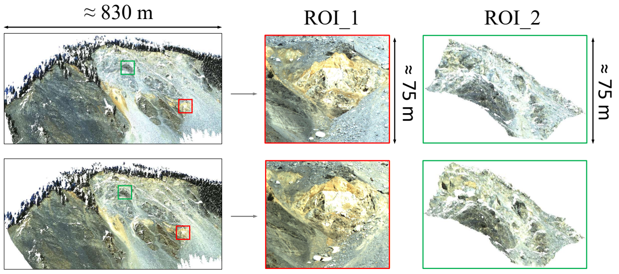

Figure 5. Images of the Brienz TLS dataset and two ROIs at two epochs. *Upper*: source epoch (Feb 2020); *Lower*: target epoch (Nov 2020). The selected areas for analysis, ROI_1 and ROI_2, are highlighted in red and green, respectively.

Brienz TLS. The mountain, located to the north of the Brienz village in the Albula valley in Graubünden, Switzerland, is an area that features unstable rock and landslide displacement activity. The landslide area moves with a rate of up to several meters per year (Krähenbühl and Nänni, 2017; Häusler and Fäh, 2018). The Brienz TLS dataset (Kenner et al., 2022) captures this motion using a TLS scanner of type Riegl VZ-6000, with an average spatial resolution of 0.08 m at a distance of approximately 1.5 km. We analyze scans taken at measurement epochs in February 2020 and November 2020. Each scan comes with 80 RGB images captured by built-in cameras, each of them with a resolution of 2560 x 1920 pixels and a GSD of approximately 0.05 m/pixel. Within the whole area, we select two region-of-interests (ROIs) for analysis, with each ROI capturing an area of 75 m x 75 m, as shown in Fig. 5. From these two ROIs, four areas—A, B, C, and D—are later selected for more detailed analysis due to their varying deformation characteristics (*cf.* Fig. 9).

Data preprocessing. Before conducting deformation analysis, the point clouds from the two different epochs are aligned to a common reference system to ensure that the subsequent analysis primarily reflects actual deformation. The registration was performed using scanner-specific software. For the Rockfall Simulator dataset, we used Cyclone Register 360 Plus to preprocess and register the scans from two different epochs. The registration RMSE is 0.001 m, which is significantly smaller than the actual magnitude of the motion (around 0.035 m). For the Brienz TLS dataset, we employed RiSCAN PRO for preprocessing and registration, with a registration RMSE of 0.047 m. The camera intrinsic and extrinsic parameters used to associate scans with RGB images were obtained from the respective scanner software.

3.2 Preliminary observations

Feature richness. To assess the richness of geometric and radiometric (RGB) features in the selected areas of the Brienz TLS dataset, we first compute the geometric variation $\mathcal{V}_\lambda$ (Weinmann et al., 2013), as depicted in Eq. (4). The local geometric feature richness for each point is then quantified by computing the standard deviation $\sigma_{\mathcal{V}_\lambda}$ within a local neighborhood defined by a 0.5 m search radius. With

$$\mathcal{V}_\lambda = \frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3}, \tag{4}$$

and λ_1, λ_2, λ_3 representing the eigenvalues, sorted such that $\lambda_1 \geq \lambda_2 \geq \lambda_3$, maximum values of 1/3 for $\mathcal{V}_\lambda$ indicate geometric variability in all three dimensions while values around 0 imply planar, linear, or otherwise geometrically degenerated structure.

Similarly, we compute the standard deviation of the grayscale intensity values derived from the RGB values to represent local radiometric feature richness, as these values provide a measure of color texture. Both geometric variation and grayscale intensity values are normalized to the range [0, 1], allowing for a direct comparison of feature richness across the selected areas.

In the Brienz TLS dataset, areas A and C (*cf.* Fig. 9) exhibit greater richness in geometric features compared to radiometric features, while in areas B and D, the opposite is true.

3.3 Manually generated reference DVFs

In real-world, ground truth data are rarely available; thus we use a procedure to generate manual reference data for the performance assessment. The procedure follows three key steps: First, we select several areas and subsequently estimate for each area an initial rigid transformation between the point clouds of the two epochs (*i.e.*, before and after the movement or deformation). This initial estimation is obtained by employing the Kab-

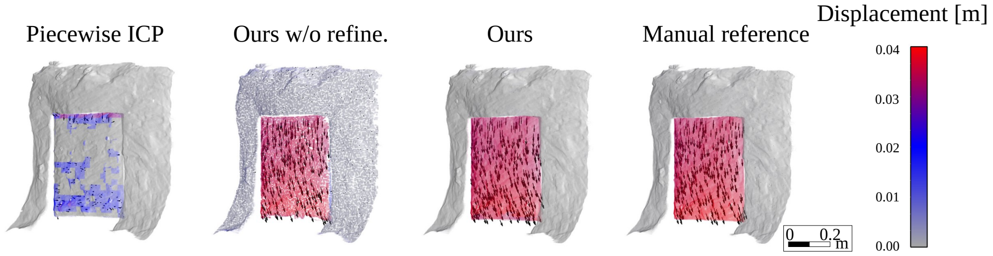

Figure 6. DVFs results on the Rockfall Simulator dataset. The color bar represents the displacement magnitude. The length of the 3D displacement vectors is proportional to the displacement magnitudes, as demonstrated by the equal scaling applied to all vectors. Only 0.05% of all estimated vectors are visualized for better readability.

sch algorithm (Kabsch, 1976) using coordinates of four manually picked points. Finally, the transformation of each area is refined using point-to-point ICP (Chen and Medioni, 1992). The ICP point correspondences are treated as the manually generated reference DVFs for the respective areas.

For the Rockfall Simulator dataset, we partition it into two areas corresponding to the moved part and the stable part. Four points near the mini prisms are picked for the initial transformation estimation of the moved part.No such calculation is needed for the stable part because it does not move or deform. The RMSE on the two parts of ICP point correspondences is about 0.003 m and 0.001 m, respectively. For the Brienz TLS dataset, four areas are selected (*cf.* Fig. 9), with four manually picked points used for initial transformation estimation within each selected area, yielding an RMSE of around 0.30 m. The final RMSE of the ICP point correspondences is 0.05 m.

3.4 Baseline methods.

We compare our method against three other approaches that also generate 3D DVFs: Piecewise ICP (Friedli and Wieser, 2016), F2S3 (Gojcic et al., 2020), and our method w/o refinement. Piecewise ICP serves as an early solution for producing 3D DVFs, while F2S3 is, to our best knowledge, so far the only method that uses deep learning to estimate 3D DVFs based on point-to-point correspondences. Our method w/o refinement presents the results primarily from the image matching algorithm we used, without applying the refinement step (*cf.* Sec. 2.3). For Piecewise ICP, we set the minimum octree cell size to 0.05 m and 5 m for the Rockfall Simulator and Brienz TLS datasets, respectively, with a minimum of 20 points per octree cell. For F2S3 and our method, the voxel sizes are set to 0.003 m and 0.100 m for the Rockfall Simulator and Brienz TLS datasets, respectively.

4. Results and Discussion

4.1 Results on Rockfall Simulator

Comparison of different DVFs. We first present the DVFs obtained from the different methods, along with the manual reference in Fig. 6. Our approach yields results that most closely match the GT in both the moved part (blue areas) and the stable part (gray areas). Piecewise ICP estimates only very small movements in the moved part. Our method w/o refinement (Ours w/o refine.) produces relatively sparse DVFs and exhibits inaccuracies at the borders (*e.g.*, the right boundary of the moved part), due to the absence of further refinement. However, our refinement component mitigates this issue by correctly clustering the areas to the left side and applying a robust transformation, resulting in a more accurate DVF.

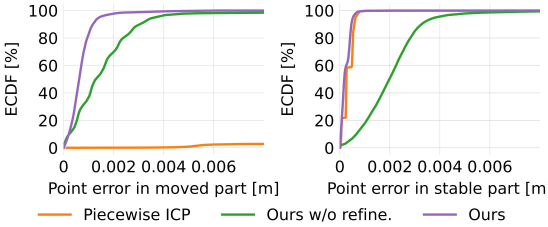

Figure 7. ECDF results on the Rockfall Simulator dataset. Only intersection points among different methods are considered to ensure a fair comparison.

ECDF results. We further compare the empirical cumulative distribution function (ECDF) of point errors, measured by the norm of DVF discrepancies against the manual reference, as shown in Fig. 7. In the moved part, our method shows strong performance, though our method w/o refinement exhibits some outliers. Piecewise ICP, on the other hand, demonstrates a significant discrepancy between the estimated displacement and the manual reference. This discrepancy arises because the physically emulated movement occurs primarily parallel to the surface, which is relatively planar. In such cases, Piecewise ICP, which relies on centroid distance, can significantly underestimate the deformation. In the stable part, both our method and Piecewise ICP perform well, whereas the results without refinement show errors that can reach up to 0.010 m.

Comparison with TS data. As previously mentioned, we observe the coordinates of four mini prisms before and after the movement using a total station. Since these coordinates are obtained in the total station’s own coordinate system, we compute only the mean movement magnitude. A search radius of 0.05 m is applied to crop areas around the mini prisms. Our method yields results closest to the total station data, with the mean displacement magnitude showing discrepancies of less than 0.004 m compared to the total station observations.

4.2 Results on Brienz TLS

DVF results. We present the DVFs generated by F2S3 and our method for two ROIs of the Brienz TLS dataset in Fig. 8. Overall, both methods produce dense DVFs suitable for deformation analysis, with our method generating denser DVFs of 664 k and 150 k points compared to F2S3’s 118 k and 111 k points in ROI_1 and ROI_2, respectively. Both methods capture the main deformation patterns within the individual regions, *e.g.*, the more uniform and smaller displacement magnitude in ROI_1 compared to ROI_2. However, discrepancies between the two methods are evident, and we illustrate these differences at the intersection points (*i.e.*, points for which both methods generate DVFs) of the two methods in Fig. 9. The average discrepancies are 0.32 m and 0.36 m for ROI_1 and ROI_2, re-

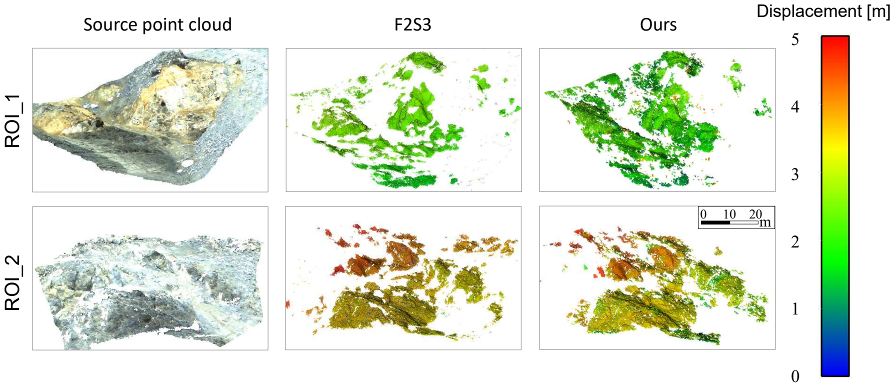

Figure 8. DVF results on two ROIs from the Brienz TLS dataset. The color bar represents the displacement magnitude.

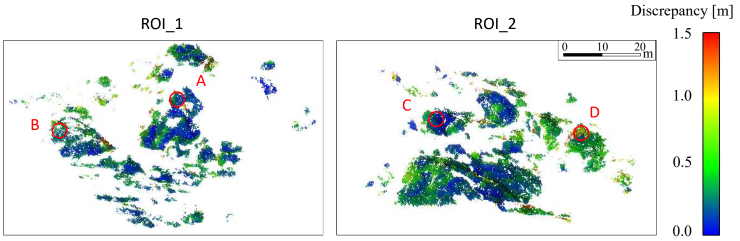

Figure 9. DVF discrepancies between the intersection points of F2S3 and our method on two ROIs from the Brienz TLS dataset. A narrow range of the color bar is applied to better visualize the discrepancies. The red circles highlight four areas selected for further analysis.

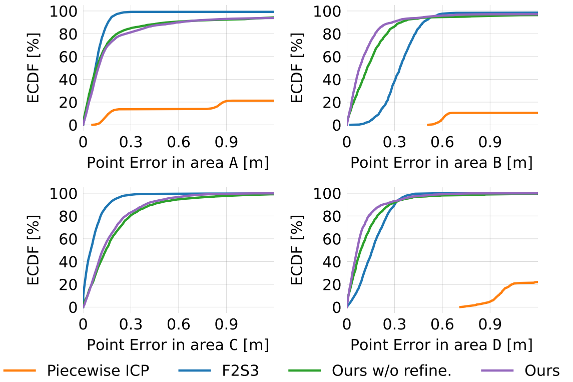

Figure 10. ECDFs for four selected areas of the Brienz TLS dataset. Only intersection points among different methods are considered to ensure a fair comparison.

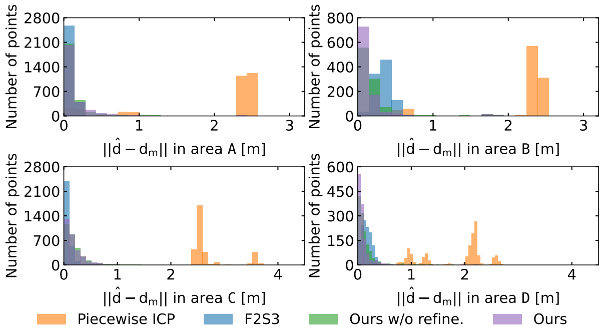

Figure 11. Histograms of displacement magnitude differences for four selected areas of the Brienz TLS dataset. $\hat{d}$ and d_m represent the displacement magnitudes derived from our method and the manual reference, respectively. Only intersection points among different methods are considered to ensure a fair comparison.

spectively. For further analysis, we focus on four selected areas (*i.e.*, areas A, B, C, and D) that exhibit varying degrees of discrepancy. A search radius of 5 m is applied to crop these areas for comparison.

ECDF results. The ECDFs for the four selected areas are shown in Fig. 10. In areas A and C, F2S3 outperforms our method, as indicated by the faster convergence of its curves compared to ours. Conversely, in areas B and D, our method performs better, with F2S3 showing some offset when compared to the manual reference. Piecewise ICP consistently underestimates deformation, likely due to the high dynamic activity in this dataset, which makes its uniform octree cell size less effective at capturing the actual deformations. These observations are further corroborated by the histogram plots in Fig. 11.

Mean magnitude results. Based on the DVFs in the four selected areas, we further compute the mean displacement magnitudes. Compared to the manual reference data, Piecewise ICP consistently underestimates the mean magnitude. In areas A and C, F2S3 yields estimates closer to the manual reference, showing an average discrepancy of 0.07 m, whereas our method has a larger discrepancy of 0.19 m. Conversely, in areas B and D, our method performs better, with an average discrepancy of 0.02 m, compared to F2S3’s 0.26 m.

3D Displacement vector results. We also provide the estimated 3D displacement vectors of our method for ROI_1 in Fig. 12. For better readability, 0.05% of all estimated vectors are randomly selected and thus represent the overall displacement pattern in ROI_1. Regions where our method struggles to estimate 3D displacement vectors are characterized by no or few vectors. The estimated 3D displacement vectors show a high degree of consistency with our previous results in Fig. 8.

4.3 Further discussion

Choice of the search radius size for local rigid evaluation. To assess the impact of radius size on local rigid deformation

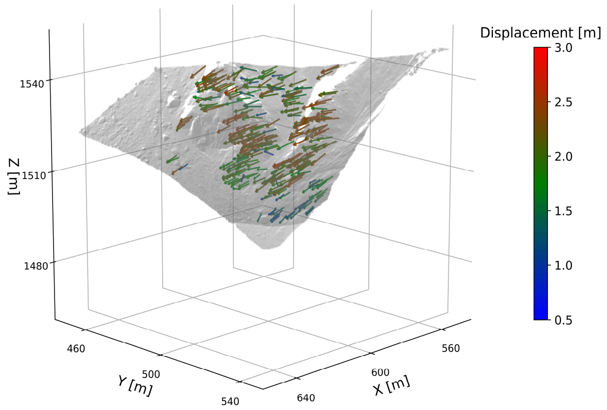

Figure 12. 3D displacement vectors for ROI_1 of the Brienz TLS dataset. Only 0.05% of all estimated vectors are visualized for better readability.

evaluation, we perform tests with the radius ranging in the interval [3, 9] m. The results presented in Tab. 1 indicate that discrepancies in the mean displacement magnitudes remain below 0.07 m when the radius is adjusted by 2 m. These discrepancies are significantly smaller than those observed between different methods. The large discrepancies in the standard deviations are primarily due to the presence of outliers. For the sake of simplicity, we choose constant radius values of 0.05 m and 5 m for the Rockfall Simulator and Brienz TLS datasets, respectively, during the evaluation.

Table 1. Choices of the local rigid radius size (unit: m). The values for different areas represent the discrepancies in the mean and standard deviation of displacement magnitudes of our method compared to the manual reference [1].

Radius size	Area A	Area B	Area C	Areas D
3	-0.106 ± 0.145	0.027 ± 0.170	-0.171 ± 0.137	0.093 ± 0.079
5	-0.082 ± 0.355	0.029 ± 0.176	-0.192 ± 0.299	-0.024 ± 0.296
7	-0.129 ± 0.366	-0.003 ± 0.182	-0.265 ± 0.365	-0.024 ± 0.280
9	-0.166 ± 0.371	-0.025 ± 0.187	-0.289 ± 0.359	-0.035 ± 0.237

Run-time comparison. We compare the run-time of F2S3 and our method in Tab. 2. For a fair comparison, both methods are evaluated on ROI_1 of Brienz dataset using a single GeForce RTX 3090 Ti with an AMD Ryzen 7 5800X 8-Core Processor. Our method demonstrates a notable time advantage in matching, as it operates in 2D space, whereas F2S3 performs computations in 3D space. Specifically, our method completes the matching and refinement process in 31 seconds, while F2S3 takes 2.8 times longer. This efficiency gain could become significant when applied to real-time monitoring applications.

Table 2. Runtime comparison of F2S3 and our method (unit: s). Refine. and feat. denote refinement and feature, respectively.

Method	2D matching	Refine.	3D feat. extraction	3D matching	Total ↓
F2S3	-	-	41	45	86
Ours	18	13	-	-	31

Limitations. Our method depends on accurate image-to-point cloud registration, *e.g.*, using scan-specific software when captured in the same epoch. While resilient to moderate illumination changes (*cf.* Figs. 5 and 8), it requires sufficient lighting and may fail in fully shaded or extremely low-light conditions (*e.g.*, dark nights). Strong illumination shifts (*e.g.*, bright sunlight vs. overcast, day vs. full-moon night) and dynamic surface changes (*e.g.*, vegetation, wetness) require further study. In regions with complex geometric structures but low color variation, the geometry-based method (F2S3) outperforms our approach (*cf.* areas A and C in Fig. 10). Conversely, in areas with planar surfaces and strong color variation, our method is superior to F2S3. This observation is further supported by our feature richness analysis (*cf.* section 3.2). Additionally, both our method and F2S3 fail to generate valid estimates in certain areas (*i.e.*, white regions in Fig. 8), primarily due to the presence of debris, where neither geometric nor radiometric features provide sufficient support for accurate correspondence establishment.

5. Conclusion

In this paper, we propose an approach that leverages RGB images to generate dense 3D DVFs for LiDAR-based landslide monitoring. Previously, these images, captured by the built-in cameras of TLS scanners, have been used primarily for visualization purposes. Our method, however, utilizes a deep learning-based image matching algorithm to produce DVFs with accuracy comparable to 3D geometry-based methods such as F2S3. Experimental results demonstrate that while F2S3 outperforms our method in areas with rich geometric features, our approach excels in areas where radiometric (RGB color) features are more prominent. The DVFs generated by our method can cover some areas where F2S3 fails to produce DVFs. Furthermore, our method shows superior computational efficiency, making it more suitable for near-real-time monitoring scenarios. By leveraging RGB radiometric features, our method expands the current algorithm toolbox, which typically focuses on geometric information. The complementary strengths of our 2D RGB image-based approach and existing 3D geometry-based methods suggest that combining both will enhance coverage and/or accuracy beyond what either method achieves independently.

Acknowledgments

Robert Kenner (Institute for Snow and Avalanche Research) has provided the Brienz TLS dataset. Helena Laasch has shared the code for Piecewise ICP. The first author was supported by the China Scholarship Council (CSC).

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

We used ChatGPT in order to improve readability and language. We reviewed and edited the resulting text and take full responsibility for the content of the entire publication.

References

Albanwan, H., Qin, R., Liu, J.-K., 2024. Remote Sensing-Based 3D Assessment of Landslides: A Review of the Data, Methods, and Applications. *Remote Sensing*, 16(3).

Bergevin, R., Soucy, M., Gagnon, H., et al., 1996. Towards a general multi-view registration technique. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(5), 540-547.

Besl, P., McKay, N. D., 1992. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), 239-256.

Campello, R. J. G. B., Moulavi, D., Sander, J., 2013. Density-based clustering based on hierarchical density estimates. *Pacific-Asia Conference on Knowledge Discovery and Data Mining*.

[1] Only the smallest 95.5% of point errors are used in the computation to exclude some large outliers.

Chen, Y., Medioni, G., 1992. Object modelling by registration of multiple range images. *Image and Vision Computing*, 10(3), 145-155.

Cignoni, P., Rocchini, C., Scopigno, R., 1998. Metro: Measuring Error on Simplified Surfaces. *Computer Graphics Forum*, 17(2), 167-174.

Dusmanu, M., Rocco, I., Pajdla, T., et al., 2019. D2-Net: A trainable cnn for joint detection and description of local features. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Fey, C., Rutzinger, M., Wichmann, V., et al., 2015. Deriving 3D displacement vectors from multi-temporal airborne laser scanning data for landslide activity analyses. *GIScience & Remote Sensing*, 52(4), 437–461.

Friedli, E., Wieser, A., 2016. Identification of stable surfaces within point clouds for areal deformation monitoring. *Proc. of 3rd Joint International Symposium on Deformation Monitoring (JISDM)*.

Girardeau-Montaut, D., Roux, M., Marc, R., et al., 2005. Change detection on points cloud data acquired with a ground laser scanner. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(3), W19.

Gojcic, Z., Schmid, L., Wieser, A., 2021. Dense 3D displacement vector fields for point cloud-based landslide monitoring. *Landslides*, 18, 3821–3832.

Gojcic, Z., Zhou, C., Wieser, A., 2020. F2S3: Robustified determination of 3D displacement vector fields using deep learning. *Journal of Applied Geodesy*, 14(2), 177–189.

Häusler, M., Fäh, D., 2018. Monitoring the slope instability at brienz/brinzauls using ambient vibrations and earthquake recordings. *Swiss geoscience meeting*, 225–226.

Holst, C., Janßen, J., Schmitz, B., et al., 2021. Increasing spatio-temporal resolution for monitoring alpine solifluction using terrestrial laser scanners and 3d vector fields. *Remote Sensing*, 13(6), 1192.

Ioli, F., Barbieri, F., Gaspari, F., et al., 2023. ICEPY4D: A Python toolkit for advanced multi-epoch glacier monitoring with deep-learning photogrammetry. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVIII-1/W2-2023, 1037–1044.

Jaboyedoff, M., Oppikofer, T., Abellán, A., et al., 2012. Use of LIDAR in landslide investigations: a review. *Natural hazards*, 61, 5–28.

James, M. R., Robson, S., Smith, M. W., 2017. 3-D uncertainty-based topographic change detection with structure-from-motion photogrammetry: precision maps for ground control and directly georeferenced surveys. *Earth Surface Processes and Landforms*, 42(12), 1769-1788.

Kabsch, W., 1976. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 32(5), 922–923.

Kenner, R., Gischig, V., Gojcic, Z., et al., 2022. The potential of point clouds for the analysis of rock kinematics in large slope instabilities: examples from the Swiss Alps: Brinzauls, Pizzo Cengalo and Spitze Stei. *Landslides*, 19(6), 1357–1377.

Krähenbühl, R., Nänni, C., 2017. Ist das dorf Brienz-Brinzauls bergsturz gefährdet. *Swiss Bull. Angew. Geol*, 22(2), 33–47.

Lague, D., Brodu, N., Leroux, J., 2013. Accurate 3D comparison of complex topography with terrestrial laser scanner: Application to the Rangitikei canyon (N-Z). *ISPRS Journal of Photogrammetry and Remote Sensing*, 82, 10-26.

Lin, Y., Caesar, H., 2024. ICP-Flow: Lidar scene flow estimation with icp. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Lin, Y., Wang, C., Zhai, D., et al., 2018. Toward better boundary preserved supervoxel segmentation for 3D point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 143, 39-47.

Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60, 91-110.

Mukupa, W., Roberts, G. W., Hancock, C. M., Al-Manasir, K., 2017. A review of the use of terrestrial laser scanning application for change detection and deformation monitoring of structures. *Survey Review*, 49(353), 99–116.

Sarlin, P.-E., DeTone, D., Malisiewicz, T., et al., 2020. SuperGlue: Learning feature matching with graph neural networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Sorkine, O., Alexa, M., 2007. As-rigid-as-possible surface modeling. *Proceedings of the Fifth Eurographics Symposium on Geometry Processing*, 109–116.

Sun, J., Shen, Z., Wang, Y., et al., 2021. LoFTR: Detector-free local feature matching with transformers. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Teo, T.-A., Fu, Y.-J., Li, K.-W., et al., 2023. Comparison between image-and surface-derived displacement fields for landslide monitoring using an unmanned aerial vehicle. *International Journal of Applied Earth Observation and Geoinformation*, 116, 103164.

Teza, G., Galgaro, A., Zaltron, N., GENEVOIS, R., 2007. Terrestrial laser scanner to detect landslide displacement fields: a new approach. *International Journal of Remote Sensing*, 28(16), 3425–3446.

Vaswani, A., Shazeer, N., Parmar, N., et al., 2017. Attention is all you need. *Proceedings of the Conference on Neural Information Processing Systems*.

Wang, Y., He, X., Peng, S., et al., 2024. Efficient LoFTR: Semi-dense local feature matching with sparse-like speed. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Wang, Z., Varga, M., Medić, T., et al., 2023. Assessing the alignment between geometry and colors in TLS colored point cloud. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-1/W1-2023, 597–604.

Weinmann, M., Jutzi, B., Mallet, C., 2013. Feature relevance assessment for the semantic interpretation of 3D point cloud data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2, 313–318.

Williams, J. G., Anders, K., Winiwarter, L., et al., 2021. Multi-directional change detection between point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 172, 95-113.

Wujanz, D., Avian, M., Krueger, D., et al., 2018. Identification of stable areas in unreferenced laser scans for automated geomorphometric monitoring. *Earth Surface Dynamics*, 6(2), 303–317.

Yang, Y., Schwieger, V., 2023. Patch-based M3C2: Towards lower-uncertainty and higher-resolution deformation analysis of 3D point clouds. *International Journal of Applied Earth Observation and Geoinformation*, 125, 103535.

Zahs, V., Winiwarter, L., Anders, K., et al., 2022. Correspondence-driven plane-based M3C2 for lower uncertainty in 3D topographic change quantification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 183, 541-559.