A NEW CONTROL POINTS BASED GEOMETRIC CORRECTION ALGORITHM FOR AIRBORNE PUSH BROOM SCANNER IMAGES WITHOUT ON-BOARD DATA

P. Strakhov [a], E. Badasen [a], B. Shurygin [a], T. Kondranin [a]

[a] Dept. of Aerophysics and Space Research, MIPT, Dolgoprudny, Russia - pavel.strakhov@phystech.edu

Commission I, WG I/4

KEY WORDS: geometric correction, push broom sensors, hyperspectral imaging, airborne imaging, ground control point (GCP)

ABSTRACT:

Push broom scanners, such as video spectrometers (also called hyperspectral sensors), are widely used in the present. Usage of scanned images requires accurate geometric correction, which becomes complicated when imaging platform is airborne. This work contains detailed description of a new algorithm developed for processing of such images. The algorithm requires only user provided control points and is able to correct distortions caused by yaw, flight speed and height changes. It was tested on two series of airborne images and yielded RMS error values on the order of 7 meters (3-6 source image pixels) as compared to 13 meters for polynomial-based correction.

1. INTRODUCTION

Geometric correction is an important part of airborne push broom images processing. Airborne systems provide higher spatial resolution and immediacy as compared to space imaging, while push broom scanners allow to achieve high spectral resolution. However, these advantages come with a price. These images always have significant geometric distortions. Despite widely applied usage of gyro-stabilized platforms, achieving uniform camera motion during airborne scanning seems to be extremely difficult. Push broom scanners observe one image line at a time, resulting in shifts of relative positions of scanned areas. Before further processing of these images it is necessary to perform geometric correction, i.e. compensate for geometric distortions and make possible using these images to create topographic maps.

Existing approaches to this problem can be divided into two categories: empirical mathematical distortion models and physical imaging process models (Toutin, 2004). Common instances of empirical models are 2D and 3D polynomial functions (Wang, 2011a; Belozyorskiy, 2010; Chaban, 2012), rational functions (Bagheri, 2014; Titarov, 2004), rubber band model (Kochub, 2012; Devereux, 1990), continuous piecewise model (Mlnhe, 2000), thin plate splines (Chan, 2010). All these approaches have a common limitation: they require even and dense distribution of control points at the image, because these models can only compensate distortions locally (Toutin, 2004). If the image does not have enough landmarks in some areas, empirical models cannot guarantee any positive correction effect in those areas.

In practice, empirical models often behave unpredictably during control points input. A seemingly correct set of control points can produce totally incorrect transformation, while random subtle changes of control points might change the picture significantly for no apparent reason. Therefore the user is sometimes forced to guess which control points' positioning provides the best result without any feedback they can objectively analyze. An empirical model that can describe intense and complicated distortions introduces a chance that these kinds of distortions will randomly appear when it is not desired. For instance, higher order polynomial models are

practically inapplicable because of randomness of the result (Toutin, 2004). As expected, second order polynomial functions behave much more predictably and are usable in simple cases, but are not able to account for complex distortions.

A compromise option is to divide an image to multiple fragments containing distortions that can be described by a simple empirical model (Wang, 2011b). However, it is not always possible and often requires large amount of work.

Physical models vary more significantly because they have to accommodate to imaging processes they are applied to and additional data sets available. For example, GPS or IMU (inertial measurement unit) data are available in many cases, although many works admit that geometric correction using this data without addition of user specified control points leads to improper result accuracy because of initial data inaccuracy (Roy, 1997; Jensen, 2011; Wang, 2012; Gusev, 2014; Balter, 2007; Liu, 2014). If on-board data is not used, geometric correction algorithms can calculate physical model parameters to minimize root-mean-square error (RMSE) of user specified control points (Reguera-Salgado, 2012). Other alternative approaches include shifting subsequent image lines to minimize difference in brightness (Vasileyskiy, 2005), using synchronized video imaging to calculate shift between lines (Ilin, 2012), using user specified control lines that determine position of multiple image lines (Jensen, 2011; Nikishin, 2011). Applicability of any physical model is limited by assumptions made about imaging process when designing that model. However, such models usually do not have the aforementioned disadvantages of empirical models and can provide more predictable and accurate result if the model fits reality well and its parameters are calculated accurately enough.

The goal of this work was to create an algorithm that can be applicable to airborne push broom scanner images without using GPS and other on-board data. The control points provided by users were expected to be used as main data source, although the algorithm should provide good accuracy even in image fragments with little to no control points. The algorithm also needs to be convenient and predictable to users and should not require large amount of control points to work.

2. TEST DATA SETS

All images used for developing and testing the algorithm were obtained using a push broom video spectrometer manufactured by ZAO "NPO "Lepton", Russia. This camera has a spatial resolution of 0.35 m at 1000 m distance, angular resolution of 1.2', swath width of 175 m at 1000 m distance, 12-bit ADC resolution, and 290 spectral bands. A series of images was taken in May, 2014 near Plavsk town, Russia, at aircraft altitude of 2000 m (see image 1). Acquired images have size of 500 by 4000-6000 pixels. A stabilizing platform was used to reduce some geometric distortions.

Image 1. Selected test image from first series, rotated by 90° counter-clockwise.

This series has the following features:

• Yaw changes introduce significant distortions that complicate using general empirical models and may require splitting each image to pieces for geometric correction to succeed;
• The imaged area contains few visible landmarks, not enough to distribute control points evenly;
• Some parts of the imaged area do not have any suitable landmarks, making control points based correction hardly possible;
• Acquired on-board data is not complete and accurate enough to use it for geometrical correction.

On the other side, the surface is almost flat, and flight height introduces relatively minor distortions compared to yaw-related distortions. These factors allowed to construct relatively simple physical model of imaging process to compensate observed distortions.

Another series of airborne images was also used to see how the algorithm works under different conditions. These images were obtained in July, 2015 at Leningrad Oblast, Russia with two cameras of the same model. However, in contrast to the first series, these two identical cameras were simultaneously used with 16° angle between the cameras and 8° off-nadir angle of each camera. Flight height was in range of 4000-6000 m. A selected test image is depictured at image 2.

Image 2. Selected test image from second series, rotated by 90° counter-clockwise.

Using control points requires presence of reference map with assumed absence of geometric distortions. Georeferenced satellite images available online were used for this purpose. Points at these images are identified by their coordinates (x_r, y_r) defined by coordinate system of their geographic projection. This allows estimating correction errors in projection units (in this case, meters).

3. ALGORITHM DESCRIPTION

Geometric correction involves calculating coordinates of each pixel of original image in coordinate system of the reference map. The described algorithm uses information about imaging process and control point location to construct the target coordinate transformation.

The following assumptions related to imaging process and input data were made to simplify the algorithm:

1. Relief does not introduce significant geometric distortions.
2. Flight height changes smoothly.
3. Roll and pitch do not change significantly.
4. Imaging of all pixels of a line is done at the same moment.
5. Control points are set with the most possible precision.

The transformation maps each image line to a line segment on the reference map. 4 parameters are enough to specify the position of the line segment (x_r, y_r, l, a): coordinates of the center of the segment in reference map coordinate system (x_r, y_r), segment length l, and counter-clockwise rotation angle a. Zero rotation angle represents a horizontal segment with motion speed vector directed upwards in the image's coordinate system.

Each pair of control points is determined by their coordinates on the reference map (x_r, y_r) and on the target image (x_t, y_t). y_t coordinate is the image line number and is interpreted as integer. x_a equals to distance between left edge of the image line and the control point divided by the line length, i.e. $x_a = 0$ represents a control point at the left edge of the image, and $x_a = 1$ represents a control point at the right edge. Each control point provides additional data about position of y_t line segment. More specifically, the point that divides the segment to x_a and $1 - x_a$ parts is fixed at (x_r, y_r) position, leaving only two parameters unknown − (l, a). Center of the line segment can be calculated based on these parameters and the control point data. Placing two control points at the same y_t line allows calculating the precise location of the according line segment. Placing more than two control points at the same line is not allowed, as additional points cannot provide new data to the algorithm and it would not be able to take into account all specified positions. Image 3a depictures described line segments before and after transformation. Black points mark segment centers, while white ones mark control points, or segment rotation centers.
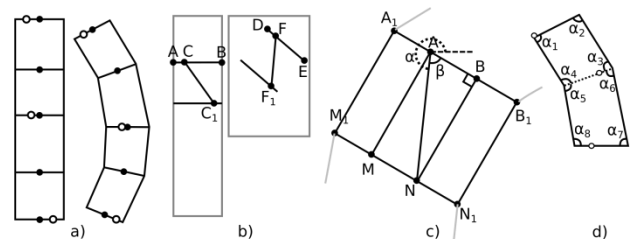

Image 3. Illustrations for algorithm description.

The user can also specify fixed l and/or a values for each line segment, i.e. determine its length or rotation.

The algorithm starts with searching for image lines containing control points. Found lines are arranged in order of their imaging time.

The next step is calculating segment lengths l. It is done by consequent analysis of pairs of found lines and their according control points data. Image 3b displays target image (left), reference map (right), and two pairs of control points (C, F) and (C_1, F_1) with coordinates $(x_{t1}, y_{t1}, x_{r1}, y_{r1})$ and

$(x_{t2}, y_{t2}, x_{r2}, y_{r2})$, where index 1 denotes the line created earlier, and index 2 denotes its subsequent line, index t means target image, index r means reference map. A pair is excluded from processing if the user has fixed length of any of two lines. Otherwise, a pair is used to calculate the length if $y_{t2} - y_{t1} < p_{dw}$, where p_{dw} is a constant algorithm parameter that is determined experimentally and is measured in image pixels (this work uses value of $p_{dw} = 20$). In other words, the frame numbers (or image line numbers) of two control points must be close enough. Target image width and height (frame count) are denoted as w_t and h_t pixels. The image is assumed to have no distortions in vicinity to the current segments pair, allowing to calculate length of transformed line segment based on length of original line and known distances between control points:

$$w_{r1,2} = |DE| = |AB| \frac{|FF_1|}{|CC_1|} = w_t \frac{\sqrt{(x_{r1} - x_{r2})^2 + (y_{r1} - y_{r2})^2}}{\sqrt{(x_{t1} - x_{t2})^2 + (y_{t1} - y_{t2})^2}}$$

Calculated length value is saved for both involved line segments. Lengths specified by the user are also assigned at this stage. If no lengths are specified by user and there are no pairs suitable for length detection, the algorithm stops with an error message.

Next, the line segment length is calculated for the rest of lines containing control points. If a line lies between two segments with determined length, linear interpolation is used to calculate its length, i.e. segment length is assumed to depend linearly on frame number. If all segments with originally determined length are positioned to one side of current line (e. g. the line is at the very beginning or end of image), the length of the closest segment is assigned to the current line. Each line segment has a determined w_r value after this step.

The next step involves calculation of preferred rotation angle for each segment pair. Calculation is done in reference map coordinate system. The length of the second segment in a pair is assumed to be equal to the first one to simplify calculations. Length difference will be accounted for in the next step. Each segment can be rotated around the control point it contains. The segments can be interpreted as opposite sides of a rectangle when rotated by desired angle. The line segments are denoted as A_1B_1 and M_1N_1, while the control points are denoted as $A \in A_1B_1$ and $N \in M_1N_1$ (image 3c). $A_1B_1N_1M_1$ is a rectangle. Let $A_1A \le M_1N$. Let's drop perpendiculars AM and NB on M_1N_1 and A_1B_1, constructing ABNM rectangle. $|AN| = \sqrt{(x_{r1} - x_{r2})^2 + (y_{r1} - y_{r2})^2}$. $|AB| = w_r |x_{a1} - x_{a2}|$. Next, $\angle BAN = \beta = \cos^{-1} \frac{|AB|}{|AN|}$. Rotation of AN relative to zero position, i.e. $(1, 0)$ vector, can be calculated as $\alpha = \text{atan2}(y_{r2} - y_{r1}, x_{r2} - x_{r1})$. Final AB rotation is $\gamma = \alpha + \beta$. The $A_1A > M_1N$ case can be processed similarly. In this case $\gamma = \alpha + \pi - \beta$, where α and β are calculated with the same formulas.

Each line segment, excluding the first and the last ones, has two rotation angles calculated for it because it is included in two pairs. Starting angle of such segment is calculated as weighted arithmetic mean of the two angles, where weight of an angle is inversely proportional to frame count between segments of according pair:

$$w_{r,i} = \frac{w_{r,i,i+1}}{y_{t,i+1} - y_{t,i}} + \frac{w_{r,i-1,i}}{y_{t,i} - y_{t,i-1}}$$

Thus the closer the neighbor segment is positioned to the current segment, the bigger its impact on its starting angle.

Calculated starting rotation angles leave segment length difference and their relative positions unaccounted for. The next step is iterative optimization of angles while taking these new factors into account. On each iteration, for each line segment two alternative rotation angles are considered, whose rotations differ from current position by $\Delta\gamma$ clockwise and counter-clockwise (this work uses value $\Delta\gamma = 0,3°$). The quadrangles that are formed by current line segment, neighbor line segments, and image borders are analyzed (see image 3d, current segment is dashed). The position that minimizes the following metric value is selected:

$$m = \sum_{i=1}^{8} |\alpha_i - 90°|,$$

i.e. quadrangles are desired to be rectangles, which indicates uniform platform motion at the image fragment between control points. At the end of each iteration selected positions are applied to all segments. System stabilization is determined by aggregate (signed) rotation difference for all segments for the last 10 iterations. If this value is less than $2\Delta\gamma$, it means that the system is stabilized and the iterations should be stopped. No system stabilization after a reasonably large number of iterations indicates a control point positioning error or incorrect length detection, which results in wrong quadrangle orientations.

Calculated rotations for line segments containing control points are final. These rotations, along with lengths, are used to calculate segment midpoint coordinates. Linear interpolation is then used to calculate midpoint coordinates, lengths, and rotation angles of the line segments which do not contain control points. If a segment does not have determined line segments before and after it (i.e. if a segment is positioned before the first line with a control point or after the last line with a control point), the rotation and length of the closest determined segment is assigned to it, and its midpoint is calculated by linear interpolation of midpoints of two closest segments containing control points, despite they are positioned to one side of it. This method is chosen because uniform motion is assumed when there are no control points in an image fragment.

The original algorithm could not be applied to the second series of images because it could not compensate for off-nadir angle presence. An additional preliminary correction procedure was introduced to compensate these distortions. Imaging process geometry was analyzed to determine how pixel coordinates should be shifted to emulate nadir viewing. Coordinates of Y axis along track remain unmodified, while X axis across track endures a coordinate transformation. Original coordinates x varying in $[0, x_{max}]$ range are mapped to corrected image coordinates x' varying in $[0, x'_{max}]$ using the following equation:

$$x' = \frac{k x'_{max} x}{x_{max} - x + kx}$$

where $k = \frac{\sin\gamma_2}{\sin\gamma_1}$ is calculated from $\gamma_{1,2}$ angles, created by the imaged line segment and lines connecting the camera and its ends. $\gamma_{1,2}$ angles can be calculated from viewing angle and field of view of the camera.

4. ALGORITHM TESTING METHOD

An implementation of the algorithm was created using hyperspectral image processing software "Albedo" to investigate effectiveness and applicability of the algorithm. Images from both series were corrected using described algorithm. First series processing results were combined in a mosaic image.

Geometric correction accuracy was assessed using RMSE values of separate set of check points specified by user. Check points were not used to calculate transformations. The accuracy was calculated in two variants. The distances between reference map check points and paired check points mapped from target image to reference map produce the accuracy measured in meters. The distances between target image check points and paired check points mapped from reference map to target image produce the accuracy measured in target image pixels.

An independent geometric correction using classic 2D 2nd order polynomial model was also applied to both series in order to compare algorithms and assess provided advantages. The same sets of check points were used for both approaches.

5. RESULTS

The selected test image from the first series (image 1) was corrected with new algorithm using 38 pairs or control points. 23 pairs of check points were also added to assess correction accuracy. Image 4 depictures the result of geometric correction combined with a reference map. Image 5 displays the mosaic image constructed from all images of the series using described algorithm. The correction accuracy was estimated to be equal to 7.0 m or 5.7 pixels of original image. The accuracy varies from 1-3 m in areas containing multiple landmarks to 5-10 m in areas lacking landmarks. Polynomial correction of the test image did not allow achieving any comparable accuracy because of significant distortions which cannot be described by polynomial model.

The selected test image from the second series (image 2) was corrected with new algorithm using 34 pairs or control points, and with polynomial model using 17 pairs of points. 42 pairs of check points were also added to assess correction accuracy. RMSE of described algorithm result was equal to 7.3 m or 3.5 pixels, while polynomial model produced 13.1 m or 5.0 pixels error. Results of applying both models are depictured at images 6-7. Corresponding reference maps are used as backgrounds at images 4-7.
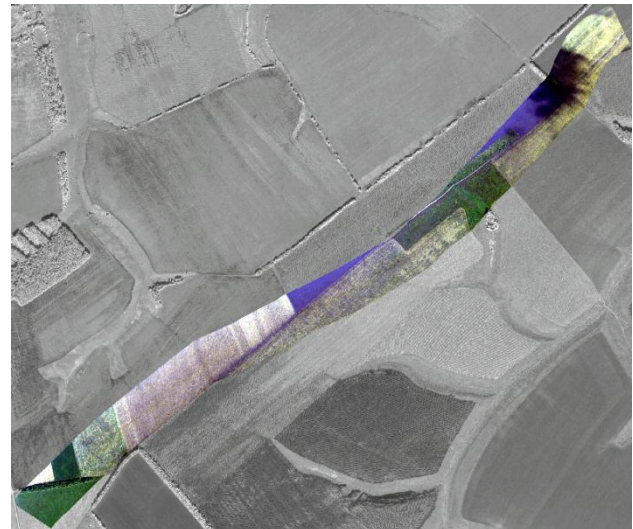

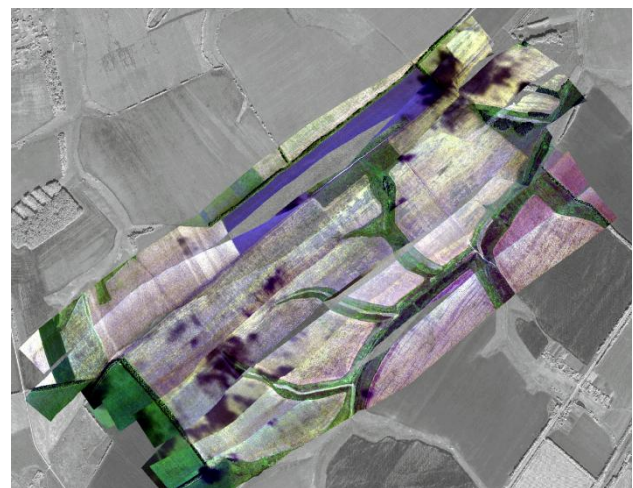Image 4. A test image from the first series corrected with described algorithm.

Image 5. All corrected images from the first series combined into a mosaic image.

Image 6. A test image from the second series corrected with described algorithm.

Image 7. A test image from the second series corrected with polynomial model.

6. DISCUSSION

Results have shown that the algorithm is capable of geometric correction of test image sets. The algorithm appears to be predictable and comfortable for users. There was no need to cut images to fragments or move control points randomly to achieve better results. Amount of control points needed is comparable to that required by other methods. Areas without landmarks are corrected with lower, but still acceptable accuracy. Successful construction of mosaic images with little artifacts at image borders also demonstrates accurate enough mapping of target image borders to the reference map. Most of remaining artifacts are caused by brightness difference and can be further corrected if needed. Another promising approach is using tie points for correcting adjacent images. However, it would require further research.

Desired accuracy of geometric correction is about 1 image pixel. However, multiple factors such as reference map accuracy, control points and on-board accuracy, image quality are often limiting achievable final accuracy. Multiple studies about airborne push broom images processing have reported results similar to this work. For instance, in (Luan, 2014) the accuracy of 2 pixels across track was achieved, and in (Liu, 2014) the achieved accuracy was in 2-6 pixels range. The accuracy achieved in this work is enough for some thematic maps to be useful, although improving accuracy is still an important task.

Visual analysis of corrected images allows to conclude that the algorithm successfully corrects distortions included into its model, such as yaw, movement speed and flight height changes. However, compensating these most significant factors revealed presence of other distortion sources, most notably the relief and slight variations of camera orientation. The algorithm is currently unable to deal with these distortions, which is an expected consequence of using a simplified model. Therefore, modifying the model to include more distortion sources is required to improve transferability and accuracy of the algorithm. Processing the second series also showed that including off-nadir angle into the model is reasonable.

The parameters optimization part of the algorithm also needs further improvement to search for globally optimal parameters rather than calculating of the most desirable pair orientations and searching for locally optimal orientation. The possibility of

control point position error presence should be introduced because it might allow providing the closest possible valid transformation given erroneous points rather than providing totally incorrect one.

In current state, a piecewise linear motion of camera is assumed, which may not be the best model of real motion, although it is indeed the simplest one. The possibility to use other kind of interpolation for calculating line segment centers should also be considered. For example, the approach shown in (Wang, 2012) might provide better results.

7. CONCLUSION

This work describes an algorithm developed for geometric correction of airborne push broom scanner images, such as video spectrometers. The algorithm uses a simplified physical model of camera motion. User provided control points are used as the only data source for the algorithm. It has been proven to be applicable to correct significant geometric distortions present on test image sets, even in areas lacking landmarks. Use of the algorithm allowed to achieve practically useful levels of accuracy and simplified the user's workflow by improving predictability and eliminating need to cut image to fragments. The most promising approaches to improve the algorithm have been determined based on received results analysis.

8. ACKNOWLEDGEMENTS

9. REFERENCES

Bagheri, H., Sadeghian, S., 2014. Geometric rectification of high resolution satellite images using mathematical intelligent & classical modeling. *Iranian Conference on Intelligent Systems*, 2014. IEEE, pp. 1–6.

Balter, B.M., Balter, D.B., Egorov, V.V., Belov, A.A., Voroncov, D.V., Orlov, A.G., 2007. Glubokaya obrabotka dannyh vertoletnogo giperspektrometra [Deep data processing of a helicopter hyperspectrometer]. *Sovremennye problemy distancionnogo zondirovaniya Zemli iz kosmosa [Current problems in remote sensing of the Earth from space]*, 2007, Vol. 1, pp. 197–208.

Belozerskiy, L.A., Murashko, N.I., Suschenya, D.S., 2010. Osobennosti polinomialnoy geometricheskoy korrekcii primenitelno k zadacham analiza izobrazheniy raznovremennoy kosmicheskoy semki [Features of Polynomial Geometric Correction Concerning to the Tasks of the Images Analysis Captured at Different Time Space Survey]. *Shtuchniy intellect [Artificial intelligence]*, 2010, Vol. 3, pp. 299–311.

Chaban, L.N., Vecheruk, G.V., Kondranin, T.V., Kudryavcev, S.V., Nikolenko, A.A., 2012. Modelirovanie i tematicheskaya obrabotka izobrazheniy, identichnyh videodannym s gotovyascheysya k zapusku i razrabatyvaemoy giperspektralnoy apparatury DZZ [Modelling and thematic processing of images identical to videodata of the hyperspectral remote sensing equipment being prepared for launch and developed]. *Sovremennye problemy distancionnogo zondirovaniya Zemli iz*

kosmosa [Current problems in remote sensing of the Earth from space], 2012, Vol. 9, M92, pp. 111–121.

Chan, J.C.-W., Ma, J., Kempeneers, P., Canters, F., 2010. Superresolution enhancement of hyperspectral CHRIS/Proba images with a thin-plate spline nonrigid transform model. *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 48, pp. 2569–2579.

Devereux, B.J., Fuller, R.M., Carter, L., Parsell, R.J., 1990. Geometric correction of airborne scanner imagery by matching Delaunay triangles. *International Journal of Remote Sensing*, Vol. 11, pp. 2237–2251.

Gusev, V.Yu., 2014. Metody i sredstva radiometricheskoy i geometricheskoy obrabotki skanovyh izobrazheniy zemnoy poverhnosti [Methods and means of radiometric and geometric processing of surface scanner images]. PhD dissertation. Moscow Aviation Institute (National Research University), Moscow, 2014.

Ilin, A.A., Vinogradov, A.N., Egorov, V.V., Kalinin, A.P., Rodionov, A.I., 2012. Metod geometricheskoy korrekcii giperspektralnyh izobrazheniy zemnoy poverhnosti [A geometric correction method of surface hyperspectral images]. *Sovremennye problemy distancionnogo zondirovaniya Zemli iz kosmosa [Current problems in remote sensing of the Earth from space]*, 2012, Vol. 9, M 1, pp. 39–46.

Jensen, R.R., Hardin, A.J., Hardin, P.J., Jensen, J.R., 2011. A New Method to Correct Pushbroom Hyperspectral Data Using Linear Features and Ground Control Points. *GIScience & Remote Sensing*, Vol. 48, pp. 416–431.

Kochub, E.V., Topaz, A.A., 2013. Analiz metodov obrabotki materialov distancionnogo zondirovaniya zemli [Surface remote sensing data processing methods analysis]. *Vestnik Polotskogo gosudarstvennogo universiteta [Polotsk State University Bulletin]*, Vol. F, 2012, pp. 132–140.

Liu, S., Wang, Z., Hao, W., Wang, R., 2014. On Geometric Correction Method of BJ-1 Panchromatic Image Covering Kingdom of Lesotho. *Asian Agricultural Research 06*, 2014, pp. 75-78, 84.

Luan, K., Tong, X., Ma, Y., Shu, R., Xu, W., Liu, X., 2014. Geometric Correction of PHI Hyperspectral Image without Ground Control Points. *IOP Conference Series: Earth and Environmental Science 17*, 012193.

Mlnhe, J.I., Jensen, J.R., 2000. Continuous piecewise geometric rectification for airborne multispectral scanner imagery. *Photogrammetric Engineering & Remote Sensing*, Vol. 66, pp. 163–171.

Nikishin, Yu.A., 2011. Razrabotka i issledovanie metodov geometricheskoy korrekcii i fotogrammetricheskoy obrabotki materialov vozdushnoy nestabilizirovannoy giperspektralnoy semki [Developing and research of geometric correction and photogrammetry processing methods for airborne unstabilized hyperspectral imaging]. PhD dissertation. Moscow State University of Geodesy and Cartography, Moscow, 2011.

Reguera-Salgado, J., Martín-Herrero, J., 2012. High performance GCP-based Particle Swarm Optimization of orthorectification of airborne pushbroom imagery. *Geoscience*

and Remote Sensing Symposium (IGARSS)*, 2012, pp. 4086–4089.

Roy, D.P., Devereux, B., Grainger, B., White, S.J., 1997. Parametric geometric correction of airborne thematic mapper imagery. *International Journal of Remote Sensing,* Vol. 18, pp. 1865–1887.

Titarov, P.S., 2004. Prakticheskie aspekty fotogrammetricheskoy obrabotki skanernyh kosmicheskih snimkov vysokogo razresheniya [Practical aspects of photogrammetry processing of high resolution space scanner imagery]. *Informacionnyy byulleten GIS-Associacii [GIS Association informational bulletin], 45.*

Toutin, T., 2004. Review article: Geometric processing of remote sensing images: models, algorithms and methods. *International Journal of Remote Sensing,* Vol. 25, pp. 1893–1924.

Vasileyskiy, A.S., 2005. Korrekciya geometricheskih iskazheniy videodannyh s cifrovyh aerosemochnyh kamer, vyzyvaemyh uglovymi kolebaniyami nositelya po krenu [Correction of dititval airborne cameras videodata geometric distortions caused by carrier roll variations]. *Sovremennye problemy distancionnogo zondirovaniya Zemli iz kosmosa [Current problems in remote sensing of the Earth from space],* 2005, Vol. 2.

Wang, C., Zhang, Y., Liu, P., Xu, Q., Gu, Y., 2011a. A self-adjustive geometric correction method for seriously oblique aero image. *IEEE International Geoscience and Remote Sensing Symposium, 2011 IEEE International*, pp. 1433–1436.

Wang, C., Zhang, Y., Wu, Y., Gu, Y., 2011b. Highly accurate geometric correction for seriously oblique aero remote sensing image based on the piecewise polynomial model. *Journal of Computational Information Systems*, Vol. 7, pp. 342–349.

Wang, D., Pan, D., Gong, F., Wang, T., 2012. Airborne geometry correction method for marine multispectrum data without attitude information. *Acta Oceanologica Sinica*, Vol. 31, pp. 59–65.