FULL-LENGTH REPRINT – differs from the original in layout but not in contents
Chmielewski L.J., Bator M., Olejniczak M., Advantages of using object-specific knowledge at an early processing stage in the detection of trees in LIDAR data.
In L.J. Chmielewski et al., editors, Proc. Int. Conf. ICCVG 2014, volume 8671 of Lecture Notes in Computer Science, pages 145-154, Warsaw, Poland, 15-17 Sep 2014. doi:10.1007/978-3-319-11331-9_18

The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-11331-9_18 .

# Advantages of Using Object-specific Knowledge at an Early Processing Stage in the Detection of Trees in LIDAR Data*

Leszek J. Chmielewski, Marcin Bator, and Marcin Olejniczak

Warsaw University of Life Sciences (SGGW), Warsaw, Poland
Faculty of Applied Informatics and Mathematics (WZIM)
{leszek_chmielewski,marcin_bator,marcin_olejniczak}@sggw.pl
http://www.wzim.sggw.pl

**Abstract.** In some imaging setups the following assumptions hold: the objects are opaque and viewed only from one point, their surface is continuous at least piecewise, and the occluding objects are small with respect to the viewed objects. In addition, in the application of our interest the images can be treated similarly to the case of the plane of light. This made it possible to design algorithms with some desired features: the segmentation based on sorting the data according to angle and the version of the object verification method using fuzzy voting with the positive and negative evidence. The algorithms have some opposite and complementary features which could be used in application to LIDAR data in the measurements of trees and forest.

**Keywords:** Opacity, sorting, angle, Hough transform, voting against, negative evidence, LIDAR, trees, forest, breast-height diameter

## 1 Introduction

One of the promising applications of the LIDAR (LIght Detection and Ranging) measurement technique is forestry. There is much literature on the measurements of trees and forest from the air as well as from the surface. Some of the most recent publications are for example [3,12,13,14]. The problem can be also related to depth and distance recovery [6] and surveillance [8].

An important challenge in LIDAR measurements, as it is in the case of all visual techniques, is opacity and occlusion. The repetitive structure of the forest makes it possible to assess the influence of occlusion on the measurement, to some extent [15]. In this paper we shall use occlusion in a positive way, to improve the quality of the results. A new concept of operating on sorted data will be presented. It will help to organise and speed up the calculations and will make it possible to operate on trees in a highly cluttered environment.

The simple observation that the viewed objects are opaque was used in our previous work [5]. In that paper the concept of voting in the Hough transform not only for, but also against the presence of an object was introduced. In the present paper it will be modified, extended and tested.

* L.J. Chmielewski et al. (Eds.): ICCVG 2014, LNCS 8671, pp. 145-154, 2014.

The concept of voting for and against has been first proposed in [4]. In the one-point version of the Hough transform the image points potentially belonging to an object voted for its presence, and the neighbouring points voted against, which yielded a well contrasted peak in the accumulator. This concept has been passed over in silence by the publications which cited that paper, however. Our formulation of this problem is entirely different and it directly addresses the shape of the object.

The rest of the paper is organised as follows. In Section 2 the setting in which the measurement problem is considered will be presented. In Sections 3 and 4 two new or modified algorithms designed to detect and measure the tree sections will be described. In the following Section the results from these methods will be shown, and in Section 6 the results will be discussed and the methods will be compared. The last section concludes the paper.

The authors of the sections are as follows: Section 3: M. Bator; Section 4: L. Chmielewski (description, implementation of object detection, visualisation) and M. Olejniczak (modifications to the voting algorithm with respect to [5], implementation of voting); remaining Sections: all the authors.

## 2 The measurement problem

In the present paper we shall try to receive as good results as possible from one horizontal layer of the LIDAR data. In the case of trees the height at which such layer is located will be the breast height (1.3 m) traditionally used in forestry to measure tree stem diameter.

The data were scanned at the stand near Głuchów belonging to the Grójec Forest District, Mazovian Voivodship (Central Poland), with the ground-based FARO LS HE880 LIDAR scanner. The LIDAR is located at the origin of the Cartesian coordinate system. The part of data chosen for presentation in this pilot study extends in the range $x \in [-9.16, 8.18]$ m and $y \in [-4.64, 5.79]$ m and contained 11 trees. The vertical coordinate extends $\pm 0.01$ m around the breast height. The set contains a total of 548 789 points. The data are visible in the Figures 2 and 4 in which the intermediate and final results are shown.

A large number of objects which interfere with the traces of the tree trunks can be seen. These are mainly smaller trees and bushes. The small plants have branches and leaves which are much smaller than the tree trunks of interest, but they form relatively dense clouds of objects that play the role of clutter. Some light rays can transverse the clutter, reflect from the thick trees and come back to the LIDAR. Our algorithms should operate on such noisy data.

The maximum possible diameter of a tree is a parameter to be set for a given forest stand; in the present case it was 40 cm. The minimum tree diameter is traditionally set to 5 to 8 cm, due to that the thinner objects are considered as tree branches rather than stems.

The important assumptions for this kind of images are: the opacity of the objects, the continuity of the surfaces of the objects, the near-vertical shape of the objects of interest, and the cross-section shape close to circular.

## 3 Algorithm Based on Sorting

In general, the algorithm is aimed at forming subsets from the set of measurement points. Each subset can be a separate tree, if it fulfils the conditions given further.

Initially, the points are transformed to the polar coordinate system $Ol\phi$.

In the below described algorithm, for a pair of points $P_i(l_i, \phi_i)$, $P_j(l_j, \phi_j)$, two conditions related to distance from the origin are used. Point $P_i$ is *considerably farther* from (or *closer* to) the origin than $P_j$ if $l_i - l_j > t_l$ (or $l_j - l_i > t_l$). In the conditions the same threshold can be used. Therefore, the number of parameters of the algorithm is one.

If there are more than one point for a given angle, the farthest one is left and the remaining ones are deleted. This conforms to the assumption on the opacity of objects: if any of the points closer to the LIDAR belonged to a tree, this tree would occlude the farther points, so they could not exist. Therefore, the closer points are noise. They did not occlude the farther points because they could have belonged to small objects from the height slightly different than the farther point. The points which remained are sorted according to angle $\phi$.

A new subset is formed, initially containing the first point.

The next point is considered and it becomes the current point. If it is considerably farther from the origin than the previous one, a new (farther) subset is formed, the current point is included into it, and the previous subset is closed. If it is considerably closer, a new (closer) subset is formed and the current point is included into it, but the previous subset is not closed. In the remaining case the current point is included into the current subset.

The following next point is considered (new current one) and the same steps are repeated. If in the previous step a (closer) subset was closed, the current point is compared with the last included point from an open subset. If there are more than one open subsets, the one is used for which the angle difference from the current point to the last included point of the subset is the smallest.

It should be noted that the angle is periodical, so the analysis can lead to including the last analysed points to the first subset formed.

After the subsets are established, each of them is assessed. Too small subsets are rejected (below 75 points). Accepted subsets are processed with the one-point Hough transform for circles [1]. Finally, existence of points behind the circle is checked. This is effective thanks to that the data are sorted.

The clear structure of the algorithm is apparent. However, it properly reflects the assumption of non-transparency and piecewise continuity of objects. It works well if the objects to be detected are larger than those which are considered as noise. No assumptions on the shape of the detected object is made. The continuity of the object's border is captured in a simple way by the parameter $t_l$ which is a measure of difference between an irregularity of one object's surface and a jump between two different objects.

The results are sequenced according to the angle, so they are perfectly suited for further approximation, for example like that in [9].

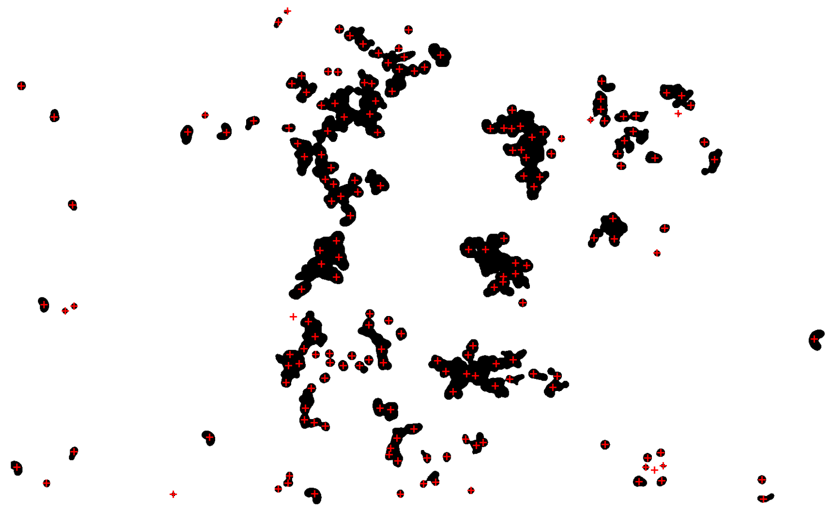


**Fig. 1.** Results of segmentation showing regions exceeding the threshold (black) and 180 maxima (crosses).

## 4 Algorithm Based on Fuzzy Voting for and Against

### 4.1 Segmentation

This algorithm works on the objects after their detection. Before detection the data are segmented with the fuzzy accumulation algorithm described in [5]. An accumulator with resolution of $0.01 \times 0.01$ m fuzzified with a parabolic fuzzification function defined on a mask $21 \times 21$ pixels is used. The result is thresholded at $0,001$ of the global maximum value. The thresholded map is shown in Fig. 1. Around each local maximum found, a circular region is selected with the radius equal to the smaller of the two numbers: distance to the nearest maximum and distance to the farthest black point in the current segment. Data belonging to these regions are stored as candidates for objects.

### 4.2 Detection and measurement

Object detection is performed as described in [5] with the two-point Hough transform, at the resolution of the accumulator $0.002 \times 0.002$ m. The number of trees found is excessive, as shown in Fig. 2. The Hough transform always returns the strongest instance of the object present in the data, irrespectively of how few data support it.

### 4.3 Voting for and against

The voting scheme has been modified with respect to the version proposed previously in [5]. The shapes of regions to which the positive nad negative votes can belong were changed and their membership functions were adjusted.

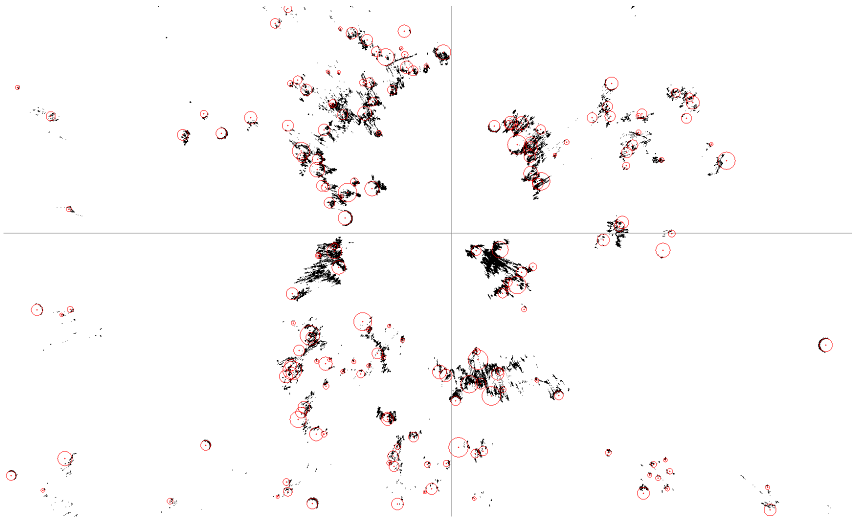


**Fig. 2.** Results of circle detection before elimination of weakly represented objects: 11 true positives and 169 false positives.

The idea is that a point located behind an object contradicts its existence, so it votes *against*. A point in front of an object votes *for* if it is at or near the boundary, and *against*, if it is inside the object.

The conditions of being behind, near the boundary or inside are fuzzy. In Fig. 3 the regions reflecting the notion of *near* are marked with grey signs. Positive and negative signs indicate the voting for and against.

The algorithm operates as follows. The input of a measurement point $P$ to the evidence concerning the circle $s(C, R)$ is considered (Fig. 3a).

To find whether the point is in front or behind the circle, the angle $\measuredangle LCP$ is compared to $\measuredangle LCS$. The line $b$ is the boundary.

In case $P$ is in front, the voting result depends on the location of $P$ with respect to the circle and its neighbourhood limited by the inner and outer circles $s_i(C, R)$, $s_o(C, R)$. Half-width of the ring between them is determined by the position of point $D_f$ on $\overline{CS}$. Between $s_i$ and $s_o$ the vote is positive, inside $s_i$ it is negative, and outside $s_o$ it is zero (*abstention*), according to the function shown in Fig. 3b. The curvilinear fragments were modelled with the sine function. The coordinate related to the function is $x_f$ extending from the centre $C$ outside.

In case $P$ is behind, the voting result depends on its location with respect to the angle $\angle CLS$ and its boundary regions $\angle D_b LS$ (and symmetrically, below $a$). Outside the angle there is abstention. Between the boundary regions (below line $d_b$) the vote is strictly against. In the boundary regions, due to the possible error in the determination of radius $R$, the vote should gradually go down from 0 to $-1$. The voting function is shown in Fig. 3c. The coordinate related to the function is the angular coordinate $x_b$ going out from the line $a$.

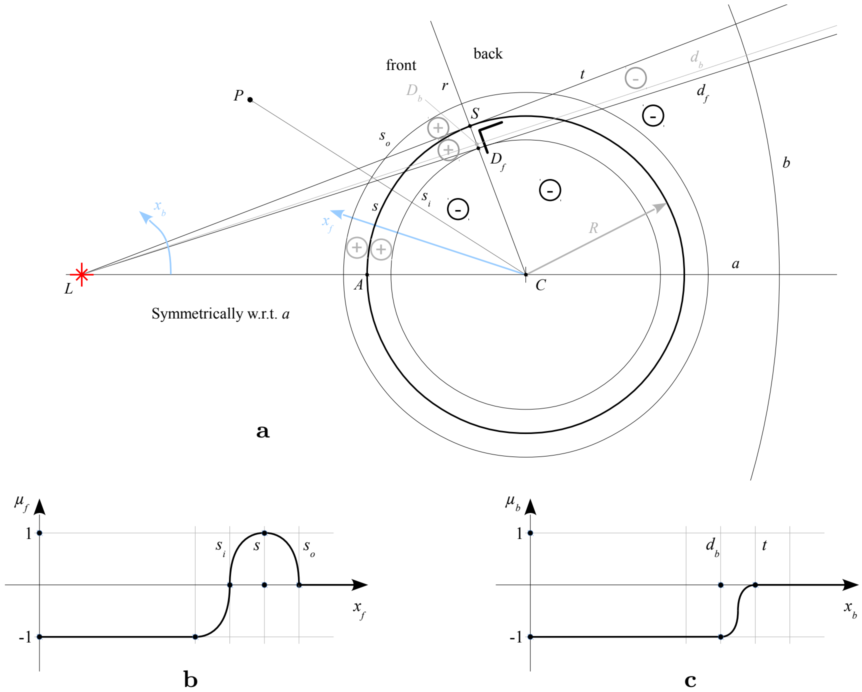



**Fig. 3.** The layout and notations used in fuzzy voting for and against the circle $s(C, R)$. (**a**) General layout; (**b**) voting membership function in front, and (**c**) behind the circle.

The amount of fuzziness in the process is regulated by the locations of points $D_f, D_b$. It can be expressed by a parameter $\delta. = |D.S|/|D.C| = |D.S|/R$. Two points $D_f$ and $D_b$ or one $D$ can be used for the front and back.

The votes coming from all the available points $P$ should be counted, separately from the front – for and against, and from the back – against. The number of votes necessary for object acceptance can be set according to the needs of the application, for example, the ratio of front negative votes to the positive ones should be limited and the minimum number of votes should be related to the circle radius. The acceptable number of negative back votes should be very small or zero.

It is interesting why the voting against was overlooked, despite the proposition made in [4], although other voting concepts were developed, including the advanced fuzzy voting schemes. It can be supposed that this was because the negative voting would result in assigning a negative value to the fuzzy membership function, which is in contradiction with the established fuzzy set theory. We pass over this assumption by stating that the negative sign represents only the fact that the given data item belongs to the set of data which contradicts the existence of the object considered.

In the presented method, otherwise than in feature-based methods (e.g. [11]), solely the shape of the object is used to assess the quality of its detection and measurement. It can be treated as an extension of approaches in which the number of votes is compared to the dimensions of the object [10]. It can also be related to methods in which shape is considered as a whole (e.g. [7]).

## 5 Results

The algorithm with sorting (Section 3) was run with the parameter $t_l = 5$ pixels. The results are shown in Fig. 4a. Additionally to finding the 11 existing trees, two more were found as false positives. The running time, including all the operations: sorting, segmenting and HT was around 1 minute. All the times were checked with a PC with Intel Core i7 3740QM at 2.7 GHz running sequential 32-bit software.

The segmentation (Section 4.1) is a fast calculation and takes several seconds. The detection (Section 4.2) on 180 sets of data took around 25 minutes (the two-point HT has the complexity $O(n^2)$) and gave 180 circles. The algorithm with voting (Section 4.3) was run with $\delta = 0.2$ for both front and back, the number of admissible negative front votes limited to 1/3 of positive front votes, and the number of back votes (only negative possible) limited to three.

The voting was performed locally, for a given object only with its own data subset as received from the segmentation, and globally, with all the available 0.5 M of data for each object. In the first case the time was less than 5 s, and in the second it was around 20 s.

All the results are shown if Fig. 4.

The sorting algorithm detected all the 11 true objects and two false positives. The voting algorithm with local data rejected 132 objects from the 180 ones found after segmentation, leaving the 11 objects and 37 false positives. The voting algorithm with global data rejected further 34 objects, leaving 11 true positives and 3 false positives. The locations and radii vary between the results of the two algorithms. The false positives found by the two methods are different.

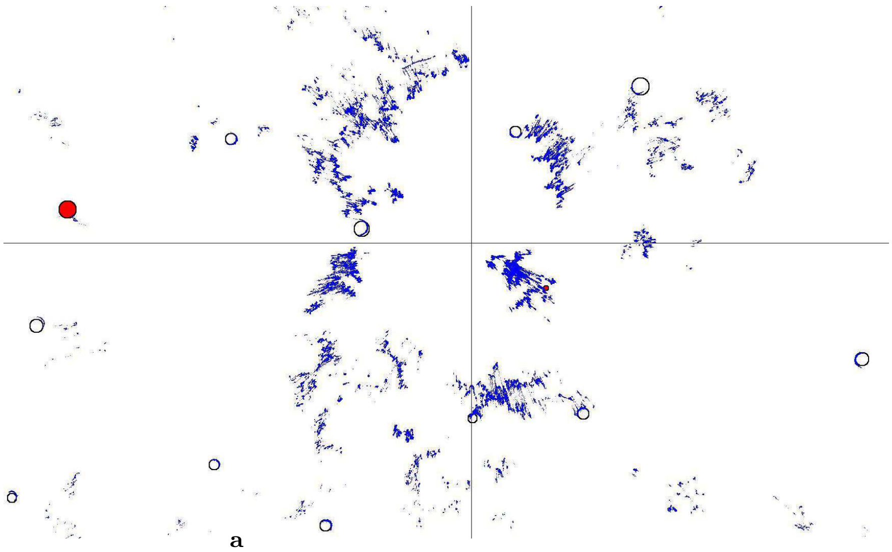
More details on the data and results are available at `www.lchmiel.pl/publ/`.

## 6 Discussion

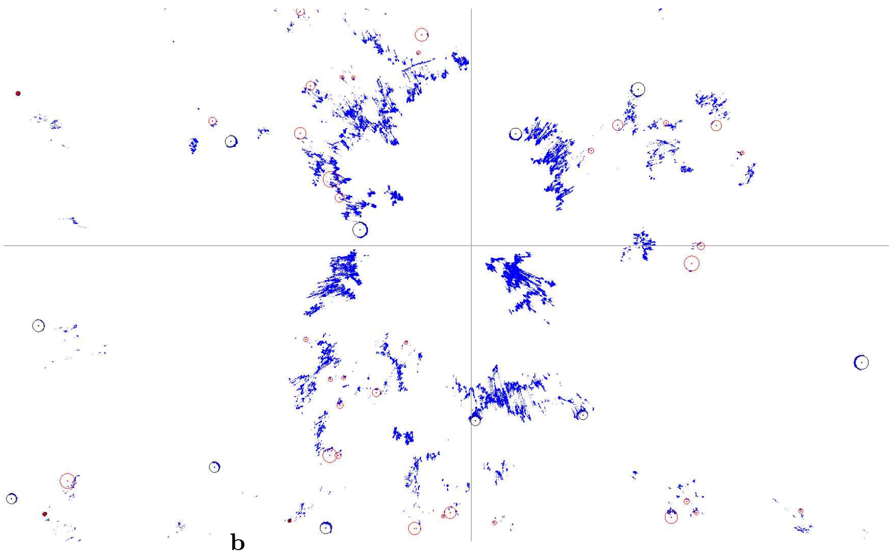
The two compared algorithms use entirely different concepts. However, to construct both of them the facts that the imaged objects are opaque and that the measuring apparatus can observe them from a single, stable point are used.

In the first algorithm based on sorting, the data are greatly reduced before the detection and measurement is done, which minimises the processing time. The shape of the object is not taken into account, so the algorithm is general. It can be used in many applications where small objects in front of the detected objects interfere in the detection process.

The second algorithm works on the detected objects. It verifies the correctness of detection. In the present form it can not be used to improve the true positive rate, but only to remove the false positives. The algorithm can be applied to different shapes of objects by properly designing the functions used.



a



b

**Fig. 4.** Results from the algorithms: (**a**) with sorting (Section 3); (**b**) with voting (Section 4), with local and global data. Blue: measuring points; black: circles found by the sorting algorithm and by the global voting algorithm; red inside: false positives; red empty: circles found by the local voting algorithm and not found by the global ones, all false negatives.

**Table 1.** Comparisons of the properties of the two algorithms used.

| No. | Property | Sorting | Voting |
|---|---|---|---|
| 1 | object opacity considered | yes | yes |
| 2 | object shape considered | irrelevant | modelled |
| 3 | front surface irregularity considered | yes | yes |
| 4 | occluding surface irregularity considered | no | yes |
| 5 | object detection method | any | any |
| 6 | when object measurement is made | after | before |
| 7 | globality/locality, natively | global | local |
| 8 | globality possible | natively | at some extra cost |
| 9 | no. of points that trigger a decision | one | parameterised |
| 10 | accuracy, in the example presented | 100% | 100% |
| 11 | specificity, the example presented | 82% | 73% |
| 12 | own speed | fast | fast |
| 13 | speed, together with object detection | fast (60 s) | slow (1500 s) |

The features of the algorithms are extensively compared in Table 1. The algorithms seem to be complementary in some aspects. For example, the sorting algorithm makes the segmentation quickly, while the initial segmentation is a weak point of the voting algorithm; specific shape is not relevant in the sorting algorithm (positive in case of segmentation), while it is considered in the voting algorithm (positive in object verification). The algorithms can be integrated in the analysis of LIDAR images of trees and in similar applications.

## 7 Conclusions

Occlusion and noise pose a challenge to the methods of object detection and measurement. The opacity of objects and interfering occluding elements and the fact that the measuring device observes the scene from one point can be used to construct algorithms with desired features.

In the algorithm for data segmentation based on sorting these observations made it possible to achieve high efficiency and generality. The structure of the algorithm is simple and it has from one to three parameters, according to the user's choice, which would simplify its optimisation if necessary.

In the algorithm for object verification based on fuzzy voting, the opacity was used to make it to enable not only positive voting, like in the classical Hough transform and its variants, but also to consider the evidence which can contradict with the existence of an object, thus enabling the rejection of false positive detections. The method can be redesigned for objects with various shapes.

The two presented algorithms could be used together in the frames of the detection of opaque objects in a cluttered environment, including the analysis of trees and forest measured with the LIDAR technology in the presence of bushes and small trees.

**Acknowledgements** We wish to thank Krzysztof Stereńczak form the Forest Research Institute and Michał Zasada from the Faculty of Forestry, Warsaw University of Life Sciences (SGGW), for providing the LIDAR data used in the experiments.

## References


1. D. Ballard and C.M. Brown. *Computer Vision*. Prentice Hall, Englewood Cliffs, New Jersey, 1982.
2. L. Bolc, R. Tadeusiewicz et al, editors. *Computer Vision and Graphics*, volume 7594 of *Lecture Notes in Computer Science*. Springer, 2012. doi:10.1007/978-3-642-33564-8.
3. G. Brolly, G. Király, and K. Czimber. Mapping forest regeneration from terrestrial laser scans. *Acta Silvatica et Lignaria Hungarica*, 9:135146, 2014. doi:10.2478/aslh-2013-0011.
4. C.M. Brown, M.B. Curtiss, and D.B. Sher. Advanced Hough transform implementations. In Alan Bundy, editor, *Proc. 8th Int. Joint Conf. Artificial Intelligence IJCAI 83*, pages 1081–1085, Karlsruhe, Germany, Aug 1983. William Kufmann.
5. L.J. Chmielewski and M. Bator. Hough transform for opaque circles measured from outside and fuzzy voting for and against. In Bolc et al. [2], pages 313–320. doi:10.1007/978-3-642-33564-8_38.
6. L. Czúni and D. Csordás. Depth-based indexing and retrieval of photographic images. In *Visual Content Processing and Representation*, volume 2849 of *Lecture Notes in Computer Science*, pages 76–83. Springer, 2003. doi:10.1007/978-3-540-39798-4_12.
7. D. Frejlichowski. Analysis of four polar shape descriptors properties in an exemplary application. In L. Bolc, R. Tadeusiewicz et al., editors, *Computer Vision and Graphics, Part I*, volume 6374 of *Lecture Notes in Computer Science*, pages 376–383, Springer, 2010. doi:10.1007/978-3-642-15910-7_43.
8. D. Frejlichowski, P. Forczmański, A. Nowosielski, K. Gościewska, and R. Hofman. SmartMonitor: An approach to simple, intelligent and affordable visual surveillance system. In Bolc et al. [2], pages 726–734. doi:10.1007/978-3-642-33564-8_87.
9. R. Kozera, L. Noakes, and P. Szmielew. Quartic orders and sharpness in trajectory estimation for smooth cumulative chord cubics. In L.J. Chmielewski, R. Kozera, B.-S Shin, K. Wojciechowski, editors, *Computer Vision and Graphics*, volume 8671 of *Lecture Notes in Computer Science*, pages 9–16. Springer, 2014. doi:10.1007/978-3-319-11331-9_2.
10. V.F. Leavers. *Shape Detection in Computer Vision Using the Hough Transform*. Springer Verlag, London, 1992. doi:10.1007/978-1-4471-1940-1.
11. T. Leś, M. Kruk, and S. Osowski. Automatic recognition of industrial tools using artificial intelligence approach. *Expert Systems with Applications*, 40(12):4777 – 4784, 2013. doi:10.1016/j.eswa.2013.02.030.
12. S. Miścicki and K. Stereńczak. A two-phase inventory method for calculating standing volume and tree-density of forest stands in central Poland based on airborne laser-scanning data. *Forest Research Papers*, 74(2):127–136, 2013. doi:10.2478/frp-2013-0013.
13. K. Olofsson, J. Holmgren, and H. Olsson. Tree stem and height measurements using terrestrial laser scanning and the ransac algorithm. *Remote Sensing*, 6(5):4323–4344, 2014. doi:10.3390/rs6054323.
14. K. Stereńczak, M. Zasada, and M. Brach. The accuracy assessment of DTM generated from LIDAR data for forest area – a case study for scots pine stands in Poland. *Baltic Forestry*, 19(2):252–262, 2013.
15. M. Zasada, K. Stereńczak, W.M. Dudek, and A. Rybski. Horizon visibility and accuracy of stocking determination on circular sample plots using automated remote measurement techniques. *Forest Ecology and Management*, 302(0):171–177, 2013. doi:10.1016/j.foreco.2013.03.041.