Archiwum Fotogrametrii, Kartografii i Teledetekcji, vol. 28, pp. 91-104
ISSN 2083-2214, eISSN 2391-9477
DOI: 10.14681/afkit.2016.007

DESIGN OF A TESTING METHOD TO ASSESS THE CORRECTNESS OF A POINT CLOUD COLORIZATION ALGORITHM

OPRACOWANIE METODY OCENY POPRAWNOŚCI DZIAŁANIA ALGORYTMU KOLOROWANIA CHMURY PUNKTÓW

Michał Pleskacz, Antoni Rzonca

Department of Geoinformation, Photogrammetry and Remote Sensing of Environment
AGH University of Science and Technology

KEYWORDS: data integration, algorithm testing, data quality, scan colorization

ABSTRACT: The paper discusses a testing method for a point cloud colorization algorithm. The point cloud colorization process is described in the context of photogrammetric and laser scanning data integration. A parallel algorithm is described following a theoretical introduction to the problem of LiDAR data colorization. The paper consists of two main parts. The first part presents the testing methodology via two aspects: (1) correctness of the color assigned to a given point, (2) testing of interpolation methods. Both tests are used on synthetic and natural data, and the results are discussed. The second part consists of a discussion of correctness factors associated with point cloud colorization as a typical case of process correctness in data integration. Three important factors are discussed in the paper. The first is correctness of the external orientation of the given image. The second is the ratio of the density of the point cloud and the GSD of the image. The third is the relative angle between the image and the scanned plane. All of the results are presented in the paper and the optimal range of the relevant factors is also discussed.

1. INTRODUCTION

Point cloud colorization is a standard point cloud process, which is applied prior to other mapping processes such as orthoscan generation, manual vectorization, and 3D visualization of the point cloud. Aerial, terrestrial, and mobile scanning data all undergo the colorization process. The perception of the details of a colorized point cloud is better and interpretation errors tend to occur rarely. Only a small number of unique scanners are currently available on the market that colorizes point clouds in the course of data acquisition. In the majority of cases, intensity scans are captured first and RGB information is assigned afterwards. The process of scan colorization is a case of photogrammetric and laser scanning data integration (Rzonca, 2013). In this case, the leading method consists of laser scanning, while the color from photogrammetric data represents additional information. In most cases, the collinearity condition is applied to the data. The data indispensable in this process consist of point clouds as well as images with internal and external orientation. In addition, all of the data should be found within the same coordinate system.

The paper discusses an evaluation method of new software development and point cloud processing performed at DEPHOS Software Ltd. The specific testing script, test data, and procedure were produced at DEPHOS, and the final results appear to be useful, especially in the context of photogrammetric and laser scanning data integration theory and practice.

2. THEORETICAL BACKGROUND

The problem of RGB quality of both images and orthoimages is very well known in photogrammetry.

The problem of RGB quality is discussed in several papers describing research focused on almost every stage of photogrammetric technology. The problem of radiometric quality of data appears in the course of data acquisition, which suggests that sensor calibration is a basic problem (Cramer, 2011; Markelin *et al.*, 2008). Radiometric correction is connected with the process of acquisition and normally occurs immediately after data capture (Norbega, Quintanilha, 2004; Rzonca, 2013). Radiometric quality control of orthoimages is widely discussed in the literature (Hoehle, Potackova, 2005; Jun *et al.*, 2010; Pyka, 2009, 2013). The aspect of radiometric quality following data fusion is also discussed (Pirowski, 2009). In this sense, point cloud colorization is a typical process, but the consequences of the errors of this process must be evaluated in order to draw useful analytical conclusions. This evaluation can be difficult and the source of the errors can be difficult to detect. In this case, the error consists of incorrect RGB assignment to specific points, and can be missed during the observation of the data. The error can be identified in overlapping zones colorized using different photos.

There are two potential sources of error in this process. The first source is the selection of an incorrect colorization algorithm and the second source is low accuracy or wrong relative configuration of the input data.

The colorization algorithm uses the collinearity equation to properly assign RGB values to specific points in a scan. There are two main procedures: The first procedure identifies the image coordinates of the puncture point on the line connecting the projection center and the scan point designated for colorization. The internal and external parameters of the image and coordinates of the point in question must be known to solve the collinearity equation (Fig. 1).

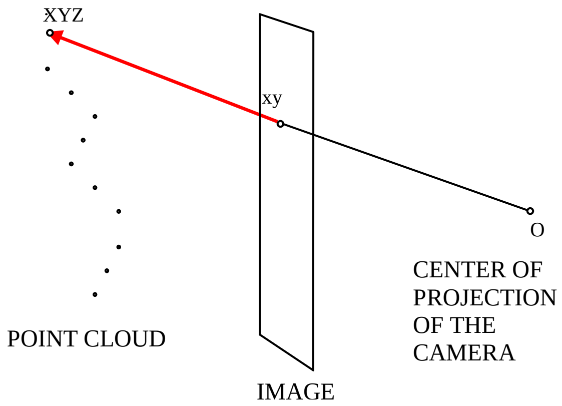

Fig. 1. Geometry of point cloud colorization (Rzonca, 2013).

The second procedure is employed to calculate RGB values based on the interpolation method selected by the user. Interpolation functions have to be checked to learn if they have been correctly implemented. There are three different interpolation methods used for point colorization.

Figure 2 shows a geometric presentation of these interpolation methods. The smoothing of the interpolated data is stronger using the bicubic method, while the nearest neighbor method provides the sharpest results.

A discussion of algorithm correctness yields an opportunity to mention three additional factors affecting the accuracy of point cloud colorization.

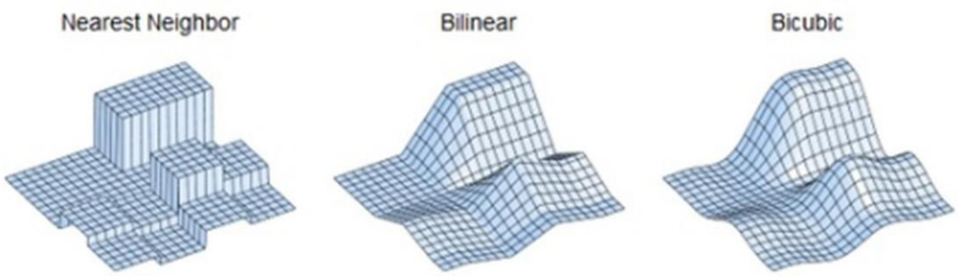

Fig. 2. Interpolation methods applied as part of the colorization algorithm (Getreuer, 2011).

The first factor is the accuracy of the linear and angular elements of external orientation (EO) of the given image related to point cloud density.

EO errors may cause translation of footprints of pixels in relation to the point cloud. In the simplest case of nearest neighbor interpolation, the translation of the center of the projected pixel should not be larger than half of the density of the point cloud. In cases of greater translation, the next point in the cloud will be colorized by the pixel corresponding geometrically with the previous point. All three interpolation methods are used in this test and discussed in this paper.

The second factor is the ratio of terrain resolution of the given image (GSD - ground sample distance) divided by the density of the point cloud. In point cloud colorization, the optimal range of the ratio should be given as a technical suggestion, helpful at the stage of data acquisition and preparation for the integration process. A ratio equal to one would be perfect, but is practically never observed.

The third factor is the relative angular orientation of the plane of the given image with respect to the point cloud to be colorized. For the purpose of clarity, the point cloud is assessed via its plane and named a “scan plane.” In this case, the optimal angle is 0, when the plane of the image and the scan plane are parallel, and the camera axis is perpendicular to the scan plane. In our research study, the maximum acceptable angle is sought depending on the method of interpolation used.

In the next few sections, we discuss in detail all issues associated with testing algorithm correctness and data specification for proper data integration processing.

3. THE TESTED ALGORITHM

The algorithm named CuScanColorizer used for point cloud colorization represents one outcome of a European Union project completed in 2015 by DEPHOS Software: “Research on mass storage, sharing, and processing of spatial LiDAR data.” The general results of the project concerning LiDAR processing were published (Będkowski *et al.*, 2015). All new algorithms need to be tested by software developers and specialists associated with specific fields; in this case, photogrammetry. The testing described in this study was performed as part of a master’s research program pursued by one of the co-authors of this paper (Pleskacz, 2014).

The herein described algorithm is executed from a batch file, where all necessary data are given. The key characteristic of the CuScanColorizer is very efficient parallel processing using graphic cards based on nVidia Cuda technology. The testing was necessary prior to the introduction of this product in the market. The algorithm uses the collinearity equation as the principal geometric condition in the colorization process, and all three interpolation methods mentioned earlier in this paper. The most important and innovative feature of the algorithm is that the data are specially organized and divided to be able to use parallel computation on a GPU (graphics processing unit). The input data here consists of a image with known internal and external orientation parameters as well as an ASCII point cloud to be colorized with coordinate and intensity values, but without RGB values. The output data consists of an RGB point cloud in ASCII format.

4. ALGORITHM CORRECTNESS TESTING

4.1. Testing procedure for correct color assignment

The testing of the algorithm was designed to assess the quality of the algorithm's output.

The test is based on an RGB comparison between the color assigned to points colorized by the CuScanColorizer and independently created using the Matlab testing tool. The Matlab algorithm uses analogical geometric and mathematical relationships as the CuScanColorizer, but the data is not specially organized for parallel computation. The Matlab algorithm uses traditional serial computation methods processed using a CPU (central processing unit), and it was implemented independently by a different programmer than the one implementing CuScanColorizer.

The color assignment was performed with the nearest neighbor interpolation method.

First, the Matlab script was used to generate a synthetic point cloud. The 3D coordinates of 17 points were then calculated for all three pixels of the synthetic image (Fig. 3). The image coordinates of 17 points were calculated in order to geometrically test the center and boundaries of the given test field (Fig. 4). The first point was found in the center of the theoretical pixel of the synthetic image. A total of 8 points were identified inside of the pixel, close to its boundary, while 8 points were found outside of the boundary. The global positions of these points were calculated using the collinearity equation, internal orientation (IO), and external orientation (EO) parameters of the given image as well as randomized global height.

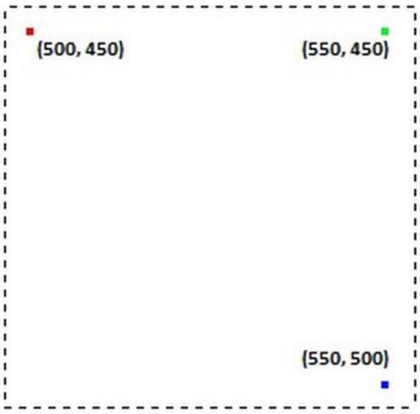

Fig. 3. Synthetic test image with 3 pixels (Pleskacz, 2014).

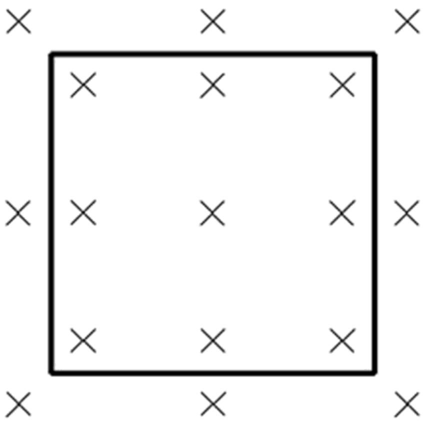

Fig. 4. Distribution of 17 test points in the synthetic cloud with a square pixel range (Pleskacz, 2014).

Colorization results produced using the CuScanColorizer and Matlab were compared in this study. The reference colors of the points calculated with the Matlab script (Fig. 5) were compared with the color point cloud produced by the CuScanColorizer (Fig. 6). White points are not colorized, because they are found outside of the pixel footprint. The result is correct – the points found inside of the green pixel are colorized, all very close, but points found outside are not colorized.

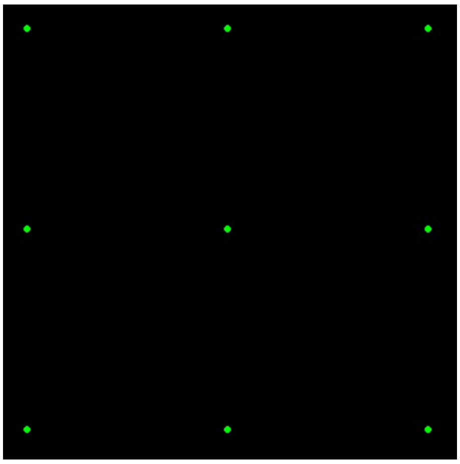

Fig. 5. Matlab reference point cloud for the green pixel.

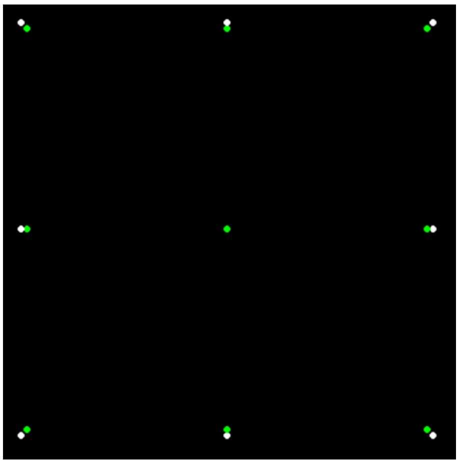

Fig. 6. Resulting point cloud - green color is correctly assigned to the right points.

Tests using synthetic data were performed for 5 sets of external orientation parameters. Table 1 shows all EO cases. Different configurations of omega, phi, and kappa angles were tested.

In addition, in order to test if the algorithm is resistant to erroneous input data, the point cloud was placed on the opposite side of the projection center (test C).

Table 1. EO parameters of 5 tests performed using synthetic data.

Test ID	Angular EO (omega, phi, kappa) [deg.]
A	0, 0, 0
B	10, -10, 10
C	10, -10, 10*)
D	80, -120, 270
E	-365, -10, 270

*) This test used a point cloud located erroneously on the opposite side of the center of projection.

The sixth test used natural data in the form of an image captured using a semimetric, calibrated camera – a Nikon D5200 with a 20mm lens and a point cloud (Faro Focus 3D). A reference test image produced with Matlab was compared with a test image produced by the CuScanColorizer. A partial view of the test field is shown in Figure 7, and the resulting image (identical for both algorithms) is shown in Figure 8.

The conclusion here is that the algorithm used by the CuScanColorizer works correctly with use of nearest neighbor interpolation.

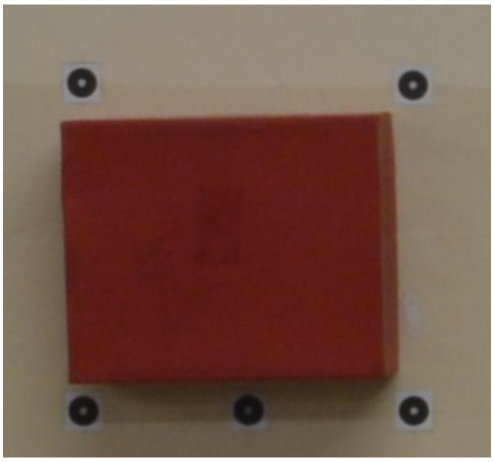

Fig. 7. Partial view of the image used for the natural data test.

Fig. 8. Colorized point cloud – the result was identical for both algorithms.

4.2. Testing procedure for interpolation methods

The second test of algorithm correctness used in this study checked the interpolation method used. Three methods of interpolation were used by the CuScanColorizer, and in this step, both the bilinear and bicubic methods were checked. The nearest neighbor was checked earlier during a test of the procedure of color assignment.

The first step of the test used synthetic data. An image of 100x100 pixels (Fig. 9) with RGB dataoverlapped an 80x80 point cloud by a certain margin. The position of the points inside of the corresponding pixel was randomized. The results of both interpolation methods were correct. The point cloud from the Matlab reference algorithm and that from the CuScanColorizer were identical for both interpolations. One example of these four resulting point clouds is shown in Figure 10.

The result of the test for bilinear and bicubic interpolation method using synthetic data was successful.

The second test performed on natural data was conducted with use of the same data as before. The obtained colors of specific points in both clouds (Matlab, CuScanColorizer) were identical for both test sessions for the bilinear and bicubic interpolation methods.

5. ACCURACY ISSUES WITH POINT CLOUD COLORIZATION

Preparations for the testing of the correctness of the algorithm served as an inspiration for more detailed studies of the problems of accuracy of colorization of point clouds as one option of photogrammetric and laser scanning data integration. Having a tested and rigorous algorithm served as a good starting point for a discussion of different aspects of integration accuracy and associated factors of accuracy of the process.

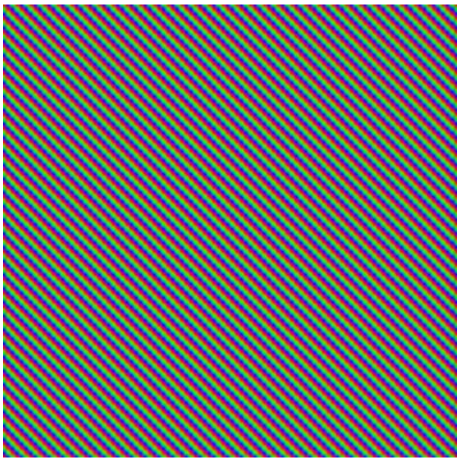

Fig. 9. Test image

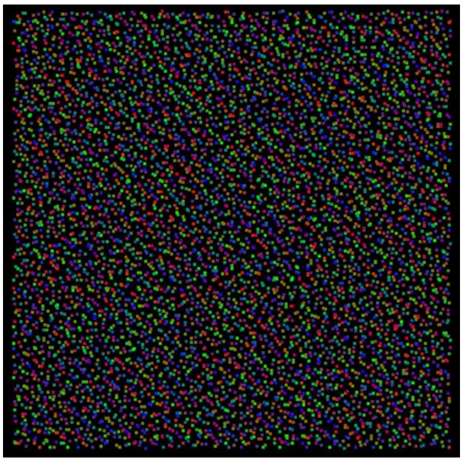

Fig. 10. Colorized point cloud from the CuScanColorizer using the bilinear method

.

There are three main factors of accuracy dependent on the parameters of the input data supplied for the purpose of the integration process: (1) accuracy of EO, (2) ratio of image GSD and point cloud density, and (3) relative angle between the main plane of the point cloud and the image frame.

The effects of these factors were evaluated for all three interpolation methods. In order to compare the reference image and resulting image, the RGB spatial distance (ΔRGB) was calculated (Stal *et al.*, 2011) using Equation (1):

$$\Delta RGB = \sqrt{(R - R_W)^2 + (G - G_W)^2 + (B - B_W)^2} \tag{1}$$

where:

ΔRGB – RGB spatial distance in color cube,
R, G, B – RGB components of the specific point,
R_W, G_W, B_W – reference RGB components of the point.

ΔRGB is used for further analysis as color deviation V (Pleskacz, 2014). RMSE is calculated according to Equation (2):

$$RMSE\_RGB = \sqrt{\frac{\sum V^2}{n}} \tag{2}$$

where:

RMSE RGB – random mean square error of color assignment,
V – equal ΔRGB,
n – number of points.

5.1. EO accuracy factor

All three interpolation methods were tested in this study. The results of these tests are presented on six different graphs. The first three graphs present the results of testing with the use of synthetic data (Figure 11). Linear and angular errors were calculated as a part of GSD (range from -1 to 1). Figure 11a shows RMSE RGB curves as functions of errors of linear EO. Figure 11b shows RMSE RGB curves as functions of errors of angular EO. Figure 11c shows the effect of errors of all EO.

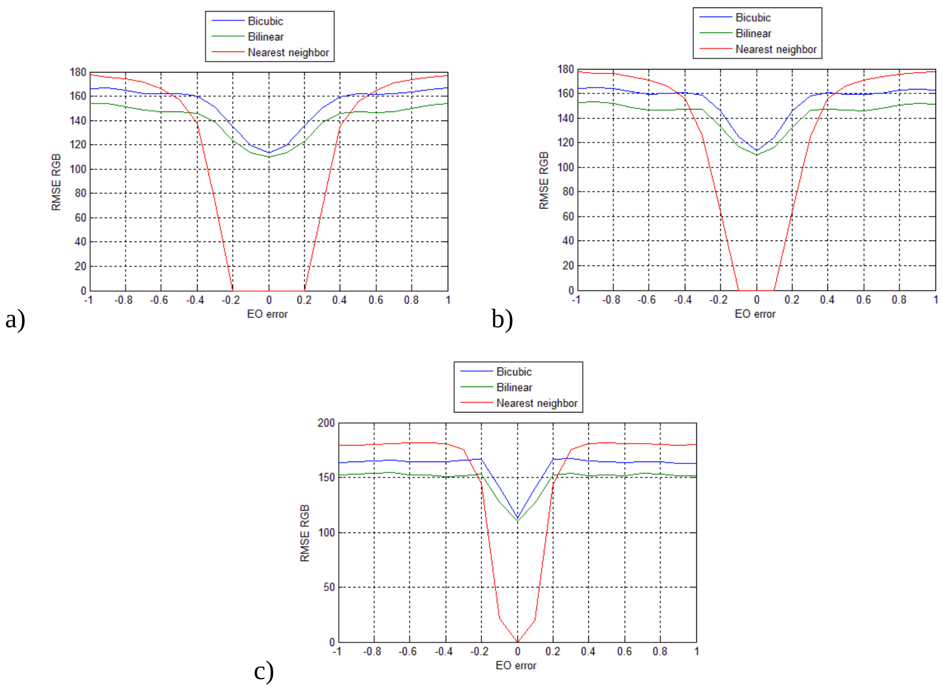

Fig. 11. Effect of EO on colorization accuracy for synthetic data: a) errors of linear EO, b) errors of angular EO, c) errors of linear and angular EO.

The same types of tests were used to evaluate natural data. Real EO was disturbed to obtain errors ranging from -1 GSD to 1 GSD. Three graphs in Figure 12 are used to show additional results, as follows: for linear errors, for angular errors, for all EO errors.

The tendencies identified for synthetic data and natural data were very similar. The only significant difference between results obtained for natural and synthetic data is that the errors for natural data are smaller than errors for synthetic data. The difference is caused by random distribution of the colors in synthetic data and similar colors of neighboring pixels in natural data. Linear EO errors exert more influence on RGB errors produced using the bilinear and bicubic methods. The nearest neighbor method is better for small EO errors than the other two methods used in the study. The obtained bilinear curve is the smoothest, but an RGB error is possible when using the nearest neighbor method.

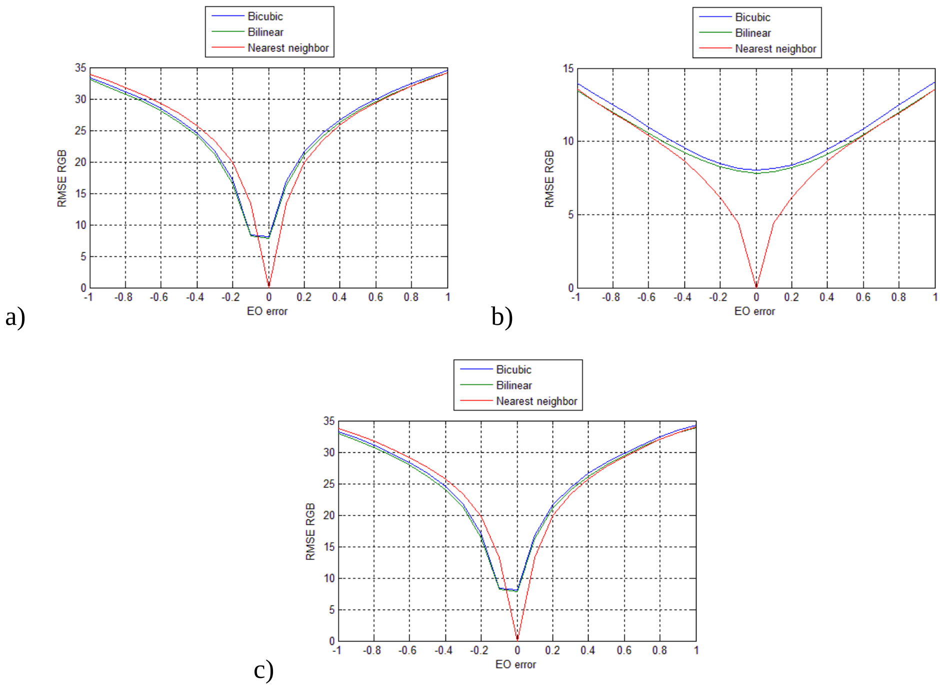

Fig. 12. Effects of EO on colorization accuracy for natural data: a) errors of linear EO, b) errors of angular EO, c) errors of both linear and angular EO.

5.2. Factor of GSD / density ratio

Another important factor of point cloud colorization is the ratio of the GSD of an image to the density of the cloud to be colorized. The goal of this test is to define a range of the ratio, thus guaranteeing a sufficient level of accuracy of the data colorization process. In order to define the technical conditions of the given data, the following test was performed:

The test image had 100 rows and columns of white pixels; in the central black square: 10x10 pixels. The test scan included a total of 14,400 points. Each pixel overlapped 16 points of the test cloud with a GSD/density ratio equal to 1. The central part of the test image of 30x30 pixels (800 white and 100 black) overlapped 14,400 points of the test cloud. Changing the height of the image (external orientation) changes the ratio, and the mean color of the central 1,600 points is also changed. With increasing height, the ratio increases, and more points are colored. For each ratio, RMSE is calculated for all 14,400 points. The reference mean color is calculated for a ratio of 1 – a total of 1,600 black points and 12,800 white points.

The test was performed for all three interpolation methods. The results are presented in Figure 13. The nearest neighbor method produced the lowest reference 0 value for a ratio of 1. In most cases, the bilinear method yields the lowest values. It may be observed that a ratio of 1.5 yields a fairly similar RMSE compared with a ratio close to 0.

The graphs provided in this paper for all the interpolation methods show that even a small difference between an image's GSD and point cloud density will produce a large error in the colorization process.

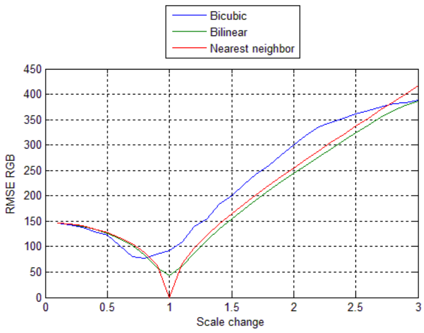

Fig. 13. GSD/density test results.

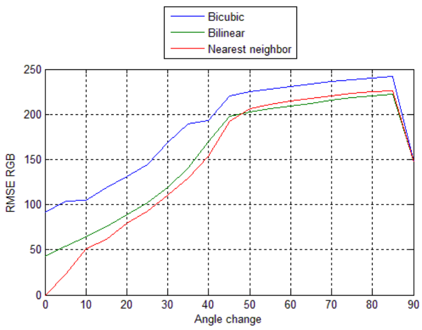

Fig. 14. Angle test results.

5.3. Testing the relative angle factor

The same data as above were used to test the influence of the factor of relative angle between the image's plane and the scanned plane. The reference data were colorized for the 0 angle with a GSD/density ratio equal to 1. The position of the camera was changed by 5 degree steps of the phi angle, and increasingly more white points were incorrectly colorized by black squares. For each position, colorization was performed, and RMSE was calculated.

The results are presented in Figure 14.

The lowest RMSE values were obtained with the use of nearest neighbor interpolation. Errors increased almost linearly up to an angle of 45 degrees. Results over 45 degrees show that the test black square is overlapping a larger area than the test point cloud. Values over 45 degrees for the relative angle did not permit point cloud colorization. The lowest values were obtained using the nearest neighbor method.

6. CONCLUSIONS

The paper presents the LiDAR data colorization process as a data integration process. An innovative algorithm based on parallel processing using graphic cards was tested with the use of the presented testing procedure. The test consisted of two steps. The first step checked if point colorization has been performed correctly, while the second step analyzed the use of three different interpolation methods. Color assignment and implementation of interpolations were conducted using natural and synthetic data. Both tests confirmed the correctness of the colorization process with different data configurations and sources.

The second part of the paper discussed the broader problem of data integration accuracy via the example of LiDAR data colorization with RGB color for photogrammetric data. There are three different factors that have to be taken into account. The first factor is the EO accuracy of a the image. The second factor is the ratio of point cloud density and image GSD. The third factor is the relative angle between the image plane and the main plane of the point cloud. The discussion of these issues is provided in the paper, but the most important is the presentation of the importance of these issues as conditions for correct data integration.

There exists one more aspect of colorization not mentioned in this paper, because it was not within the scope of the study. The color in real data processing is calculated from different images overlapping a specific point on a given scan, and the color should be calculated as a weighted mean. The problem of adequate weighting of the color is another potential area of future research.

ACKNOWLEDGEMENTS

Preparation of this research work was possible thanks to support provided by the Management Board of DEPHOS Software sp. z o.o., for which the authors express their sincere thanks. Research financed under the statutory research at AGH UST no. 11.11.150.949

LITERATURE

Będkowski J., Bratuś R., Prochaska M., Rzonca A., 2015. Use of parallel computing in mass processing of laser data. *Archiwum Fotogrametrii, Kartografii i Teledetekcji*, 27, pp. 45-59.

Cramer M., 2011. Geometry Perfect – Radiometry Unknown? *Proceedings, Photogrammetric Week 11'*, Wichmann Verlag, Heidelberg, pp. 65-77.

Hoehle J., Potackova M., 2005. Automated Quality Control for Orthoimages and DEMs. *Photogrammetric Engineering & Remote Sensing*, 71, pp. 81-87.

Getreuer P., 2011. Linear Methods for Image Interpolation, *Image Processing On Line* o www.ipol.im/pub/art/2011/g_lmii/

Jun P., Mi W., Deren L., Junli L., 2010. A Network-Based Radiometric Equalization Approach for Digital Aerial Orthoimages, *IEEE Geoscience and Remote Sensing Letters*, 7(2), pp. 401-405.

Markelin L., Honkavaara E., Peltoniemi J., Ahokas E., Kuittinen R., Hyyppa J., Suomalainen J., Kukko A., 2008. Radiometric Calibration and Characterization of Large-

Format Digital Photogrammetric Sensors in a Test Field. *Photogrammetric Engineering & Remote Sensing*, 74, pp. 1487-1500.

Nobrega R.A., Quintanilha J.A., 2004. Comparative analysis of automatic digital image balancing and standard histogram enhancement techniques in remote sensing imagery. *Revista Brasiliera de Cartografia*, 56/01, 55-64.

Pirowski T., 2009. Ranking metod integracji obrazów teledetekcyjnych o różnej rozdzielczości – ocean formalna scalenia danych Landsat TM I IRS-PAN. *Archiwum Fotogrametrii, Kartografii i Teledetekcji*, 20, pp. 343-358.

Pleskacz M., 2014. Radiometric quality analysis of colored point cloud, M.Sc. thesis, AGH Kraków 2014.

Pyka K., 2009. Jak ocenić jakość fotometryczną ortofotomapy? *Archiwum Fotogrametrii, Kartografii i Teledetekcji*, 19, pp. 363-372

Pyka K., 2013. Mozaikowanie ortoobrazów z zastosowaniem transformacji Falkowej, Monography, Wydawnictwo AGH, Kraków, 2013.

Rzonca A., 2013 - Quality Control System Preparation For Photogrammetric And Laser Scanning Missions of Spanish National Plan of Aerial Orthophotography (PNOA), *Archiwum Fotogrametrii, Kartografii i Teledetekcji*, 25, pp. 189-201.

Rzonca A., 2013 - *Integration of photogrammetric and 3D laser scanning data for monumental objects documentation*, Wydawnictwo AGH, Kraków 2013.

Stal C., De Maeyer P., De Ryck M., De Wulf A., Goossens R., Nuttens T., 2011. Comparison of geometric and radiometric information from photogrammetry and color-enriched laser scanning, *FIG Working Week 2011*, Bridging the Gap between Cultures, Marrakech, Morocco, 2011.

OPRACOWANIE METODY OCENY POPRAWNOŚCI DZIAŁANIA ALGORYTMU KOLOROWANIA CHMURY PUNKTÓW

SŁOWA KLUCZOWE: integracja danych, testowanie algorytmów, jakość danych, koloryzacja skanów

Streszczenie

Publikacja omawia opracowanie metody oceny poprawności działania algorytmu służącego do przypisania składowych RGB punktom chmury pochodzącej ze skaningu laserowego. Metoda testowania tego algorytmu jest przedstawiona w kontekście problemu kontroli merytorycznej algorytmów do przetwarzania danych przestrzennych. Proces kolorowania traktowany jest jako jeden z przypadków integracji danych skaningowych i fotogrametrycznych. W ramach wprowadzenia teoretycznego autorzy omawiają problemy badawcze, które wynikają z potrzeby sprawdzenia poprawności oraz dokładności procesu kolorowania. Podane są kryteria, według których można określić, czy badany algorytm jest poprawny pod względem merytorycznym: czy kolorowane są odpowiednie piksele i czy metody interpolacji są zastosowane prawidłowo. Następnie określony jest wpływ dokładności elementów orientacji zewnętrznej oraz rozmiaru piksela terenowego zdjęć na poprawne kolorowanie. Na koniec omówiono problem nierównoległości płaszczyzny tłowej do powierzchni chmury punktów, co też może mieć wpływ na jakość kolorowania.

Po rozważaniach teoretycznych opisane zostały metody testowania poprawności przyporządkowania punktom koloru oraz poprawności implementacji algorytmów interpolacji. Obie metody zastosowane są na danych syntetycznych oraz na rzeczywistych danych pomiarowych. Następnie dyskutowane są inne czynniki, niezależne od poprawności algorytmu kolorowania, wpływające na dokładność kolorowania chmury punktów. Pierwszy czynnik to dokładność elementów orientacji zewnętrznej fotogramu, który służy do kolorowania. Kolejnym czynnikiem jest różnica pomiędzy rozdzielczością terenową fotogramu i kolorowanej chmury punktów. Trzecim czynnikiem jest kąt pomiędzy kolorowana powierzchnią chmury punktów a płaszczyzną tłową fotogramu. Badanie algorytmu zostaje rozszerzone o podanie ogólnych zasad dotyczących parametrów technicznych danych integrowanych w ramach omawianego procesu w zakresie powyższych trzech czynników. Badanym, przykładowym algorytmem jest CuScanColorizer - innowacyjny algorytm firmy DEPHOS Software, który wykonuje kolorowanie chmury punktów, wykorzystując do tego metodę przetwarzania równoległego na procesorach graficznych opartą na technologii nVidia CUDA. W podsumowaniu podane są wyniki zastosowania metody kontroli poprawności algorytmu wraz z oceną przykładowego, badanego algorytmu oraz wskazaniem parametrów optymalnych z punktu widzenia stosowania procesu kolorowania chmury. Jako dodatkową konkluzję zawarto ocenę poprawności algorytmu CuScanColorizer.

Dane autorów / Authors details:

Mgr inż. Michał Pleskacz
e-mail: michal.pleskacz90@gmail.com
telefon: 792692454

Dr inż. Antoni Rzonca
e-mail: arz@agh.edu.pl
telefon: 505103540

Przesłano / Submitted 26.10.2016
Zaakceptowano /Accepted 30.12.2016