Assessment of the possibility of using the SAND library for processing point clouds in the Big Data environment on the example of UAV-LiDAR data for a forest

Dorota MARCZYKOWSKA, Jarosław CZAJKA, Wojciech OSTROWSKI, Magdalena PILARSKA-MAZUREK, Poland

Key words: Engineering survey, Geoinformation, Laser scanning, Remote sensing

SUMMARY

UAV-LiDAR surveys deliver very dense point clouds, with over 300 pts/m2. This density is close to that of terrestrial laser scanning (TLS) and allows to use UAV data in almost the same way as TLS in some applications. One of the application areas is forest management, where high density of point cloud enables automation of many forest inventory and planning processes, but TLS surveys are nearly impossible to be done due to dense vegetation and very low efficiency of measurements. Together with Dragonfly Vision company, we are carrying out a project in which a solution using UAV-LiDAR data for forest inventory and planning processes is built. Such a dense point cloud for large areas means billions of points to process and requires a lot of computing power. A tool combining GPU and the benefits of distributed environment has very considerable potential to improve analysis of this kind of data. SAND library is a part of CENAGIS project, developed by the Faculty of Geodesy and Cartography of the Warsaw University of Technology. This Python library allows to analyze and process point clouds using GPU and is designed to work with pySpark. Appropriate experiments were carried out to determine the possibility of using SAND library and the calculation time needed for large data sets. The possibilities of creating canopy height model (CHM) and determining forest structural attributes were also examined. In this study multiple ways to create height model using SAND were tested, such as: maximal raster value, planes fitting, and pit-free-like approach with processing data in layers. Capability of determining single tree features based on point cloud, like height or 2D area of canopy's extent was also investigated. The library implements few clustering methods and statistics for clusters, which can be used to analyze individual trees. All these functions show not only that manipulating a point cloud of forest by SAND and pySpark is possible, but also that these instruments together enable creation of comprehensive tools for foresters. These tools are able to process data obtained for large areas in a reasonable time. Thanks to the variety of methods, algorithms and the availability of a wide parameterization, SAND library also provides an excellent opportunity for research on the algorithms for creating individual products related to forest management.

Assessment of the possibility of using the SAND library for processing point clouds in the Big Data environment on the example of UAV-LiDAR data for a forest

Dorota MARCZYKOWSKA, Jarosław CZAJKA, Wojciech OSTROWSKI, Magdalena PILARSKA-MAZUREK, Poland

1. INTRODUCTION

The project entitled *Improving methods of acquisition and processing of remote sensing data from unmanned aerial vehicles* concerns research on the possibility of using laser scanning sensors and multi- and hyperspectral cameras in environmental applications related to forested areas. Laser scanners are an excellent tool for assessing the height of tree stands (Naesset, 1997). The monitoring of their condition is the main purpose of remote sensing work in forest districts and national parks. Multispectral and hyperspectral cameras successfully provide information on the type and condition of vegetation. The integration of two techniques is a method of synergy of remote sensing information proven in the literature (Guimarães et al., 2020). The aim of the project is research and subsequent implementation of services for collecting laser scanning data from UAV platforms (ULS) and multi- /hyperspectral data and their processing into products enabling management of forested areas. The potential application of the developed methodology is associated with smaller three stands, parts of national parks and individual forest districts, for which it is unprofitable to carry out manned flight missions several times a year. As part of the project, selected solutions are implemented to guarantee the required accuracy in the form of applicants ‘services. The novelty of the solutions is the development of methodology for the integration of lidar and multi- / hyperspectral data from UAV platforms in specific applications related to forested areas, which are the most difficult to ensure accuracy and quality of remote sensing data.

Laser scanning data is a discrete description od of the terrain and all the objects that are on the terrain. Laser scanning point cloud charactrizes with 3D information (X, Y znd Z coordinate for each point) and attributes such as: classification, intensity, ect. UAV laser scanning (ULS) data, comparing to airborne laser scanning (ALS) data, characterize with higher point cloud density. For ALS data the density of the point cloud is from few up to few dozens of points /m^2, while for ULS data the density of the point cloud is few hundrets of points /m^2.

The goal of the "Centre for Scientific Geospatial Analyses and Satellite Computations including laboratories for testing and authorising geomatic products (CENAGIS)" is to establish a unique, integrated infrastructure aiming at implementation of modern research works in the field of geomatics, geospatial engineering and geoinformation science. The objective of CENAGIS project is to establish the centre for scientific, geospatial analyses at world level, with the use of recent technologies, in particular geoinformation and IT technologies, allowing for remote access to unique research laboratories for the large groups of scientists and collaborating, innovative technological companies (https://www.cenagis.pw.edu.pl/cenagis_eng/About). A part of CENAGIS infractructure is a BigData component. A part of BigData component is a SAND (Statistical Analysis of Nonstructured Data) library for advanced lidar point cloud processing.

2. SAND LIBRARY

The SAND (Statistical Analysis of Nonstructured Data) library was created within the CENAGIS project, the resources of which include clouds of points from ALS (Airborne Laser Scanning) for the entire territory of Poland. It is a very large set of data, and in order for users to process this data in a convenient and efficient way, a proper tool was needed. It is not only performant, but also integrated with the technologies used in CENAGIS subsystems.
SAND is a programming library for processing point clouds. This library performs most of the operations on the GPU, using technologies such as CUDA and C ++, but it also has a Python interface. It is a tool that allows many diverse operations on point clouds, such as: raster creation, neighborhood exploration and clustering. An important feature of this tool is that its implementation uses structures and data types that are supported by Apach Spark, which is crucial for the possibility of using SAND in a distributed computing system.
The library is adapted for use on a single computer with an NVIDIA graphics card, it is not a distributed framework, so using it in a distributed environment using pySpark requires the use of pandas UDF, which takes the pandas DataFrame structure as input and returns the same type of structure. Of course, the computing cluster must be equipped with graphics cards and make them available with spark executors.

3. CENAGIS SAND AND FORESTRY

UAV-LiDAR data greatly facilitates the forest inventory process based on the analysis of single trees (Guimarães et al., 2020). A particularly important product made from point clouds is Canopy Height Model (CHM; Goodbody, 2017). On its basis, segmentation of individual trees is carried out and geometrical features of individual trees, such as tree height or crown coverage area are determined. A normalized point cloud is needed to create it. For normalization, however, a Digital Terrain Model (DTM) is needed, and before that, ground classification has to be performed. All these elements can be done with SAND. The following sections describe the functions available for this and examples of slightly more complex algorithms that can be implemented using the methods available in this library.

3.1 Ground classification

One of the functions of the SAND library is ground classification. This function is based on densification of the set by adding successive points that meet given conditions, mainly related to the direction of the normal vector and the height difference. The process is based on neighborhood analysis and is performed entirely on the GPU. In this case, one classification method can be used. However, there is a possibility to manipulate the values of many parameters, such as: the size of the neighborhoods (radius and number of points) used to determine the normal vectors, or in the ground growth stage - the thresholds of the angles between the normal vectors, the maximum height difference, the number of iterations, or the minimum distance between seed points. By using the point cloud tile size appropriate for the given parameters, it is possible to parallelize the classification process with the use of Apache Spark.

Default values have been defined for all parameters related to ground classification. Classification of ground in ULS data for forest area is not simple due to the different density of points on the ground (lower density under trees, much higher in intervals between them) and the presence of low vegetation. For this reason, for this type of data, the default classification parameters may not be as appropriate as they are for a built-up area. Figure 1 and Figure 2 show cross section of a part of the classified cloud with the default values of the process parameters.

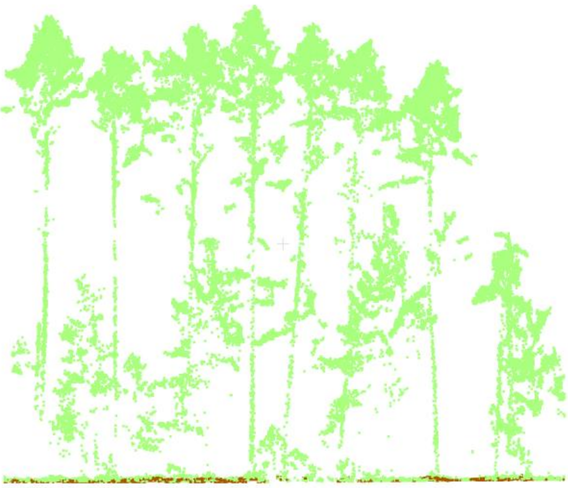

Figure 1 Result of ground classification with default values of parameters. Ground points are brown.

Figure 2 Result of ground classification with default values of parameters. Zoomed view. Ground points are brow.

An experiment was also performed comparing the computing time on a single local graphics card and computing time in the CENAGIS BigData environment for tiles with the size of 20 m and 5 executors (4 CPU + 1 GPU). The processed data covered an area with an area of 1 square kilometer (1 billion points). Table 1 shows the results of this experiment.

Table 1 Times of ground classification process

Environment type	Entire time of computation
1 local GPU	14 h
Spark – 5 x (4 CPU + 1 GPU)	3.5 h

3.2 DTM

Once the ground is classified, it is possible to proceed to the next important stage, i.e., the creation of a Digital Terrain Model (DTM), which is necessary for the point cloud normalization. SAND library allows to create raster from point clouds using several methods. Pixel values can be determined using a statistical value, such as: a minimum, maximum, or average value of subset of point cloud. You can also fit a plane for a certain area and assign the height resulting from this plane to the raster pixels. There are also natural neighbor methods, as well as triangulation and then interpolation of raster values based on determined triangles. The last two methods implement CPU-only algorithms, while the others implement GPU-based algorithms. Additionally, it is possible to interpolate the raster values in areas or individual pixels where there is no data in the point cloud. There are several options for choosing the method of this interpolation, including statistical analysis of neighborhood and the shape of the area in which pixels with data are searched for.

3.3 Cloud normalization

To create a Canopy Height Model (CHM) and to determine the height of the trees, a normalized point cloud is needed. For this, after creating the DTM, it is necessary to subtract the height of the terrain from the Z coordinate of the cloud points. This can be done in many ways. One of them is finding the closest point of the DTM raster for each point on the cloud and taking the value of the pixel found as the ground at a given point in the cloud (Stereńczak at al., 2016). SAND offers two ways to make the point cloud normalization. The first option is a function that is used to pass attributes between clouds. The algorithm searches for the points of one set of the closest points among those in the other set, and then rewrites the selected attributes. The second option is to perform the operation of finding points in a given surroundings, limited to one neighbor, and then assigning values from the selected neighbors of the raster to the points of the cloud. The second approach allows for more control over the process, e.g., looking for points only in a square-shaped area. Both paths use functions which computation is based on the GPU. Such an approach of assigning the height of the ground may lead to the formation of sharp height changes in the resulting set at the boundaries of the DTM raster pixel range (Figure 3).

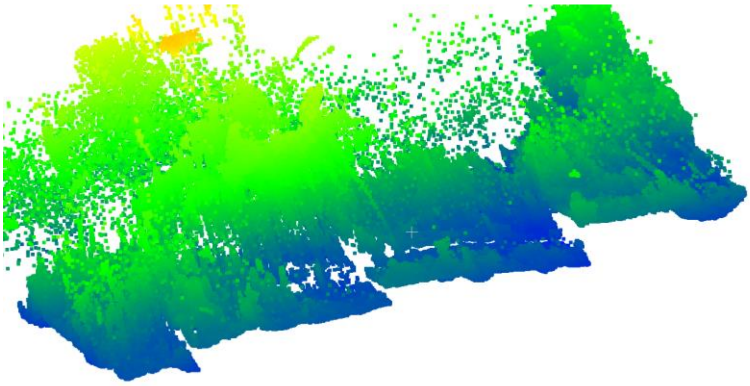

Figure 3 "Stairs" effect for ground after normalization with the nearest neighbor interpolation method

This effect is leveled when the pixel size corresponding to the density of cloud points on the ground is used. Unfortunately, this density in the case of forest scanning from UAVs is not constant, so it is not possible to create a regular grid that would meet this condition. The "stairs" effect can be removed by applying smooth interpolation between the pixel centers of the DTM. There are many such methods in SAND, e.g. natural neighbor interpolation, determination of the average value of several neighbors, determination of the weighted average value of the neighborhood with weights inverse to distance or with weights set by the user are available. All these methods, except the natural neighbor method, are performed on the graphics card.

3.4 CHM

CHM is one of the most important point cloud product in forestry analysis. It is especially important for forest inventory in the single tree analysis approach. Various algorithms such as: the Silva method (Silva et al., 2016), Li2012 method (Li et al., 2012), Dalponte method (Dalponte and Coomes, 2016) or marker controlled watershed algorithm (Chen et al., 2006) are used to segment the point cloud into single trees. CHM is used by most of these methods. In addition, CHM is used to detect treetops, read their height, and assess the size and distribution of gaps in the stand.

Many methods of creating CHM are used and known. The main examples are: highest point interpolation, pit-free algorithm, spike-free algorithm, graph-based progressive morphological filtering (GPMF; Quan et al., 2021). Some of these algorithms can be implemented using tools available in SAND library. One of such an example is the pit free method, which includes dividing a point cloud into horizontal layers, creating a separate raster for each layer and then merging them together. According to Khosravipour et al. (2014), rasters are created by triangulating a given data layer and determining the heights resulting from the given triangles. With SAND library this approach can be implemented, but it will mainly perform the computation on the CPU. The parametric flexibility and modularity of the library also enable the implementation of different versions of the algorithm. In the case of the pit free, for example, instead of triangulation the maximum value from a given neighborhood and interpolation of values for areas with too few points in the cloud can be used. Such an operation would be performed entirely on the graphics card. Of course, instead of the maximum value, another statistical value or the height resulting from the fitted plane can be used. The user also has full control over the number and size of layers. He can also make the raster creation dependent on height. The illustration below (Figure 4) shows an exemplary fragment of a CHM created with the modified pit free method.

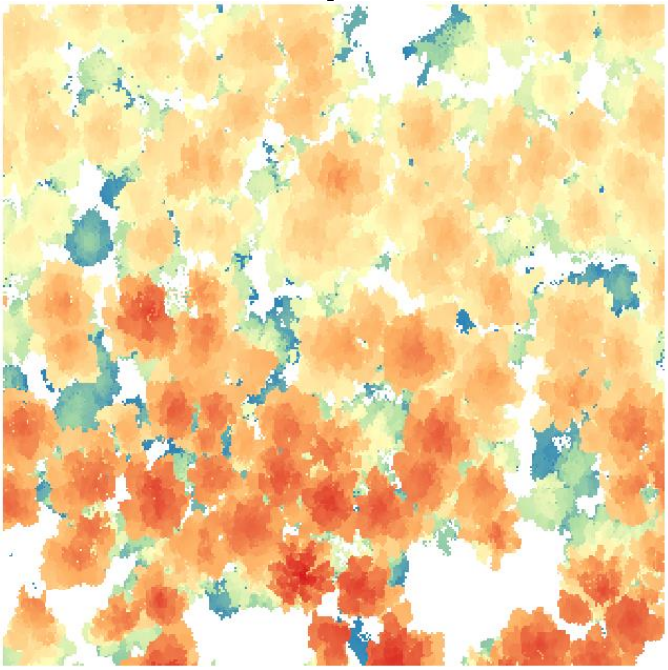

Figure 4 Fragment of CHM created using pit free with maximum value method

The algorithm for creating CHM directly available in SAND is the maximum value algorithm. In this case, in addition to the pixel size, user may define the input data set, e.g. only the first returns in point cloud are used, or points below the given height are not taken into account. This method is performed using the GPU and, like all raster creation methods, it can be performed in a distributed environment, using several graphics cards in parallel. Using the CENAGIS BigData environment with 10 executors with 1 GPU and 4 CPU allows to reduce the computing time over 8 times relative to single machine with one graphics card for dataset containing 162 million of points.

3.5 Single tree detection

In the forest inventory, Individual Tree Detection (ITD) is necessary to conduct when single trees are analyzed. In most cases, ITD method detects treetops as local maxima. The SAND library allows to perform this operation either directly on the point cloud or on the CHM raster. The user can set the size and shape of the search area. There is also an option to detect groups of the highest points, which allows to omit individual peaks and analyze the shape of the top of the tree. The figures below show results of searching the local maxima in a point cloud in a circular neighborhood with a radius of 3.0 m. In the first case (Figure 5), only 1 extremum is returned, and in the second case (Figure 6) - the 100 local highest points of the cloud are returned.

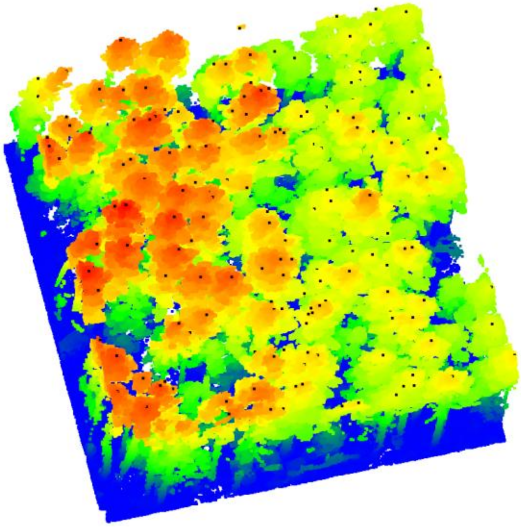

Figure 5 Local maxima in range 3.0 m - single points

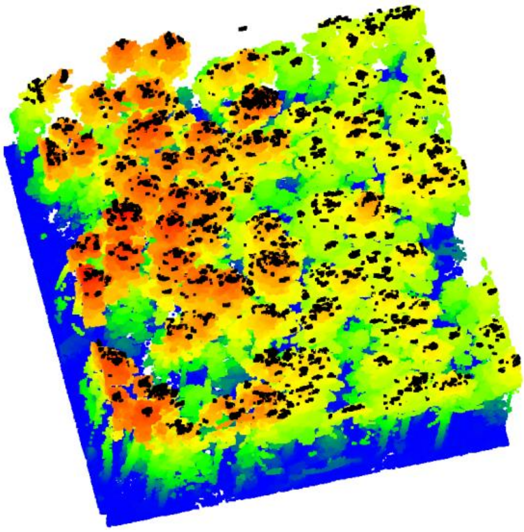

Figure 6 Local maxima in range 3.0 m - the highest 100 points

3.6 Single tree segmentation

For the analysis of individual trees, it is necessary to segment the point cloud or CHM into individual trees. SAND is a library focused on point clouds processing and therefore the segmentation methods implemented in it are rather intended for this type of data. Among the available segmentation methods, following methods are available: the K-means method, segmentation with given polygons, plane detection, region growth method and the mean shift method with different kernels shapes. All these methods, apart from plane detection, can be used in the segmentation of the forest into single trees, and all of them are implemented in such a way that they use the graphics card for the calculations. K-means and region growth methods require predefined cluster centers, preferably in the form of previously designated treetops. There is no such requirement in mean shift method. These three methods enable clustering at the 3D level, which is important for canopies of trees growing above each other. For the mean shift method various kernel shapes have been implemented, such as a sphere, a cylinder, a cuboid, or a Pollock model (Xiao et al., 2019), which in extreme cases takes the shape of a cone and a half-ellipsoid. Figure 7 and Figure 8 show examples of the results of UAV-LiDAR data clustering using the mean shift and K-means methods.

Figure 7 Example of result of K-means segmentation

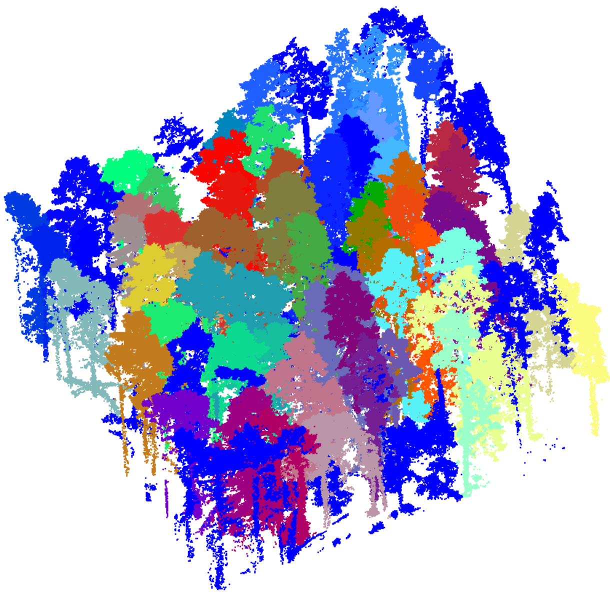

Figure 8 Example of result of mean shift segmentation

3.7 Single tree features

After individual trees detection, their analysis can be proceeded. This analysis involves, among others, tree species classification and health analysis, determining the number of individuals with given values of features, tree density, and stock. Among the examined features there are also those related to the geometry of the object and SAND may be helpful in determining the value of such features. It allows you to quickly determine statistics for clusters, such as the maximum and minimum value and the range, which is useful for determining the height of a tree and the size of its crown. Knowing the height of the crown base and having the CHM, it is possible to quickly estimate the volume of the crown. Another function available in SAND library is the determination of the 2D envelope of the point cloud, which can be used, for example, to create the outline of a crown or the outline of a vertical cross-section of a tree. Such an outline can be made in two ways: as a convex figure or as an alpha-shape figure. This function is one of the few functions that only uses the CPU. In enveloping multiple clusters, the process can be paralleled, and multiple processors can be used for computation in a distributed environment.

3.8 Additional rasters

SAND also enables the creation of rasters which present among all land cover or spatial distribution of dominant tree species. Having a classified cloud, it is possible to create a raster with pixel values representing e.g. the median or the mode of values from a point cloud. It is also possible to generate a raster that facilitates the analysis of species diversity in a given area by determining the abundance of species within a pixel or the ratio of the number of individuals of a selected species to the area. Another function of the library is generation of raster describing the density of trees or the surface of tree crowns in relation to the non-tree surface. Such raster would facilitate the analysis of the distribution of gaps between trees.

3.9 Function advancement levels in SAND

The SAND library is multi-level in terms of the level of interface advancement. This allows the SAND library to be used by both less advanced users as well as those who know exactly how they expect the generated product or those who want to conduct research and have control over each parameter. The drawing (Figure 9) presents an illustrative division of the library into levels.

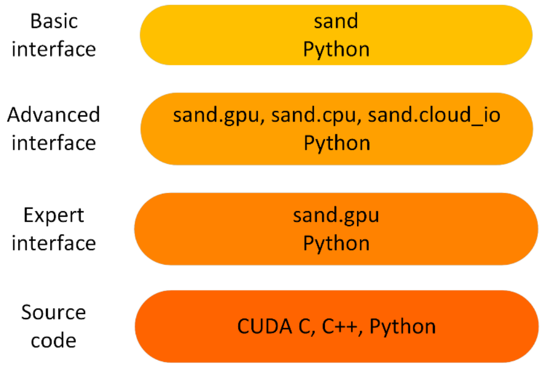

Figure 9 Diagram of SAND library structure

The highest level implemented in Python (basic interface) is intended for the least advanced user who needs to get a result with as little work and knowledge as possible. This interface implements the simplest methods with the default values of most parameters. Functions from this level are very useful if the user does not want to change the parameter values so that they will suit better to the input data and only needs an illustrative result, regardless of its quality. It is also the level at which a given operation requires the least amount of code from the user.

The next level (advanced interface) includes functions that allow to perform several operations simultaneously. For example, counting a neighborhood and then simultaneously determining several statistical values for it.

The next level (expert interface) is also in Python and includes functions that perform single, small operations such as determining the neighborhood, determining one statistic, filtering neighbors, determining the value of raster pixels by a given method. This level allows the greatest freedom to manipulate the parameters and elements of the entire process. Scientists or users who need to use some sort of atomic operation will be the most willing to use this level.

The lowest level of the SAND library (source code) is only available in C ++. This is a level that generally includes functions from the level above, but in a different language. The lowest level of the library contains operations performed on the graphics card and is written in the CUDA C language.

4. CONCLUSIONS

The SAND library is a tool with wide possibilities in the context of point cloud processing. Many of its features facilitate the creation of products needed for forest inventory and planning. The variety of the level of advancement of functions allows it to be widely used both for demonstration, educational or production purposes, as well as for research purposes. The tool has a wide functionality, which allows you to create comprehensive solutions. One of the main advantages of this library is its compatibility with a distributed environment supported by Apache Spark, which is rare among point cloud and GPU tools. This is possible thanks to the use of Apache Arrow structures for data transfer. Another advantage of this library is its Python interface. Thanks to this, the tool is convenient to use and consistent with the BigData CENAGIS computing environment. It can be used in a distributed environment, but also on a single machine. Computing with Apache Spark makes sense for a really large dataset. For smaller files that can fit in the graphics card memory at once or require data division into only a few tiles, a better solution is to use a single machine due to the difficulty of selecting parameters for the process of data dissipation and a considerable overhead of the computation time associated with this process.

ACKOWLEDGMENT
This paper is co-financed by an interdisciplinary research team by Dragonfly Vision company and Faculty of Geodesy and Cartography of Warsaw University of Technology within the project entitled, “Improving methods of acquisition and processing of remote sensing data from unmanned aerial vehicles", financed by the National Centre for Research and Development, grant no POIR.01.01.01-00-1027/20.

REFERENCES

Chen, Q., Baldocchi, D., Gong, P. & Kelly, M., 2006. Isolating Individual Trees in a Savanna Woodland Using Small Footprint Lidar Data. Photogramm. Eng. Remote Sens. 72, 923–932.

Dalponte, M. & Coomes, D. A., 2016. Tree-centric mapping of forest carbon density from airborne laser scanning and hyperspectral data. Methods Ecol. Evol. 7, 1236–1245.

Goodbody, T. R. H., Coops, N. C., Marshall, P. L., Tompalski, P., & Crawford, P. (2017). Unmanned aerial systems for precision forest inventory purposes: A review and case study. Forestry Chronicle, 93(1), 71–81. https://doi.org/10.5558/tfc2017-012

Guimarães, N., Pádua, L., Marques, P., Silva, N., Peres, E., & Sousa, J. J. (2020). Forestry remote sensing from unmanned aerial vehicles: A review focusing on the data, processing and potentialities. Remote Sensing, 12(6). https://doi.org/10.3390/rs12061046

Khosravipour, A., Skidmore, A., Isenburg, M., 2016. Generating spike-free digital surface models using LiDAR raw point clouds: a new approach for forestry applications. Int. J. Appl. Earth Obs. Geoinf. 52, 104–114. http://dx.doi.org/10.1016/j.jag.2016.06.005.

Naesset, E. (1997). Determination of mean tree height of forest stands using airborne laser scanner data. *ISPRS Journal of Photogrammetry and Remote sensing*, *52*(2), 49-56.

Li, W., Guo, Q., Jakubowski, M. K. & Kelly, M., 2012. A New Method for Segmenting Individual Trees from the Lidar Point Cloud. Photogramm. Eng. Remote Sens. 78, 75-84.

Silva, C. A., Hudak, A. T., Vierling, L. A., Loudermilk, E. L., O’Brien, J. J., Hiers, J. K., Jack, S. B., Gonzalez-Benecke, C., Lee, H., Falkowski, M. J. & Khosravipour, A., 2016.

Imputation of Individual Longleaf Pine (Pinus palustris Mill.) Tree Attributes from Field and LiDAR Data. Can. J. Remote Sens. 42, 554–573.

Stereńczak, K., Ciesielski, M., Bałazy, R., & Zawiła-Niedźwiecki, T. (2016). Comparison of various algorithms for DTM interpolation from LIDAR data in dense mountain forests. *European Journal of Remote Sensing*, *49*, 599–621. https://doi.org/10.5721/EuJRS20164932

Quan, Y.; Li, M.; Hao, Y.; Wang, B. (2021): Comparison and Evaluation of Different Pit-Filling Methods for Generating High Resolution Canopy Height Model Using UAV Laser Scanning Data. Remote Sens. 2021, 13, 2239. https://doi.org/10.3390/rs13122239

Xiao, W., Zaforemska, A., Smigaj, M., Wang, Y., & Gaulton, R. (2019). Mean shift segmentation assessment for individual forest tree delineation from airborne lidar data. *Remote Sensing*, *11*(11), 1–19. https://doi.org/10.3390/rs11111263

BIOGRAPHICAL NOTES

Dorota Marczykowska
She has received her M.Sc. degree in geodesy and cartography (specialization: geodesy and satellite navigation) in 2018 from the Warsaw University of Technology, Warsaw, Poland and then received Eng. degree in geoinformatics in 2019 from the same University. She has been working as a research assistant in the Department of Photogrammetry, Remote Sensing, and Spatial Information Systems, Faculty of Geodesy and Cartography, since 2021. Her research interests include UAV photogrammetry and aerial and UAV lidar. The main topic of her research is creating tools for processing and analyzing of UAV lidar in different applications.

Jarosław Czajka
He is a CEO of Dragonfly Vision company, which specializes in capturing various kind of UAV data, including RGB, multispectral and lidar data, as well as training future UAV pilots. He is personally also a UAV pilot and privately a drone enthusiast.

Wojciech Ostrowski
He received his M.Sc. degree in geodesy and cartography (specialization: photogrammetry and remote sensing) in 2013 from the Warsaw University of Technology, Warsaw, Poland, where he has been working as a research-teaching assistant in the Department of Photogrammetry, Remote Sensing, and Spatial Information Systems, Faculty of Geodesy and Cartography, since 2013. His research interests include aerial and UAV photogrammetry and UAV lidar. The main topic of his research is the photogrammetric processing of oblique aerial images and the application of Machine Learning for urban semantic 3D meshes. His other research interest involves GIS and photogrammetry applications in archaeology. He is a member of the Polish Society for Photogrammetry and Remote Sensing, Computer Applications and Quantitative Methods in Archaeology (CAA), and Aerial Archaeology Research Group (AARG).

Magdalena Pilarska-Mazurek
She has received his PhD degree in geodesy and cartography in 2021 from the Warsaw University of Technology, Warsaw, Poland, where she has been working as a research-teaching assistant in the Department of Photogrammetry, Remote Sensing, and Spatial Information Systems, Faculty of Geodesy and Cartography, since 2016. Her research interests include UAV photogrammetry and aerial and UAV lidar. The main topic of her research is the processing and analyzing of aerial and UAV lidar and the application of UAV data in different applications. She is a member of the Polish Society for Photogrammetry and Remote Sensing.

CONTACTS

M.Sc. Dorota Marczykowska
Warsaw University of Technology
Plac Politechniki 1
00-661 Warsaw
POLAND
Email: dorota.marczykowska@pw.edu.pl