New algorithms for satellite data verification with and without the use of the imaged area vector data

Andrey Kuznetsov
Samara State Aerospace University
Moskovskoye shosse, 34
Russia, Samara
kuznetsoff.andrey@gmail.com

Vladislav Myasnikov
Samara State Aerospace University
Moskovskoye shosse, 34
Russia, Samara
vmyas@geosamara.ru

ABSTRACT

This paper presents a solution of remote sensing data verification problem. Remote sensing data includes digital image data and metadata, which contain parameters of satellite imaging process (Sun and satellite azimuth and elevation angles, creation time, etc.). The solution is based on the analysis of special numerical characteristics, which directly depend on the observation parameters: sun position, satellite position and orientation. These characteristics are based on model-oriented descriptor, proposed by one of the co-authors of this paper. We propose two fully automatic algorithms for remote sensing data analysis and decision-making based on data compatibility: the first one uses vector data of the imaged area as a prior information, the second doesn't. After algorithms description we provide results of conducted experiments and explain appliance limits of the proposed algorithms.

Keywords

Satellite digital image, vector map, shadow buffer zone, model-oriented descriptor, amplitude-phase mismatch, Canny edge detector, edge tracing

1. INTRODUCTION

Widely used in the modern world remote sensing data (RSD) consist of two main components: a digital image and its metadata, which describe the process and the observation parameters. During RSD transmission from source to destination, this data can be distorted accidentally (due to errors) and intentionally (by hackers). When this happens, the satellite image itself and/or its metadata can be changed. The problem of forgery detection in digital images, when observation parameters and image metadata are not used or unknown, is being solved in [Chr12a, Glu11a, Far09a, Far09b, Kuz14a].

Nowadays, there are papers devoted to the analysis of light parameters inconsistency for local parts of a single object in digital images [Mya12a]. These algorithms use only image data during analysis, because additional information about observation parameters is absent (the research is carried out for digital images obtained by ordinary cameras that do not store observation information). Due to the lack of this data, there is nothing to compare with angles and lengths of shadows in the analyzed image. Metadata of satellite images and imaged area vector maps allow to analyze the consistency of objects and their shadows. During literature analysis there were not found any papers aimed at the detection of inconsistency in shadows and objects in satellite images.

In this paper we propose a new solution for detection of digital satellite image and observation parameters inconsistency using model-oriented descriptors, proposed in papers [Mya12a, Mya12b] by one of the co-authors of this work.

2. PROBLEM DEFINITION

To identify irrelevance between an image and its metadata, we will analyze the shadows of tall objects on the image. There will be used buildings with height of at least 12 meters (for example, houses with 5 floors and more), which have a simple rectangular form on a satellite image received by nadir observation. The length of the analyzed shadows of such a building is 10-15 m - if the length exceeds this value the shadows may be imposed on neighboring buildings (in dense urban areas), which may impair analysis quality. It is better to identify objects and their shadows with such linear characteristics on high-resolution images (0.5-1 m). This is why we will use Geoeye-1 satellite images (spatial resolution – 0.5 m). This parameters characterize the restrictions wherein the performed algorithms will work correctly.

Image metadata contains the following observation parameters of the satellite image:

1. image coordinates $\mathbf{s} = (s_1, s_2, ..., s_k)^T$, where $s_i = (x_i, y_i)$ is a reference point of a satellite image, k is a number of reference points;
2. satellite position coordinates $\mathbf{p} = (\varphi_{az}, \varphi_{el}, h_{alt})$, where φ_{az} is the azimuth incidence angle of a satellite's sensor, φ_{el} is the elevation incidence angle of a satellite's angle, h_{alt} corresponds to the altitude value of a satellite;
3. Sun position coordinates $\boldsymbol{\alpha} = (\alpha_{az}, \alpha_{el})^T$, where α_{az}, α_{el} are azimuth and elevation incidence Sun angles respectively.

Fig. 1 shows the relative position of azimuth and zenith angles of Sun and spacecraft.

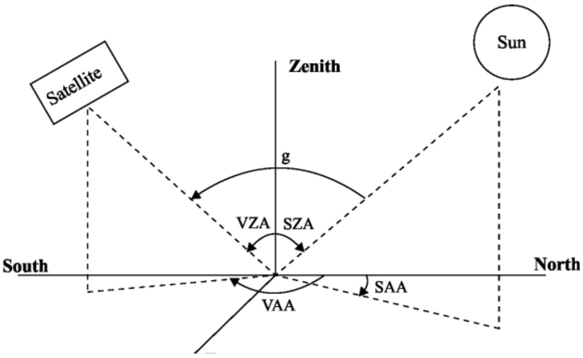

Figure 1. Arrangements of the angles for Sun and spacecraft (VAA – azimuth angle, VZA – zenith angle, SAA – azimuth Sun angle, SZA – zenith Sun angle, g – phase angle).

3. AMPLITUDE-PHASE MISMATCH

Model oriented descriptor of a digital image was proposed in [Mya12a, Mya12b]. It is a new descriptor type, which is formed on the basis of differential and probabilistic properties of the local neighborhood of the analyzed image [Mya12b].

At the heart of the model-oriented descriptor is the use of gradient field probability distribution that describes the model of the analyzed image fragment. Descriptor's components for a particular image area are calculated as the values of the probability density of a specific gradient field or its individual components. Such specificity of the proposed descriptor calculation allows to classify it as a *model-oriented* and to use it as a part of some classifier's decision rule or as a numerical characteristic of an image local area. For some image processing problems solution it is convenient not to use descriptor components, but its derivative values, called descriptor features, which were introduced in [Mya12b]. As it is shown in this work, all the proposed descriptor features have a useful property – their possible values lie in the range [0, 1]. This means that larger values correspond to greater similarity of particular image part (and, as a consequence of its gradient field) to the potentially possible realizations of the gradient field (the model). For a number of standard models of the random gradient field there were obtained explicit expressions for model-oriented descriptor features [Mya12b]. One of these models and the corresponding feature (amplitude-phase mismatch) are used later in this work [Mya12a, Mya12b].

The base of a model-oriented descriptor is the use of probability distribution of the gradient field, which characterizes the model of the analyzed image fragment. Values of descriptor's components for a particular image fragment are calculated as the values of probability density of the argument in the form of a specific gradient field or some of its components.

For a formal definition of this descriptor, we introduce some notation. Let D be an analyzed image area (area of some real object's shadow), for which the function $\varphi(t_1, t_2)$ is defined. The values of this function define orientation (angle) of a brightness difference line (along shadow's boundaries) in the corresponding position (t_1, t_2).

We will call the following equation as an amplitude-phase mismatch (APM) ζ for an image area D:

$$\zeta = \frac{SGD}{SGM}, \zeta \in [0,1], \tag{1}$$

where *SGD* and *SGM* are represented in the form:

$$SGM = \sum_{(t_1,t_2)\in D} |g(t_1, t_2)|,$$

$$SGD = \sum_{(t_1,t_2)\in D} |g(t_1, t_2)| \left(\frac{\cos(\varphi(t_1,t_2) - \arg(g(t_1,t_2))) + 1}{2} \right).$$

At this point $g(t_1, t_2)$ is a concrete implementation of the gradient field for the given image fragment, $|g(t_1, t_2)|$ and $\arg(g(t_1, t_2))$ are its modulus and direction (phase) respectively. It is obvious that the closer APM's value ζ to 1, the more image area D matches a template, represented by $\varphi(t_1, t_2)$ function. APM, in fact, shows how far is the real gradient direction value from $\varphi(t_1, t_2)$ direction.

4. PROPOSED SOLUTION

Image and observation parameters verification procedure with imaged area vector map use

Let us consider a situation when there is a priori information about the imaged area – a vector map of this area. By carrying out a geometric calibration of a snapshot and putting it on a vector map of the imaged area, it is possible to determine positions of physical objects on the space image. Depending on the angle φ_{el} the roofs of the objects in the image can be displaced according to the spacecraft inclination angle, whereas the vector objects correspond to the foundation of these buildings and are situated as it would be for nadir satellite imaging. The example of combining the space image received from Geoeye-1 (0.5 m) satellite and the vector map of the imaged area is presented in Fig. 2.

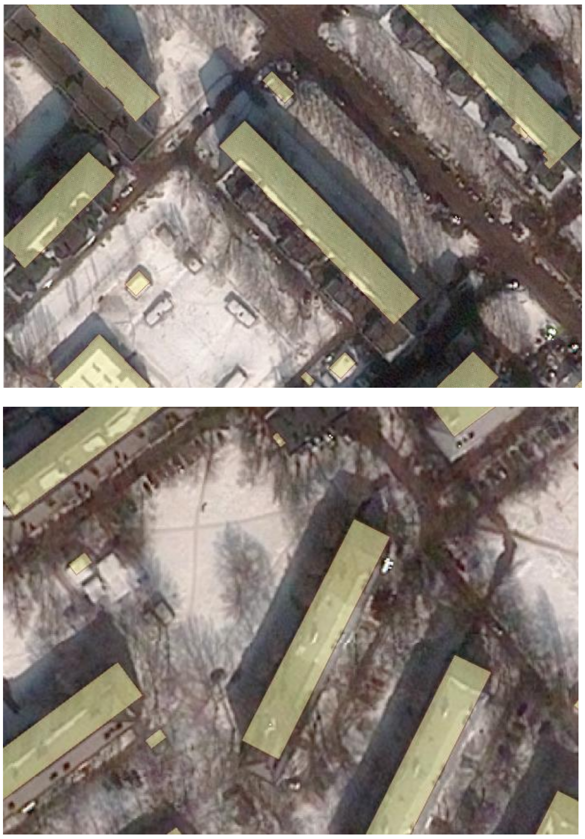

Figure 2. Satellite image and vector map combination for analyzed objects, $\varphi_{el} = 35$.

From now on we will neglect geolocation accuracy for the proposed solution description and conducted experiments.

Using semantic data of the buildings vector layer we select only those buildings, which height is more than 10 meters $h_b > 10$ (buildings height values are listed in the semantic data of the vector layer). The contour of each building is described by four points $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)\}$. The distance between any two points will be denoted as:

$$d_i^j = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} .$$

Let us define the Sun position in the satellite image coordinate system (x_{sun}, y_{sun}) so, that

$$d_i^{sun} >> \max_{i,j} d_i^j, i, j \in \overline{1,4}, i \neq j .$$

For the analyzed building it is necessary to determine the pair of its sides for which the thrown shadow angle remains right (the angle whose vertex is the farthest from the Sun, e.g., B'A'C'):

$$i_{\max} = \arg\max_{i \in \overline{1,4}} d_i^{sun} .$$

There is shown an example (Fig. 3) of an object ABDC and its thrown shadow with length s (we assume that the imaging process was carried out at nadir point). The shadow is thrown by the sides AB and AC.

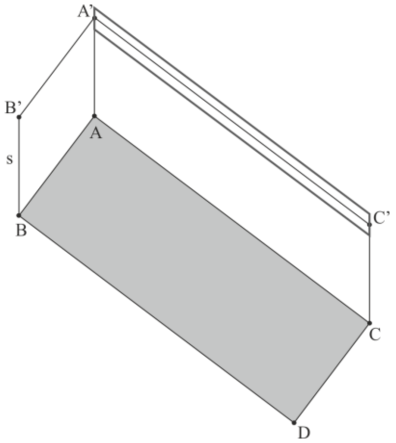

Figure 3. Shadow buffer zone for one side of the building.

Let us determine the length of the shadow of the object s as $s = h_b tg\alpha_{el}$. We then can determine the buffer zone of the shadow boundaries – geometric locus at the edge of the shadow thrown by a building side (there is allocated a buffer zone for the shadow edge A'C' in Fig. 3). In the shown in Fig. 3 case the buffer zone is a parallelogram with two sides parallel to AC, and the other two belong to a line lying at the shadow inclination angle α_s (calculated with respect to the direction of the X axis of the rectangular coordinate system of the analyzed image). We will call the shadow angle of the longer side of the parallelogram as *the direction of the buffer zone boundary*. The buffer zone for the whole building shadow lies along the polyline BB'A'C'C and consists of 4 parts: along BB', B'A', A'C' and C'C.

Let *L* be the buffer zone height (see Fig. 4), then the length of the parallelogram sides, which limit the shadow side of the buffer zone A'C ', is calculated as follows:

$$d_{A'_-}^{A'_+} = H = \frac{L}{\sin\beta},$$

where $\beta = \alpha - \alpha_s$.

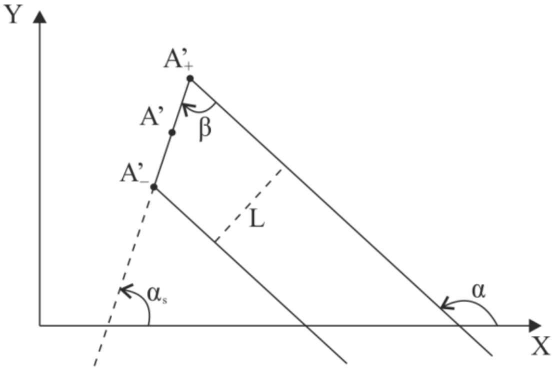

Figure 4. Shadow buffer zone edge calculation for a building side.

Shadow buffer zone borders, parallel to the side AC, will be separated by the distances $s - \frac{H}{2}$ and $s + \frac{H}{2}$ from the side AC. Coordinates of the buffer zone corners are calculated in a trivial way.

Doing similar calculations for the other three shadow borders, we get coordinates of the corners of their buffer zones. The result of the building shadow buffer zone construction (*D*) is shown in Fig. 5.

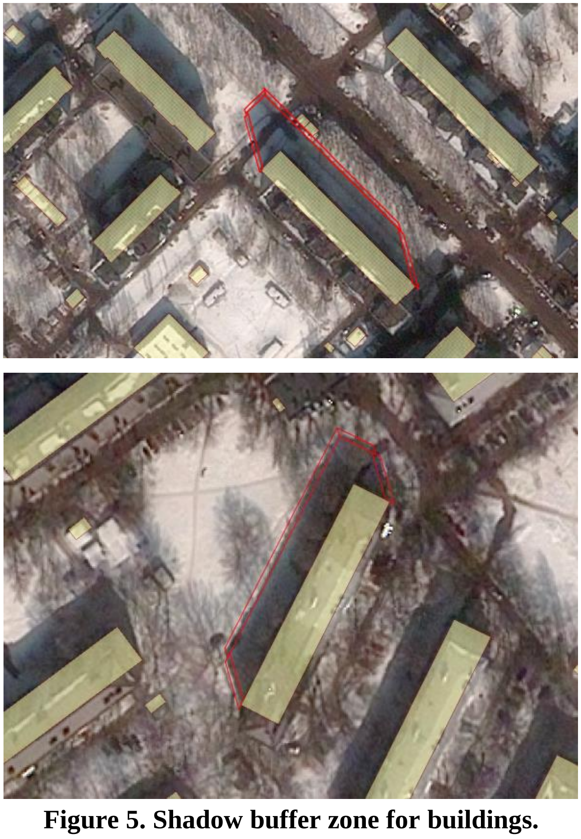

Figure 5. Shadow buffer zone for buildings.

For each of the buffer zones we will calculate APM values (1), which characterize correspondence of the real object's shadow in the satellite image (according to the orientation of the buffer zone) to the value, calculated using metadata parameters.

Image and observation parameters verification procedure without imaged area vector map use

If there is no vector map of the imaged area we need to detect buildings and corresponding shadows using only image analysis methods. We will use high resolution images for analysis as in the previous algorithm. In this paper we propose the algorithm that allows to identify the corresponding buildings corners and shadows thrown by these corners using Canny detector [Gas03a, Can86a]. This method provides precise detection results for noisy images and detected edges are one pixel in width, which enables to trace them further [Ren02a].

Let us take the following image for analysis (see Fig. 6):

$$f(m,n), m \in [0, M), n \in [0, N),$$

where *M*,*N* are image linear dimensions of the image.

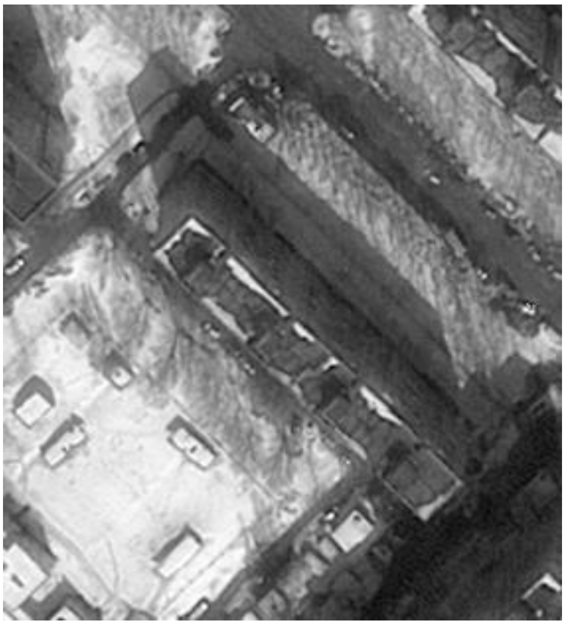

Figure 6. Analyzed image part.

Before its analysis it is necessary to make some pre-processing steps:

1) convert the image to grayscale (if it is multichannel);

2) filter noise to smooth the edges.

The image $f'(m,n)$ is a result of the above preprocessing operations.

After that we apply Canny edge detection algorithm to the preprocessed image (by means of *OpenCV*). For detector configure there are used two parameters: the first one is used to select the most significant boundaries (th_1), the second one is used to combine edge segments into contours (th_2). In this paper we use empirically selected parameters for edges detection $th_1 = 50, th_2 = 120$. These values provide the best precision for edges detection. The result of Canny edge detector will be denoted as $c_{f'}(m,n)$.

The algorithm of Canny edge detection result $c_{f'}(m,n)$ for image verification consists of two steps:

1) detection of corresponding angles of buildings roofs and shadows thrown by them;

2) detection of shadows edges parts that are collinear with shadow inclination angle, calculated using the values of analyzed image metadata.

The first step of the proposed algorithm include detection of angles between the edges that are close to 90°. The closeness of these values will be determined by a threshold parameter $\Delta_{rightAngle}$. Each building has a right angle of the roof, which corresponds to a right angle of its shadow. For each edge pixel (x_b, y_b) we produce eight-connected tracing procedure [Ren02a] in opposite directions and estimate the angle γ between these traced edge parts. If the following condition $\gamma \in \left[\frac{\pi}{2} - \Delta_{rightAngle}\ , \frac{\pi}{2} + \Delta_{rightAngle}\right]$ is satisfied, then (x_b, y_b) point is placed in the list of points, which may be a building roof angle or a shadow angle. Then the points list is filtered and only those pairs of points $(x_1, y_1),(x_2, y_2)$ are selected for which the following condition is fulfilled:

$$\left| arctg2\left(\frac{y_1 - y_2}{x_1 - x_2}\right) - \alpha_s \right| < \Delta_{sun}\ .$$

There is also taken into account the minimum and maximum possible heights of buildings, which depend on the length of objects shadows.

The result of detection of buildings roofs and thrown shadows corresponding angles is presented in Fig. 7.

The second step of the proposed algorithm is to identify edges of the shadows, which direction coincides with shadow inclination angle, calculated using the values of analyzed image metadata. In the basis of this operation also lies the tracing of $c_{f'}(m,n)$. Let K_s be a restriction on the maximum pixel length of the traced edge. When we determine a list of K_s points for a given point, we approximate it

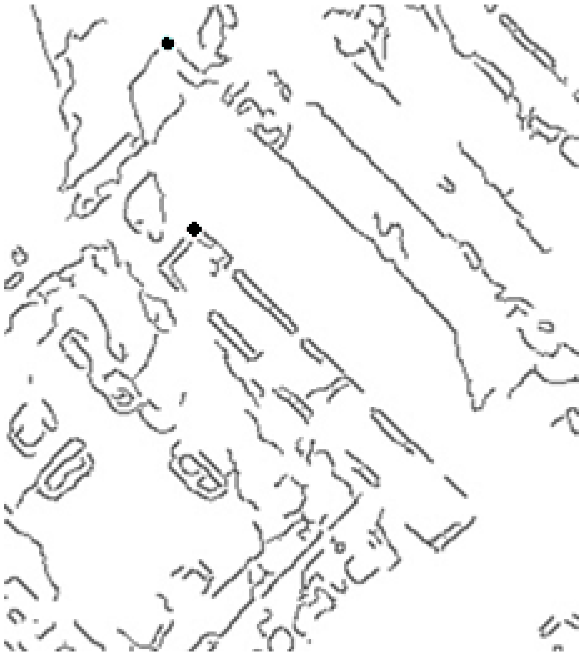

Figure 7. Corresponding angles detection for a test building.

a line [Gas03a] using *Line2DFitting* function of *OpenCV*. As a result we obtain a point (x_{line}, y_{line}) belonging to this line and line direction vector (d_x, d_y). The result of this operation is presented in Fig. 8.

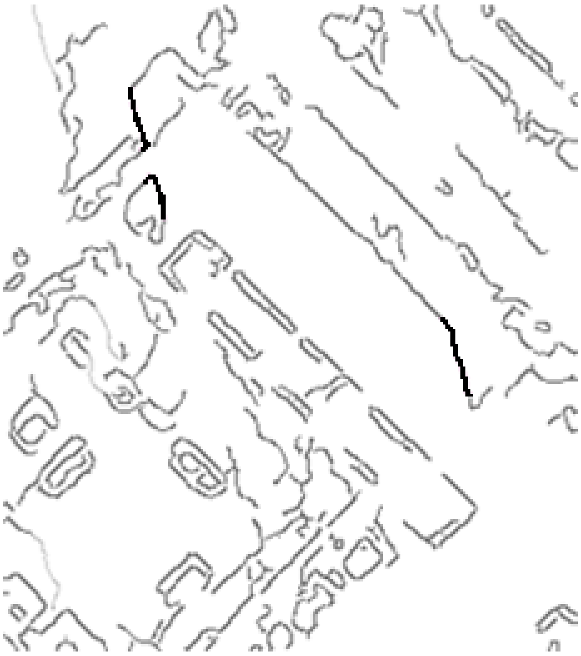

Figure 8. Detection of shadow edges collinear to metadata shadow angle.

Using the list of corresponding right angles and the list of shadow edges closest to them there is formed a geometric model of the building shadow for which the APM value is calculated, as described in the previous subsection.

5. EXPERIMENTAL RESULTS

During the algorithm research we define APM threshold values, which will be used for a decision making of satellite image to observation parameters correspondence. To conduct an experiment we choose Geoeye-1 satellite images (0.5 m resolution) and a set of 26 vector objects, randomly selected among the objects belonging to the territory of the snapshot. Taking into account the randomness of objects selection, some shadow buffer zones boundaries may appear in the shadow region of other vector objects, or may be blocked by other objects in the image.

APM values are calculated for any image channels in two ways:

1) APM value is calculated for each side of the shadow buffer zone of the object, so a training sample will consist of 104 variables – 4 values for each vector object;

2) APM value is calculated for the whole shadow buffer zone of the object - the size of a training sample will be 26 variables.

Shadow buffer zone boundaries are calculated for a given correct shadow inclination angle $\alpha_s = 75^\circ$, which was calculated based on the satellite image metadata parameters.

Let L be the volume of a training sample, then the APM threshold value for the i-th training sample method is defined as follows:

$$t_i = \min_{k \in L}\{\zeta_k\} \cdot 0.9,\ i \in \{1,2\},$$

where ζ_k is the APM value for k-th object of a training sample, 0.9 is a constant defined experimentally.

Fig. 9 and Fig. 10 show the distribution of APM values for both techniques of creating a training sample and the corresponding threshold values. Thresholds in the figures are as follows $t_1 = 0.34$ and $t_2 = 0.6$.

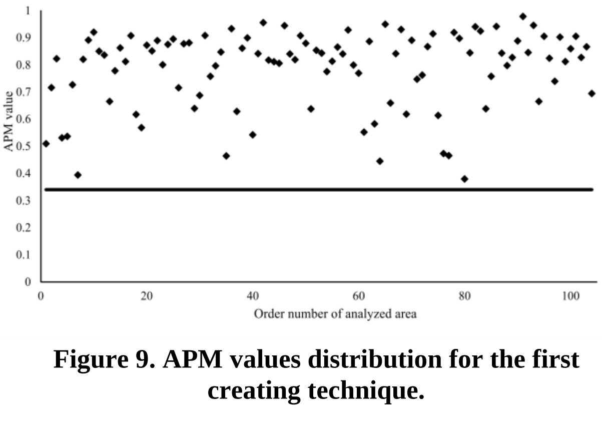

Figure 9. APM values distribution for the first creating technique.

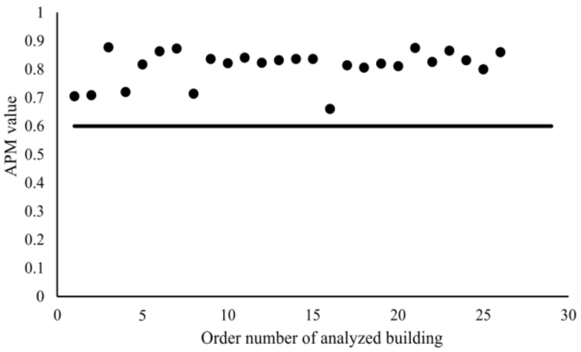

Figure 10. APM values distribution for the second creating technique.

Decision making of satellite image to observation parameters correspondence is performed as follows. There is selected a test sample of 20 buildings (vector objects) for the analyzed satellite image. On the first stage APM values are calculated for each element of the shadow buffer zone and the object doesn't pass a test if:

$$\exists\ j \in \overline{0,3},\ \zeta_j < t_1 \tag{2}$$

During the second stage APM values are calculated for the entire shadow buffer zone and the decision is made in a similar way:

$$\zeta < t_2 \tag{3}$$

Satellite image does not pass the validation test, if at least one test sample object does not pass a two-stage test procedure (2) – (3).

In order to confirm the correctness of APM threshold values t_1, t_2 selection we take a satellite image and a test sample of 20 vector objects, which belong to the territory of the snapshot. We then construct a relationship between the values of shadow inclination angle and the number of objects that did not pass the two-stage procedure of satellite image validation (see Fig. 11).

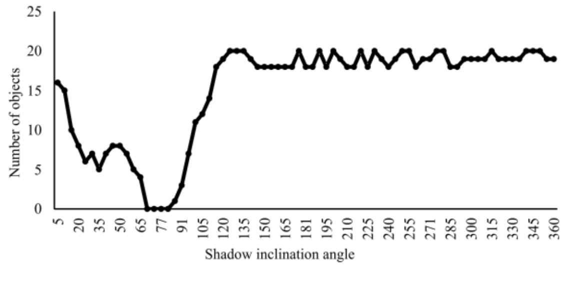

Figure 11. Dependency of test sample objects number that failed validation test from shadow inclination angle.

For presented in Fig. 6 buildings the APM value $\zeta_1 = 0.77, \zeta_2 = 0.93$ exceeds the threshold APM value. This allows to make a decision that this object corresponds to observation parameters.

According to the results of conducted experiments it can be concluded that the developed algorithm detects inconsistency of a satellite image and its observation parameters when the deviation of shadow inclination angle from its correct value is $\Delta_{\alpha_s} > 5^\circ$ for the calculated APM threshold values t_1, t_2. This is acceptable for the analysis of satellite images.

Both of the proposed algorithms have low computational complexity and can be used for real-time satellite image analysis (Geoeye-1 satellite image with size 10000×10000 pixels and 1000 vector objects average analysis time is 5400 ms on Intel Core i5 3470, 8Gb RAM).

6. CONCLUSION

In this paper we presented new algorithms for detecting inconsistencies of satellite image data and its observation parameters: with and without the use of imaged area vector map. The proposed solution makes it possible to detect inconsistencies of objects and observation parameters at a deviation angle greater than 5°. The paper also provides recommendations on parameters choice and detection algorithms usage limits. Further we are going to compare different shadow detection algorithms as one of the steps of the proposed solution and to develop an algorithm for buildings with more complex geometry.

7. ACKNOWLEDGMENTS

This work was partially supported by the Russian Foundation of Basic Research (projects: 13-07-12103-ofi, 15-07-01164-a), by the Fundamental research program № 15 of the Russian Academy of Sciences Presidium and by the Ministry of Education and Science (state order 2298).

8. REFERENCES

[Can86a] Canny, J. A computational approach to edge detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on PAMI. Volume 8, Issue 6, pp. 679-698, 1986.

[Chr12a] Christlein, V., Riess, C., Jordan, J., Riess, C., and Angelopolou, E. An Evaluation of Popular Copy-Move Forgery Detection Approaches. IEEE Transactions on Information Forensics and Security. Volume 7, No. 6, pp. 1841-1854, 2012.

[Gas03a] Gashnikov, M., Glumov, N., Ilyasova, N., Myasnikov, V. [et al]. Methods of computer image processing (2-nd edition reviewed). Moscow: “Fizmatlit Publisher”. 784 p., 2003.

[Glu11a] Glumov, N., Kuznetsov, A., and Myasnikov V. The algorithm for copy-move detection on digital images. Volume 7, No.3, pp. 360-367, 2013.

[Far09a] Farid, H. Image Forgery Detection. IEEE Signal processing magazine, pp. 16-25, 2009.

[Far09b] Farid, H. Exposing digital forgeries from JPEG ghosts. IEEE Transactions on Information Forensics and Security. Volume 1, No. 4, pp. 154-160, 2009.

[Kuz14a] Kuznetsov, A., and Myasnikov V. A fast plain copy-move detection algorithm based on structural pattern and 2D Rabin-Karp rolling hash. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Volume 8814, Issue 1, pp. 461-468, 2014.

[Mya12a] Myasnikov, V. Method for detection of vehicles in digital aerial and space remote sensed images. Computer optics. Volume 36, No. 3, pp. 429-438, 2012.

[Mya12b] Myasnikov, V. Model-based gradient field descriptor as a convenient tool for image recognition and analysis. Computer optics. Volume 36, No. 4, pp. 596-604, 2012.

[Ren02a] Ren, M., Yang, J., and Sun, H. Tracing boundary contours in a binary image. Image and Vision Computing. Volume 20, Issue 2, pp. 125-131, 2002.