

Markerless Ad-hoc Calibration of a Hyperspectral Camera and a 3D Laser Scanner

Felix Igelbrink¹, Thomas Wiemann¹, Sebastian Pütz¹,
Joachim Hertzberg^{1,2}

¹ Knowledge Based Systems Group, Osnabrück University,
Wachsbleiche. 27, 49090 Osnabrück, Germany
`figelbrink|twiemann|spuetz@uni-osnabrueck.de`

² DFKI Robotics Innovation Center, Osnabrück Branch,
Albert-Einstein-Str. 1, 49076 Osnabrück, Germany,
`joachim.hertzberg@dfki.de`

Abstract. Integrating 3D data with hyperspectral images opens up novel approaches for several robotic tasks. To that end, we register hyperspectral panoramas to cylindrically projected laser scans. With our approach, the required calibration can be done on board a mobile robot without the need of external markers using Mutual Information. Qualitative results show the robustness of the presented approach, and an application example demonstrates possible future applications for hyperspectral point clouds.

Keywords: mobile robot, 3D laser scanning, hyperspectral data, image registration

1 Introduction

These days terrestrial laser scanners are able to acquire billions of points in a single scan. Laser scanning is used for building 3D environment models [?] in many applications. In robotics, these models are used for many purposes, e.g., localization, mapping, manipulation, and reasoning to safely interact with the environment.

Integrating spectral information into such models is highly desirable. Hyperspectral cameras split up the spectrum of the incoming light into buckets of wavelength intervals to capture the intensity distribution. These cameras are usually built as line cameras, where one dimension of a frame refers to the spatial image component and the other to the spectral distribution.

Mapping hyperspectral information to high resolution 3D point clouds makes the grounding of material characteristics possible. Currently, hyperspectral images are mainly collected in remote sensing from airplanes, drones or satellites. This limits the level of detail to a rough estimation of the location of detected materials due to the limited resolution. The resulting 2D maps have been used

for identifying forests, fields, rivers, residential areas, and other structures of interest. In robotics, such segmentation and classification could be used to complement a plethora of applications, e.g., navigability estimation, semantic mapping and localization. In particular for navigation for and exploration purposes, it is important not only to know which materials are present in a given environment, but also precisely *where* they are located.

To combine 3D spatial information from terrestrial laser scanners and hyperspectral images, the two sensors need to be calibrated with respect to each other. In this paper, we present a method to automatically calibrate a hyperspectral image of a line camera to a high-resolution point cloud without using external calibration patterns. It is fully integrated into a mobile robot, controlled by the Robot Operating System (ROS). We present and evaluate the used calibration technique and the present preliminary results, e.g., using the well-known NDVI (Normalized Difference Vegetation Index) to segment pathways in a hyperspectral point cloud for navigation. Furthermore, we extended and enhanced the state-of-the-art calibration techniques to satisfy the time constraints, which need to be observed in a robotics setting.

2 Related Work

Fusing point clouds and color data from RGB camera images is state of the art in the fields of robotics and photogrammetry. The most common solution for the required extrinsic calibration is to use specific calibration patterns with detectable common feature points, e.g., a checkerboard. These correspondences are used to compute the extrinsic parameters by linear transformation and Levenberg-Marquardt optimization [11].

To align point clouds and RGB or hyperspectral image data with such a model, multiple methods have been presented: In [3], a calibrated RGB camera on top of a laser scanner is used to create an RGB-colored point cloud, which is registered to the hyperspectral image using SIFT features and a piecewise linear transformation. However, this approach requires to mount an additional camera, which may not be possible on mobile robots due to space constraints.

In [1] this is solved by using manually placed high-reflectance markers in the scene that can be detected in both the hyperspectral image and the reflectance channel of the laser scanner. These markers and additional automatically found correspondence points are used for the image matching. The use-case was further simplified by capturing only images with small panorama angles. However, for in-place recalibration on 360° images, placing markers is no option. Marker-less solutions [7, 12] rely solely on automatically detected correspondence points using SIFT and SURF features. However, as the image domains of reflectance data and the hyperspectral imaging are different, we found that establishing stable correspondences in both images is strongly dependent on the environment and on lighting conditions. An approach for registering image data from different domains without requiring the computation of special features is Mutual Infor-

mation (MI). The MI metric is derived from the idea of Shannon entropy [6]; it is state of the art for registering of *CT* and *MRI* images.

In [9], Normalized Mutual Information (NMI) is applied to the registration of laser scans and hyperspectral images. It requires no key-point search or manual correspondences and can therefore be used in our context. Therefore, we deem this approach as a suitable starting point for our own registration problem.

3 Generating Hyperspectral Panoramas

We installed our hyperspectral scanning system on the mobile robot *Pluto*, based on a VolksBot XT platform. It features a *Riegl VZ400i* high resolution 3D laser scanner and a *Resonon Pika L* hyperspectral line camera, mounted on top of the rotating laser scanner. The camera records up to 297 independent spectral channels between 400 nm and 1000 nm, which are accumulated to a hyperspectral panorama image, while the scanner is rotating and recording the point cloud data.

Unfortunately, the camera does not deliver time stamps and guarantees no continuous frame rate. Hence, synchronization of the arriving frames with the angular information from the laser scanner, which delivers time-stamped data, is required to produce consistent panoramas. During a scan, all frames from the hyperspectral camera are buffered and associated with the current scan angle. After a scan is completed, the panorama is built up from the buffered line images. To produce a coherent aspect ratio, the resulting panorama is scaled according to the angular resolution of the laser scan. Missing frames are automatically corrected by interpolating adjacent lines, to produce a consistent hyperspectral panorama image, from which a pseudo-RGB image is computed for the registration. For later use, the full spectral data is saved separately. Fig 2 presents an example for this panorama generation in pseudo-RGB (derived from the appropriate channels).

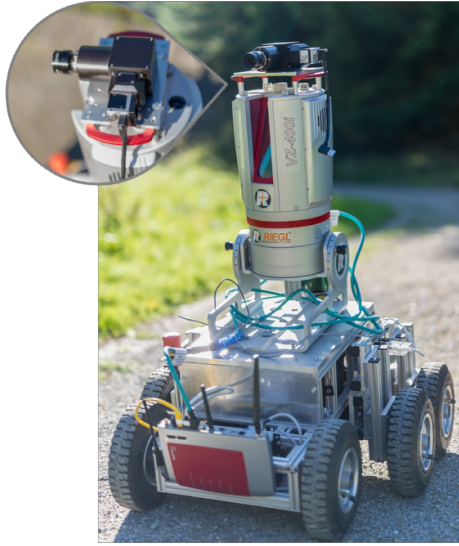


Fig. 1: *Pluto* equipped with a *Riegl VZ400i* terrestrial laser scanner and a *Resonon Pika L* hyperspectral camera.

4 Extrinsic Calibration

The extrinsic calibration of the laser scanner and the hyperspectral panorama is done in 3 successive steps, which will be presented in the following sections.



Fig. 2: Pseudo-RGB section of a panorama generated by the system.

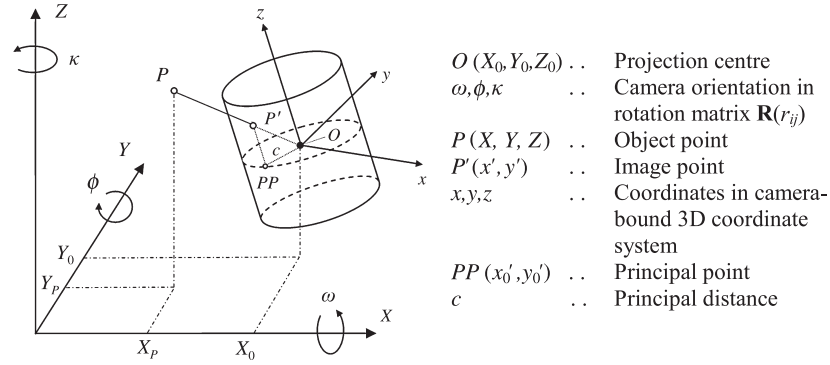


Fig. 3: Cylindric camera model principle [5].

4.1 Cylindric Camera Model

The first step in the extrinsic calibration is to find an appropriate projection of the 3D scan points into a panoramic image plane that can be compared with the hyperspectral panorama. Since the hyperspectral camera is rotating together with the laser scanner, the image geometry cannot be described by a pinhole model and a central perspective projection [1, 2]. Central projection is only valid for the across-track direction, while an angular component needs to be considered in the along-track direction. This is achieved by the cylindric camera model proposed in [5]. It describes the projection of 3D points into a cylindrical image as shown in Fig. 3. For each data point $P(X, Y, Z)$ in the 3D scan, the following steps are performed to compute its pixel coordinates:

1. Transform world coordinates into camera coordinates:

$$p = \mathbf{R}^{-1} (P - O) \quad (1)$$

where O is the camera origin in world coordinates, $\mathbf{R}(\omega, \phi, \kappa)$ is the rotation matrix, and $p(x, y, z)$ is the resulting 3D point in camera coordinates.

2. Project p to a cylinder surface surrounding the camera origin:

$$\begin{aligned} x_i^{pano} &= x_p - c \arctan\left(\frac{-y_i}{x_i}\right) + \Delta_{x^{pano}} \\ y_i^{pano} &= y_p - \frac{cz_i}{\sqrt{x_i^2 + y_i^2}} + \Delta_{y^{pano}} \end{aligned} \quad (2)$$

Here x_p and y_p denote the x and y components of the principal point, c the principal distance and $\Delta_{x^{pano}}$, $\Delta_{y^{pano}}$ are optional correction terms to include compensation for radial distortion of the camera lens and imperfectly aligned axes [5]. For our system, we do not include them into the optimization problem, since we are dealing with range measurements.

3. Finally, convert the obtained 2D coordinates into the pixel coordinates of the resulting image:

$$\begin{aligned} x_i^{image} &= \frac{x_i^{pano} + \min_j x_j}{I_x} & I_x &= \frac{\max_i x_i - \min_i x_i}{R_x - 1} \\ y_i^{image} &= \frac{N}{2} - \frac{y_i^{pano}}{I_y} & I_y &= \frac{\max_i y_i - \min_i y_i}{(R_x \cdot \xi) - 1} \end{aligned}, \text{ where } \quad (3)$$

are the value increments for one image pixel in x and y directions. R_x is the horizontal resolution of the projected image, and ξ is the aspect ratio of the resulting image, which is computed from the projected coordinates x_i , y_i .

The height of the resulting point cloud panorama is usually greater than that of the hyperspectral panorama due to the limited aperture angle of the hyperspectral camera. Therefore, the regions not covered by both sensors are automatically cut from the image after the projection. A section of a panorama image generated from the point cloud using the reflectance channel as intensity values is shown in Fig. 4.

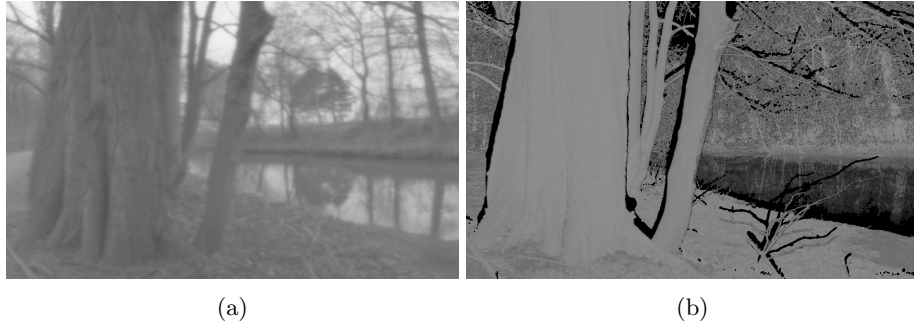


Fig. 4: Comparison between a grayscale section of the hyperspectral panorama (a) and the projected point cloud (b). The black borders of nearby objects in the point cloud are caused by the transformation into the camera coordinate system.

4.2 Mutual Information

After the projected scan image is computed, we compare it with the hyperspectral image using the Mutual Information metric. This metric measures the statistical dependence between two random variables. It is derived from the Shannon entropy [6]. It is especially well suited for multimodal images because it accounts for areas having different intensity values in both images.

Normalized Mutual Information (NMI) [8] is an extension of the regular Mutual Information that attempts to remove dependency of regular MI on the total amount of information contained in both images, as this dependency might cause MI to produce false global maxima where the overlap of both images is small. NMI is computed from the Shannon entropies as follows:

$$NMI(M, N) = \frac{H(N) + H(M)}{H(M, N)}, \quad (4)$$

where

$$\begin{aligned} H(M) &= - \sum_{m \in M} p_M(m) \log \frac{1}{p_M(m)} \\ H(N) &= - \sum_{n \in N} p_N(n) \log \frac{1}{p_N(n)} \\ H(M, N) &= - \sum_{m \in M} \sum_{n \in N} p_{M,N}(m, n) \log \frac{1}{p_{M,N}(m, n)} \end{aligned} \quad (5)$$

are the individual and joint entropies, respectively. M, N are the discrete random variables, p_M, p_N is the probability distribution over M and N , respectively, and $p_{M,N}$ is the joint probability distribution of both variables. For image registration, the probability distributions p_M, p_N , and $p_{M,N}$ can be approximated by a histogram of the intensity values

$$\hat{p}(X = k) = \frac{1}{n} \sum_{i=1}^n \phi_k(X_i), \quad k \in [0, 255], \quad \text{where } \phi_k(x) = \begin{cases} 1 & \text{if } x = k \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Now, the objective function for the optimization problem can be defined as:

$$\hat{\Theta} = \arg \max_{\Theta} NMI(M, N; \Theta) \quad (7)$$

which has its global maximum at the optimal parameters of the camera model.

4.3 Smoothing the Objective Function

The basic algorithm, detailed in the previous sections, produces a nonlinear, non-convex objective function $\hat{\Theta}$ with many local maxima (see Fig. 6). Although this function can be solved on its own already, it requires expensive exhaustive optimization algorithms like, e.g., particle swarm optimization [9] or simulated annealing for robust conversion.

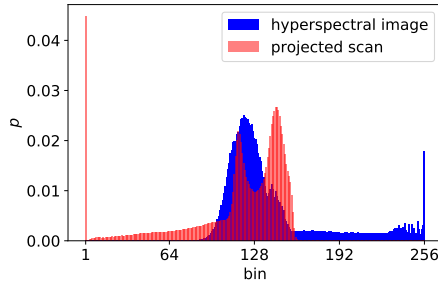


Fig. 5 Histograms of the hyperspectral image and the projected scan with 256 bins. The projected scan uses the reflectances mapped to $[0..255]$.

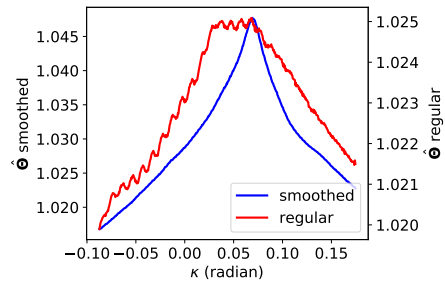


Fig. 6 Example of the resulting objective functions from the regular (unsmoothed) and the smoothed histograms.

To enable fast optimization, the objective function $\hat{\Theta}$ has to be smoothed. Analysis shows that the non-convex behavior is mainly caused by the histogram approximation of the probability distributions, which suffers from a high mean-squared error (MSE). Smoothing $\hat{\Theta}$ aims at reducing the MSE, which can be achieved by various means. A common approach is to smooth the histograms by using a continuous interpolation method, e.g., kernel density estimation (KDE) or B-Splines, for approximating the probability distributions [4, 10]. This smooths the objective function and makes the target function differentiable, enabling to use efficient robust gradient-descent-based optimization algorithms. However, such an interpolation is expensive to compute for high-resolution images like the 360° panoramas in our use case. Additionally, the authors in [4] use multiple image pairs simultaneously to further smooth out the function. This is quite effective, but results in a significantly increased computational load, because the projection of the laser points has to be repeated for each scan in every step.

In order to maintain computational efficiency, we use a much simpler approach for our method. First, we smooth out sharp image edges using a Gaussian filter on the projected images at the borders of objects, which are produced by distant points in the laser scan with potentially very different reflectance values or with no corresponding scan points at all due to the transformation as shown in Fig. 4b. This in turn smooths the distribution differences in the computed histograms between consecutive iterations resulting in a smoother target function.

Additionally, we observed that the resulting histograms from the projected reflectance images tend to be very sparse when using one bin for each possible intensity value. In Fig. 5, the projected scan image mostly uses the lower half of the bins, while the hyperspectral image covers the upper half with many noisy local maxima. These maxima have a significant effect on the entropy values. Reducing the number of bins therefore has a significant smoothing effect on the objective function, while preserving most of the contained information. In

our experiments, we reduced the number of bins to 16. To smooth the function even further, we convolve the resulting histogram with a Gaussian kernel. These steps result in a very smooth objective function with a clear global maximum and significantly reduced local maxima, as shown in the blue plot in Fig. 6, while still being computationally efficient.

The smoothed objective function can be optimized using standard optimization algorithms that do not require the gradient or Hessian, as the model parameters θ are not involved in $\hat{\Theta}$ in a differentiable way, due to the histogram step. We use the Nelder-Mead algorithm with the initial simplex spread over the entire search space. This algorithm converges to the global optimum after around 100 – 200 evaluations of the target function.

When multiple scan/image pairs are available from the same setup, a similar technique as in [4] can be utilized. Although not necessary for convergence, using multiple scans may result in more robust calibration parameters.

5 Experiments

We have implemented our method in Python using the `numpy`³ and `scipy`⁴ libraries. As the projection of the point cloud into an image and the computation of the histograms are by far the most costly parts of the method, we have also implemented a GPU version of these parts using NVIDIA CUDA. This reduces the necessary computation time for one iteration significantly to about 50 ms. The registration of one scan/image pair usually converges within 30 s from the start which is significantly faster than the time to acquire one scan.

To evaluate our method, we acquired a set of outdoor scans using the setup depicted in Fig. 1. We kept the setup fixed, so the model parameters are assumed to be the same for all scans. We constructed the search space from the offset along the z -axis Z_0 from 0.0 m to 0.5 m, the yaw angle κ from -30° to 30° , and the vertical component of the principle point y_p from -0.2 to 0.2 , derived from rough estimations of the setup’s alignment.

All other model parameters are irrelevant for our setup, as the camera is mounted directly on top of the laser scanner; so they are assumed to be 0 (1 for the principle distance c), but are still included in the implemented model to support alternative setups. This initialization limits the search space as well as the runtime and allows for in-field registration on the mobile robot platform. A global registration using no prior knowledge about the parameters is not possible using our method, because our smoothing of the histograms does not remove the nonlinearity of the objective function entirely.

For our experiments, we recorded scan/image pairs at several different outdoor locations, each including both natural and human-made structures. Each scan contains roughly 11 to 14 million points and each hyperspectral panorama has a size of approx. 7500×900 pixels. The number of histogram bins was set to 16 for all scans. All images were down-sampled to a horizontal (x) resolution of

³ <http://www.numpy.org>

⁴ <https://www.scipy.org>

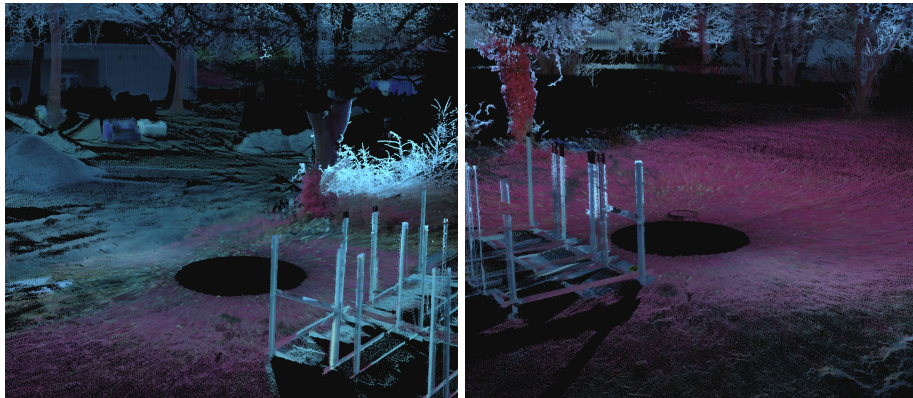


Fig. 7: Calibration results in false colors for several different scans (best viewed in color). The circular black spot represents the scanning position.

5000 pixels as a compromise between computing time and accuracy of the final result. The calibrations were executed on a PC using an Intel i7 4930K CPU as well as a NVIDIA 770GTX GPU for the CUDA Implementation.

In section 5.1, we present qualitative results of our calibration as well as a short analysis of the obtained parameters. We evaluate the run time of our method in section 5.2. Finally, we outline possible applications for our setup by detecting plants and pathways using the hyperspectral point cloud in section 5.3.

5.1 Qualitative Results

To visualize the results of the registration, some scan sections with added hyperspectral data are visualized in Fig. 7. The colors are switched, so that the red colors indicate strong reflectance in the infrared spectral bands to highlight areas with many plants (chlorophyll) and the green and blue color channels were flipped, so that blue areas indicate regions with less plants. The light-blue areas in the image are sky pixels from the panorama, which end up as the colors of tree twigs, because the laser scanner produces high noise in such areas with many small structures. Some mis-registration is visible in both images on the metal structure near the scan position. This is caused by the perspective difference between the laser scanner and the hyperspectral camera, producing images with differing information, especially for close objects. So to obtain good registration results, most larger objects should be a few meters away from the scanner position, as nearby objects are represented by a larger amount of pixels than far-away objects and therefore contribute more to the NMI.

Fig. 8 shows the parameter variation for all scans. The variance of the translation Z_0 is clearly much larger than the other parameters. This is expected [4], as the slight translation between laser scanner and camera has less influence as the points are far away from the scanning position and as the angular difference between corresponding points vanishes. Both other parameters are very stable

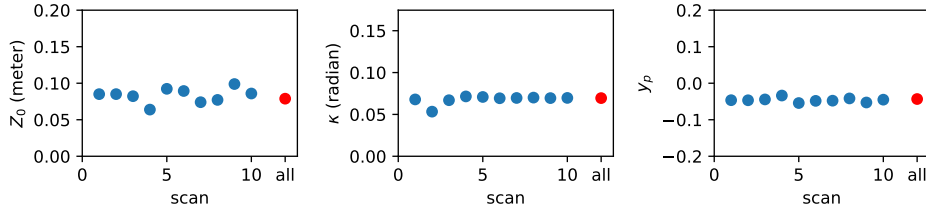


Fig. 8: Variation of the estimated parameter values in different scans (blue) using the same setup and all scans combined (red). TODO: MEAN und Std-Abweichung einzeichnen

for almost all scans. We did another registration using all 10 scans simultaneously by summing up their histograms (red points in Fig. 8). Especially the translation estimation benefits from adding more data from different scan positions.

5.2 Run Time Analysis

Method	1 Scan	5 Scans	10 Scans
Basic NMI (particle swarm)	6.3 h	> 20 h	> 40 h
Basic NMI GPU	498.63 s	2342.76 s	4358.91 s
NMI + smoothing (256 bins)	342.48 s	1777.93 s	3631.66 s
NMI + smoothing (16 bins)	545.77 s	2383.72 s	4271.45 s
GPU implementation	19.87 s	117.62 s	143.39 s

Table 1: Run times of the different approaches for one calibration on a CPU and on the GPU implementation. TODO: Accurry

We analyzed the run time of our method and the base version without any smoothing and compared the pure CPU and GPU solutions. The results are shown in Tab. 1. The horizontal resolution was set to 5000 pixels for all scans. The run times for the basic version of the algorithm on the CPU for 5 and 10 scans were extrapolated from the run time of the GPU version of the same algorithm due to their very high run time.

Obviously, the CPU version without smoothing of the objective function is not feasible due to high run time. Smoothing and reduction of the histogram size decreased the run time significantly due to the Nelder-Mead optimization, requiring significantly less evaluations of the target function. Note that the CPU version using the full 256 bins converged faster than the solution using 16 bins, but did so on a local maximum only. The GPU implementation is much faster, converging to the global maximum in just under 20 s for a single scan. It is fast enough to provide a fast calibration of the camera in the field at high resolutions.

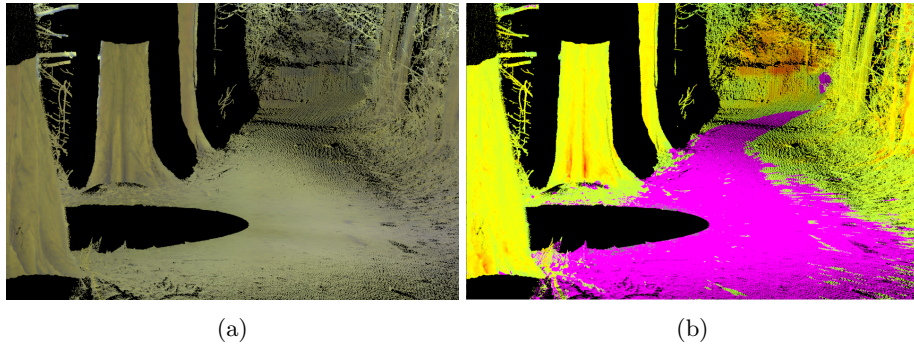


Fig. 9: Point cloud viewed in RGB (a) and NDVI (b). Yellow and red colors in the NDVI image indicate plants, while drivable, vegetation-free pathways are highlighted in magenta. Best viewed in color.

5.3 Application Example

To demonstrate the practical benefits of hyperspectral point clouds, we demonstrate an application example, where we used the well-known NDVI index for segmentation of a path way. For that, we computed the NDVI index for the point cloud taken together with the panorama presented in Fig. 2. Using a simple threshold filter, we were able to detect the existing pathway as presented in Fig. 9. Comparing the two pictures, it is obvious that such an easy segmentation would be hard to do based on a RGB image. This example is just one of the new possibilities that open up when using hyperspectral 3D data in robotic applications, which will be further explored in future work.

6 Conclusion

In this paper, we have presented an approach to calibrate a hyperspectral line camera against a terrestrial laser scanner on a mobile robot. For this, we used the well-known Normalized Mutual Information (NMI) approach with smoothing of the objective function to increase the robustness and reduce the run time. We have shown that the reduction of the number of bins in the histograms delivers more stable results and that the use of several scans can be beneficial for improving the quality of the estimated parameters. The combination of these approaches allows to achieve accurate calibration on a mobile system without the need for key-point detection or artificial markers in the scene, which is extremely beneficial for real life applications, since a calibration can be done on demand in the field. If the system is equipped with a GPU, the computation can be sped up significantly using our CUDA implementation, making it possible to compute the registration ad-hoc if needed.

Bibliography

- [1] S. Buckley, T. Kurz, J. Howell, and D. Schneider. Terrestrial lidar and hyperspectral data fusion products for geological outcrop analysis. *Computers & Geosciences*, 54:249–258, 2013.
- [2] T. Luhmann, S. Robson, S. Kyle, and I. Harley. *Close Range Photogrammetry: Principles, Methods And Applications*, volume 25. 2010.
- [3] J. Nieto, S. Monteiro, and D. Viejo. 3D geological modelling using laser and hyperspectral data. In *Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International*, pages 4568–4571. IEEE, 2010.
- [4] G. Pandey, J. McBride, S. Savarese, and R. Eustice. Automatic extrinsic calibration of vision and lidar by maximizing mutual information. *Journal of Field Robotics*, 32(5):696–722, 2015.
- [5] D. Schneider and H. Maas. A geometric model for linear-array-based terrestrial panoramic cameras. *The Photogrammetric Record*, 21(115):198–210, 2006.
- [6] C. Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001.
- [7] A. Sima, S. Buckley, T. Kurz, and D. Schneider. Semi-Automated Registration Of Close-Range Hyperspectral Scans Using Oriented Digital Camera Imagery And A 3d Model. *The Photogrammetric Record*, 29(145):10–29, 2014.
- [8] C. Studholme, D. Hill, and D. Hawkes. An overlap invariant entropy measure of 3D medical image alignment. *Pattern recognition*, 32(1):71–86, 1999.
- [9] Z. Taylor and J. Nieto. A mutual information approach to automatic calibration of camera and lidar in natural environments. In *Australian Conference on Robotics and Automation*, pages 3–5, 2012.
- [10] R. Xu, Y. Chen, S. Tang, S. Morikawa, and Y. Kurumi. Parzen-window based normalized mutual information for medical image registration. *IEICE transactions on information and systems*, 91(1):132–144, 2008.
- [11] Q. Zhang and R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference On*, volume 3, pages 2301–2306. IEEE, 2004.
- [12] X. Zhang, A. Zhang, and X. Meng. Automatic Fusion of Hyperspectral Images and Laser Scans Using Feature Points. *Journal of Sensors*, 2015: 1–9, 2015.