# Image Processing in Thermal Cameras

**Chapter** · January 2018

DOI: 10.1007/978-3-319-64674-9_3



**3 authors**, including:

Grzegorz Bieszczad
Military University of Technology
**61** PUBLICATIONS **417** CITATIONS

SEE PROFILE

# Image Processing in Thermal Cameras

**Tomasz Sosnowski, Grzegorz Bieszczad and Henryk Madura**

## 1 Introduction

In modern security systems more and more commonly thermal camera are used exploiting infrared radiation imaging to perimeter observation and thread detection, especially when there is a need to proceed in limited visibility conditions of complete darkness. Implementation of cameras in security system causes a substantial increase of information fed to security system operator. Thermal image processing system can help the security system operator by enhancing the relevant information in the image, which enables to discern important image details [1]. More advanced imaging systems detect threats automatically and present information about that fact directly to the operator. To perform such tasks, special algorithms are implemented for detection and tracking of objects in the image. On the grounds that, the detection algorithms used for tracking and vision systems operating in the visible light cannot be directly used for the analysis of thermal images [2], special methods have been developed for detection and tracking of objects on the thermal image.

T. Sosnowski (✉) · G. Bieszczad · H. Madura
Military University of Technology, Institute of Optoelectronics, Warsaw, Poland
e-mail: tomasz.sosnowski@wat.edu.pl

G. Bieszczad
e-mail: grzegorz.bieszczad@wat.edu.pl

H. Madura
e-mail: henryk.madura@wat.edu.pl



A. Nawrat et al. (eds.), *Advanced Technologies in Practical Applications for National Security*, Studies in Systems, Decision and Control 106,
https://doi.org/10.1007/978-3-319-64674-9_3

## 2 The General Structure of the Infrared Camera—Algorithms and Methods for Image Processing in Infrared Cameras

Thermal cameras are more and more often used as the observation device in security systems for perimeter protection, military systems for object detection, identification and tracking, in pollution detection and many more. In such systems, it is important to process infrared information that the resulting image is faithfully corresponding to the observed situation. More and more common use of infrared cameras as surveillance equipment means that they should be as simple to use as possible. This forces the need for implementation of automatic thermal image processing and analysis methods. These methods allow to simplify the operation of the camera by means of automatic adjustment of operating parameters of the thermal imager. The methods used should also allow the work of the infrared camera not only as a tool to support observation, but also for detecting and identifying emerging objects and phenomena. As used in the device processing method is dependent on the particular application and the type of the analyzed information [1], therefore they cannot be universal or selected once and for all. Moreover such automatic systems performing processing and image analysis must have a relatively small size and low power consumption. In general an electronic system in the thermal camera can be divided in three essential modules [3–6]:

- focal plane array module,
- control and digital processing module,
- imaging module.

Simplified schematic of electronic system is shown on Fig. 1.

In detector array module there is a high performance analog to digital converter that allows transforming analog signals form detectors to digital form possible to process in control and image processing module. In detector array module there are also sophisticated power sources for powering and biasing internal circuitry of the array and special filters that enables noise immunity.

Control and image processing unit sends synchronization and control signals responsible for proper readout from the detector array module [6, 7]. For example if the camera has to work in broad range of temperatures without temperature stabilization, the control unit has to control the biasing voltages of the array.

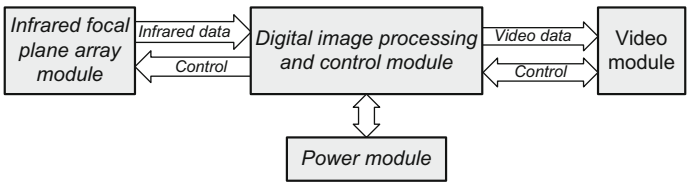


**Fig. 1** Schematic of electronic modules in microbolometric thermal camera

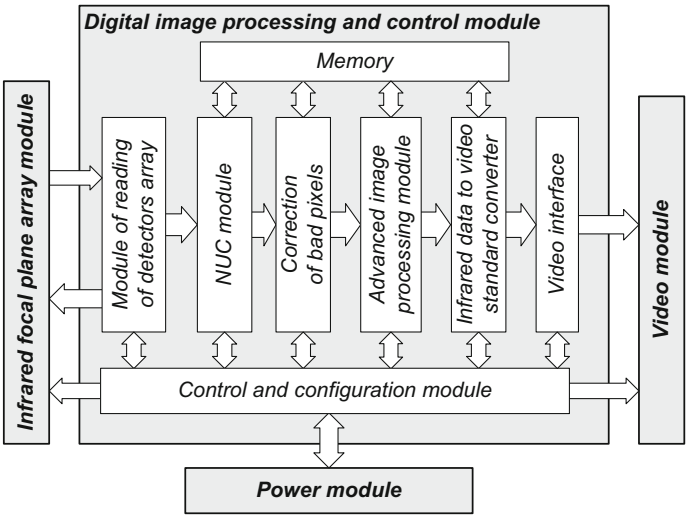


**Fig. 2** Functional block diagram of control and image processing module

Biasing voltages are largely responsible for the sensitivity of the detectors on the focal plane array and for maintaining an adequate common mode level of the output signal.

Control and image processing unit is responsible for configuration of all other modules and image processing of collected infrared image data. The main tasks performed by the image processing and control module are the following: control of focal plane array to read the values of all detectors in the matrix, nonuniformity correction, correction of signal from defective detectors and generation of data for the display module. The block diagram of control and image processing module is shown on Fig. 2.

The main problem of hardware implementation of complex image processing algorithms such as image enhancement algorithm, is their high computational cost. The implementation of complex algorithm by means of software in a general-purpose processor usually does not give satisfactory results. For complex image processing algorithms implemented in real time Application Specific Integrated Circuits are often used. However, due to a predefined architecture typically they have limited functionality and have a relatively long development time and considerable price. The alternative is to use reconfigurable computing architecture based on programmable devices [4].

The use of programmable system is a flexible and powerful solution that allows the execution of complex operations of image processing in real time. Data processing modules organized in pipeline, usually exchange data using standardized data bus, for example with VideoBus [5, 6]. This allows to swap the order of operations performed by image processing modules without interfering with the overall pipeline system. Therefore, the control and digital image processing module

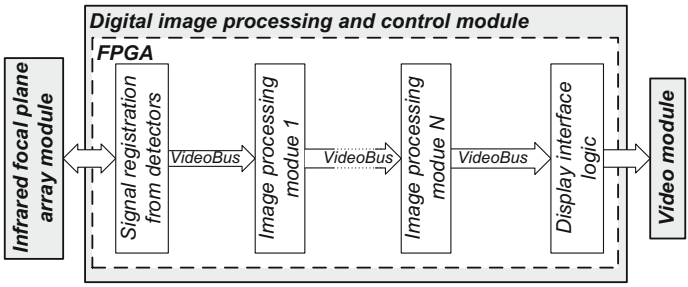


**Fig. 3** Block diagram of image processing chain implementation in FPGA

is often built on the basis of two basic systems: programmable system e.g. the FPGA and microcontroller circuit. The programmable system performs the processing of image data, which requires significant computing power, the microprocessor performs all activities related to the control and other activities that demand relatively small computational power. A block diagram of image data processing modules implementation in the FPGA is shown in Fig. 3.

Generally speaking, image processing in the thermal imaging camera can be divided into three groups of methods and algorithms. The first group of algorithms and processing methods are the algorithms and methods that are necessary for the operation of the thermal imager. This group includes such processing as nonuniformity correction of detector responses in FPA [7, 8] and algorithms for replacing signal from faulty detectors. The second group of algorithms are the algorithms used to improve image quality, in order to enable and facilitate the interpretation of the thermal visualization by the operator or the vision system. The third group of image processing algorithms are the methods of data analysis for the automatic detection and tracking of objects in an image and interpret the scene (i.e. Machine vision).

## 3 Basic Thermal Image Processing Algorithms

A first group of algorithms and processing methods are the algorithms and methods that derive from the working principles of the infrared detector array, and the physical laws relating to infrared radiation. The main processing algorithms is a correction of response nonuniformity of each detector in the array and the detection and replacement of damaged detectors. There are many algorithms for detectors response nonuniformity. The most commonly used and popular methods include one-point and two-point correction methods [7, 8].

On Fig. 4 is schematically shown the nonuniformity correction algorithm of the FPA with the two point method. Regarding the replacement of defective detectors, the procedure is divided into two phases. The first phase involves detection and localization of the faulty detector. Second phase involves interpolation of the signal

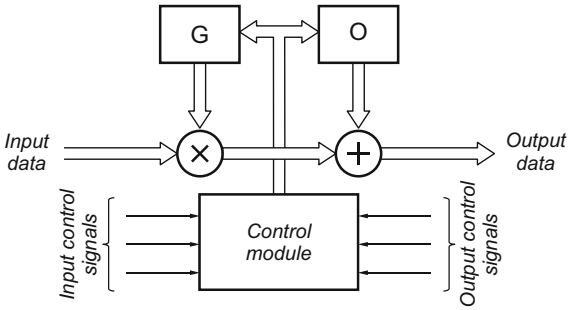


**Fig. 4** Block diagram of hardware nonuniformity correction module

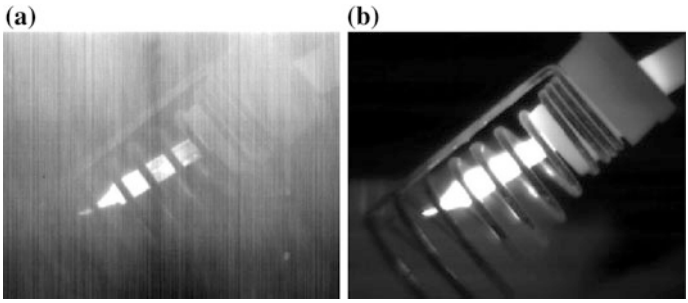


**Fig. 5** Exemplary thermal images before (**a**) and after (**b**) nonuniformity correction

for signal replacement from faulty detector. Generally, one can distinguish the following manifestations of damaged detector:

- detector returns a value outside the dynamic range of the readout circuit,
- detector returns a constant value, independent of the incident radiation (does not respond to changes in incident radiation),
- detector has too low or too high sensitivity to incident radiation,
- detector is characterized by high noise, independent of the incident radiation,
- detector flashes, that means it significantly alters its value with a very low frequency in range of 1 Hz or less (this symptom occurs in cooled detectors only).

Algorithms and methods for detecting malfunctioning detectors are based on the causes of faults in the detectors. The most commonly used methods for detecting defective sensors include: offset criterion, noise criterion, sensitivity criterion, and algorithms for blinking pixels detection. On Fig. 5 the effect of non-uniformity correction operation and removal of defective detectors is shown.

# 4 Initial Image Processing—Image Enhancement

In general, thermal image is a visualization of the infrared radiation emitted by the observed object and its surroundings. Visualization of the thermal scene is significantly different from the image recorded by the video camera. It follows that the thermal image is often difficult in interpretation. Proper interpretation of the thermal image is associated with proper understanding of infrared radiation properties as well as specific object properties and surrounding scenery. Throughout the process of perception (analysis and interpretation of the image) an important role plays not only objective factors like the emissivity of the object, but also subjective factors ex. human perception of visual information. Unfortunately, the optimum parameters setting of infrared camera alone (sharpness, temperature range) does not guarantee a correct detection and interpretation [2, 9, 10].

Therefore, in a thermal imaging camera, special image processing is performed, for example to ensure automatic dynamic range control of the entire image or selected area. The primary purpose of techniques to improve the quality of the image (image enhancement) is so that the image obtained as a result was more suitable for a particular application, not only to make it more pleasing for human observer. Differences between visible spectrum video and thermal image causes that adaptation of image processing techniques is more challenging and demands custom solutions [2, 9, 11].

The most common algorithms to improve image quality are: image contrast modification, sharpening, removing geometry distortions, smoothing and denoising etc. Described in the article methods to improve the infrared image quality comprises in three categories: context-free methods (also called point methods), contextual image processing, methods for histogram modification.

## *4.1 Context-Free (Point) Image Processing Methods*

The most common Operation in the early stages of image processing is the point (context-free) processing. It is a fundamental and at the same time the simplest operation performed in image processing. Point image processing is typically implemented in hardware using a look-up-table (LUT) processing. LUT processing performs pixel value transformation using an array of data contained in the memory. A diagram of LUT technique implementation is presented on Fig. 6.

A typical example is the LUT operation is correction of brightness and contrast of the infrared image. Change in the brightness or contrast corresponds to an adequate change in values within the LUT (Fig. 7).

Figure 7a shows the changes taking place in the table when modifying the image contrast. Increasing the contrast causes the angle of the slope to increase, what means that the differences in the levels of brightness of the individual pixels in the output image are also increased. The maximum number of brightness levels of the

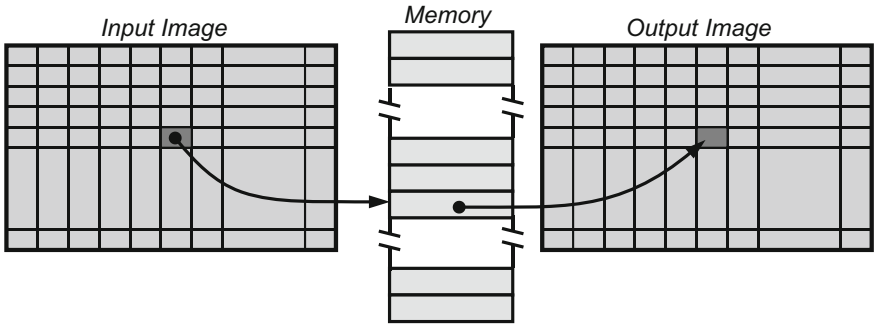


**Fig. 6** The implementation scheme of "look-up-table"

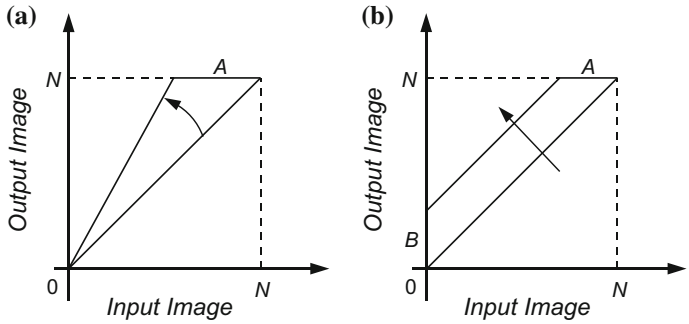


**Fig. 7** Modifying the brightness (**b**) and contrast (**a**) of the image using the LUT

input and output is constant and equals N. For this reason, increasing the contrast is followed by cutting the dynamic range of input stream as shown on the picture by section A. All levels of brightness of the original image in range indicated by section A results in the truncated level of brightness equaling N. Increasing the brightness of the image corresponds to the translation of conversion function in the vertical direction as shown on Fig. 7b. This is equivalent to adding certain value to every cell in LUT. In this case the cut off segments also occurs. Some of the input brightness levels will not be present in the final image. Absent levels not represented in final image are indicated by section B. Nature of the functions carried out by the LUT values can be closely matched to the specific application. For example, we may care about increasing the contrast only in the central region of the brightness levels. On Fig. 8 is an example illustration of the LUT operation result to normalize the images from the infrared camera. Applied operation has enabled a significant increase in image contrast.

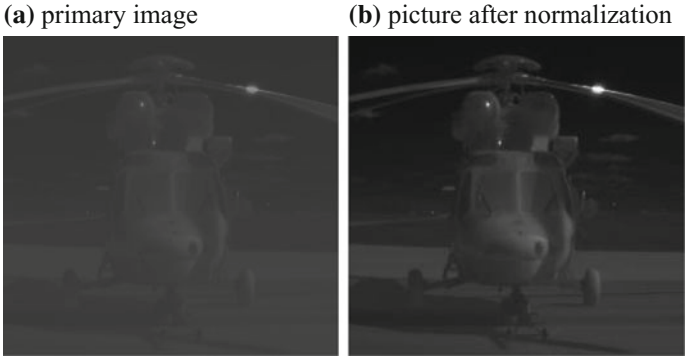


**Fig. 8** Example application of LUT operation to normalize the image

## *4.2 Contextual Image Processing*

Contextual image processing involves replacement of original value of the pixel with value calculated based on the context (environment) of this point. In other words, the new value is calculated based on the neighborhood of the original pixel. The function realizing contextual image processing to be described by the equation:

$$L'(m,n) = \frac{1}{\sum_{i,j \in K} w(i,j)} \sum_{i,j \in K} L(m-i, n-j) w(i,j) \tag{1}$$

where: $L(m,n)$—function representing input values, $L'(m,n)$—function representing output values of the image, $w(i,j)$—weighting factors (kernel of the transform function). Kernel shape is generally square, with an odd number of cells on each side. Based on the kernel coefficients and neighborhood pixels, a new value is calculated. Although the shape of the mask can be any in practice most commonly used mask size 3 × 3.

For realizing functions of the context image processing in real-time, the hardware dedicated processing unit can be used. Dedicated real time processing unit consist of delay lines (buffers), to ensure a continuous flow of the stream of data, and in the same time simultaneous access to neighborhood of actually processed pixel. This allows to perform corresponding arithmetic operation in one batch. Processing module for the kernel size of 3 × 3 is schematically shown in Fig. 9. Contextual image processing module opens possibility to perform vast amounts of image processing methods only by changing applied kernel coefficients. The most commonly used methods with application of contextual image processing are among others: low-pass filtering, high pass filtering, edge detection, feature detection, denoising etc.

The low pass filter elements of the image passes a low frequency signals, while suppressing or blocking elements with a high frequency. These filters are often used

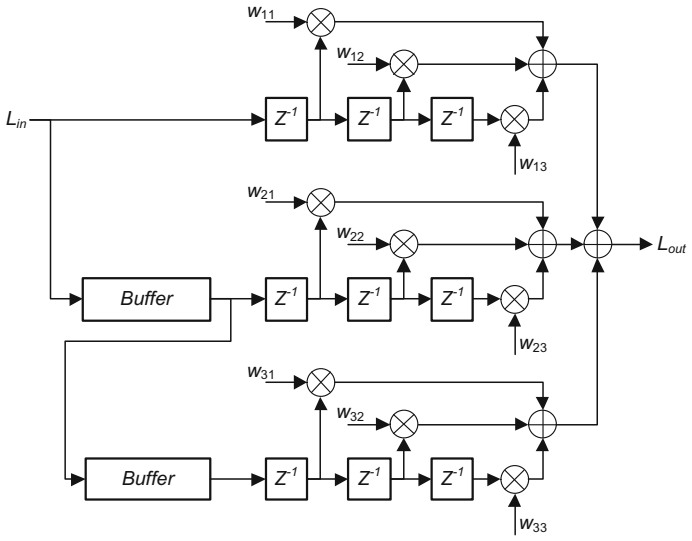


**Fig. 9** Block diagram of the convolution module for kernel size of 3 × 3

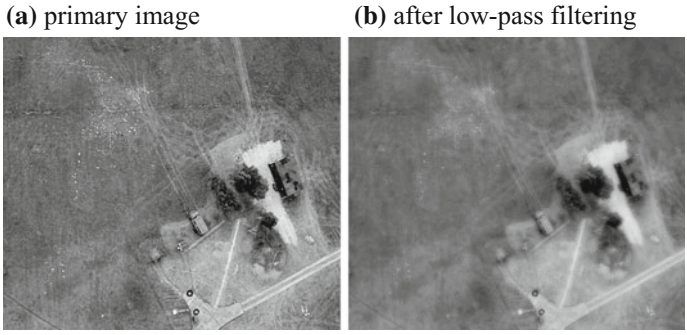


**Fig. 10** Example application of low-pass filtering in order to remove single point interference

to reduce noise in the image, on the other side they causes reduction of fine details in the image. On Fig. 10 an exemplary effect of the low pass filter is shown. Used filter is intended to remove image noise.

High Pass Filtering is used to strengthen the high frequency information present in the image, while preserving the integrity of the low-frequency information. High Pass Filtering is important when objects in the image has to be highlighted or identified.

A high-pass filter increases the sharpness of the image but on the other hand the negative effect occurs from amplification of the noise. On Fig. 11 are shown the examples of high pass filtering.

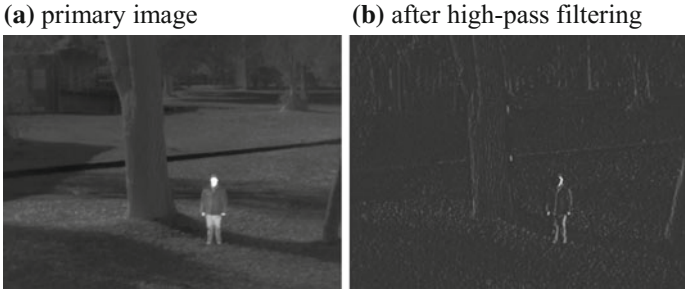


**Fig. 11** An example of the of high-pass filtering application to highlight objects in the image

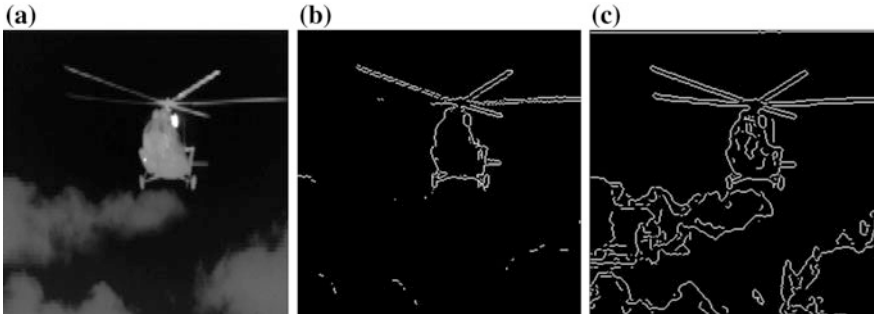


**Fig. 12** Edge detection in the image: **a** original image, **b** Sobel filter, **c** Canny filter

Filters used for edge detection are sometimes called contour detectors. These filters are commonly used for the classification of shapes of objects in the image. They operate on the principle of gradient detection. The gradient is defined as the spatial difference of brightness in the image. Gradient achieves the highest value in places where there are the biggest changes in the brightness between adjacent pixels.

Laplacian operators are often used for edge detection. In comparison with other methods, they are omnidirectional. On Figs. 12 and 13 there is an illustration of edge detection and enhancement.

## *4.3 Histogram Modification*

Histogram of brightness levels is the statistical distribution of the occurrence of a pixels of each brightness level in the image. In the picture we are dealing with both a finite number of pixels and a finite number of brightness levels, implies that the histogram is a discrete function. Histogram modification comprises the operations performed on the image in order to obtain the required shape of the histogram.

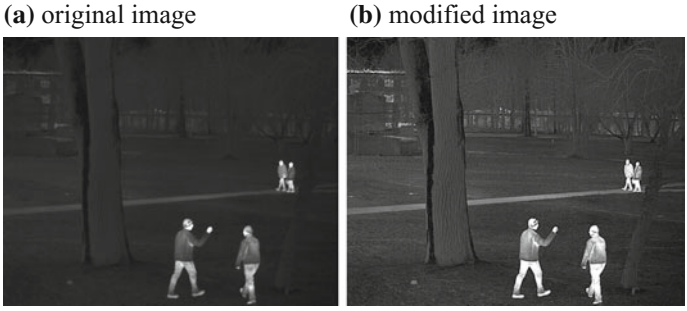


**Fig. 13** Modification of the thermographic image by edge sharpening

One of the basic and most common methods of this type is the histogram equalization. Histogram equalization technique involves such modification of the source image so that the resulting histogram is as flat as possible. Assuming that the histogram represents brightness levels occurrence probability p(i) then we can calculate the cumulative distribution D(i) of the probability distribution using the formula:

$$D(n) = \sum_{i=1}^{n} p(i) \tag{2}$$

Based on the cumulative distribution we obtain an intermediate LUT, which is the function for converting brightness levels in the current image to new levels of brightness. LUT table calculation can be expressed by the formula:

$$i' = \text{int}\left[\frac{D(i) - D(0)}{D(0)}(K-1)\right] \tag{3}$$

where: $i$—the index of the brightness of the original image, $i'$—the brightness of the resulting image, $D(i)$—cumulative distribution for $i$-th brightness level, $K$—number of brightness levels, int—rounding operator. Histogram equalization can be used to highlight details that are barely visible in the image because of the low contrast, but it should be noted that this is not an universal method and for some histogram shapes it does not give satisfactory results.

The module for histogram calculation is one of the more complex modules in digital imaging, however the principle of operation of the module alone is fairly simple and involves counting the occurrences of each pixel value. The implementation of incrementing operation consists of three basic steps: read the memory location, increment the value, and then store the resulting values back to memory. The system calculates the histogram has one cycle of computing operations to perform three operations. Therefore, the implementation requires a relatively fast memory of considerable size. Schematically, the histogram calculation module is shown in Fig. 14. On Fig. 15 is an example of the application of histogram equalization operation of the thermal image of the helicopter.

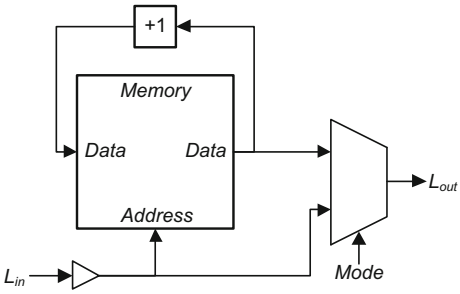


**Fig. 14** The block diagram of the module that calculates histogram

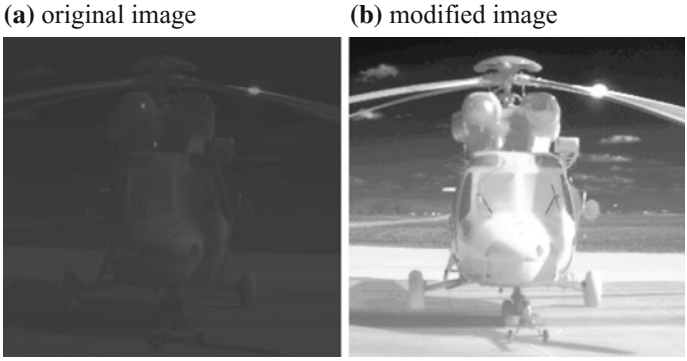


**Fig. 15** Example applications of the histogram equalization operation

## 5 Detection, Recognition and Object Tracking Algorithms

Image analysis is very often used in the detection, classification, recognition and identification of moving objects and targets. Automatic target recognition (ATR) system should be designed to help analyze the image by the analyst by increasing the amount of data that he can process. I also should help by increasing the possible time to spent on desired region of interest ROI by automatic preselection of more interesting objects. It makes possible to reduce the number of analysts and accelerates the analysis of the situation. In general, we require algorithms that allow ATR to distinguish targets from ordinary objects and allow to distinguish between different classes of targets. An important requirement for ATR system is to show exact location and orientation of the moving targets. Generally, in the ATR system the objective is implemented by the sequence of operations

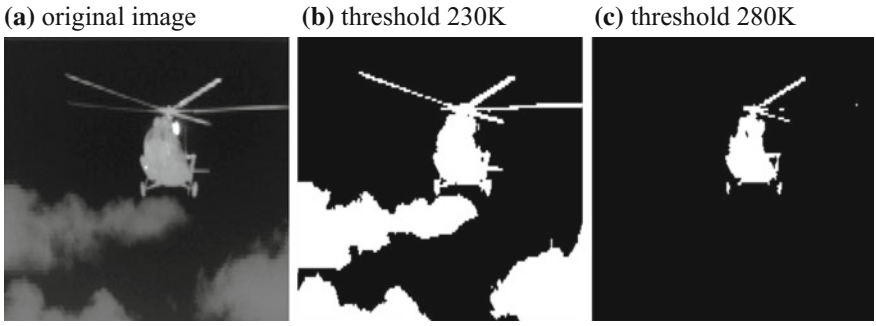


**Fig. 16** Infrared image binarization **a** for different threshold temperature: 230 K (**b**), 280 K (**c**)

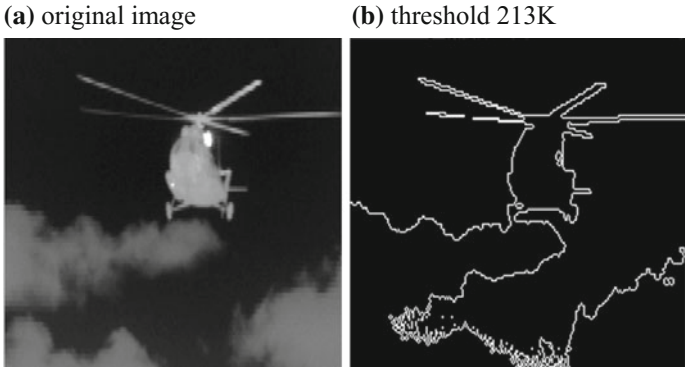


**Fig. 17** Detection of closed areas—temperature threshold 213 K

consisting of: detection, classification, recognition and identification. In preprocessing the methods of edge detection are commonly used to detect spatial discontinuity. It is quite an effective way to get a quick effect by indicating interesting elements in the image without significant degradation of image. Edge detection than can help operator to identify shape of the object.

For objects detection one can use some simple image processing methods such as: binarization, edge detection, skeletonization, and morphological transformations such as erosion dilation composed together to form more complex algorithms. Examples of use of the above operations for the thermographic images were presented in the Figs. 16, 17, 18 and 19.

Then, on the basis of images resulting from pre-processing operations, potential objects has to be separated. Separation involves the division of the image into fragments corresponding to different detected objects.

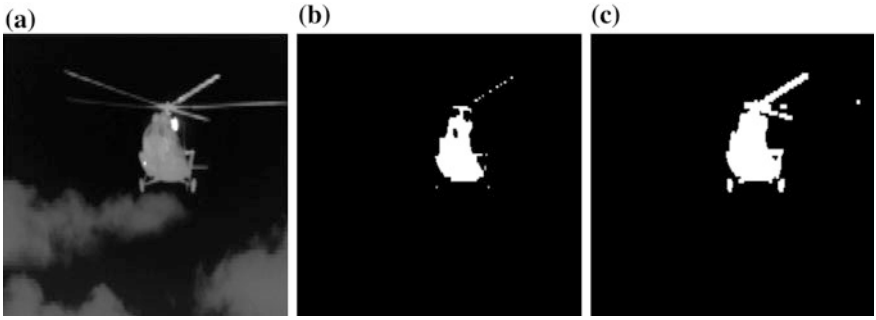


**Fig. 18** Examples of use in the thermographic image (**a**) morphological operators like erosion (**b**) and dilatation (**c**)

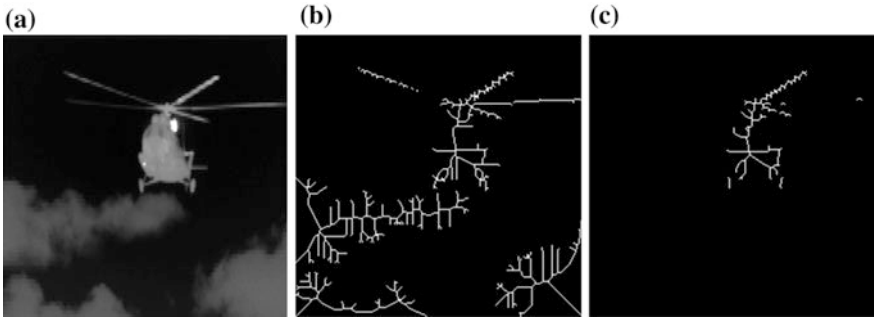


**Fig. 19** Infrared image skeletonization **a** for different threshold temperature: 240 K (**b**), and 280 K (**c**)

## *5.1 Object Detection Method*

Object detection involves performing a series of operations on the thermal image to highlight the image interesting, from the point of view of its appointed task. The developed method for objects detection performs the task by recording the thermal image and calculating the reference image. Then a comparison is made between newly acquired thermographic image and the reference image. Object detection is then performed by discriminating changes from the background image. Simplified diagram of the algorithm to detect objects in the infrared image is shown in Fig. 20.

According to the algorithm, the first step in the analysis is the determination of the initial reference picture. Determination of the reference image $f^*$ s based on averaging a certain number of images for which there is no object detected in accordance with formula:

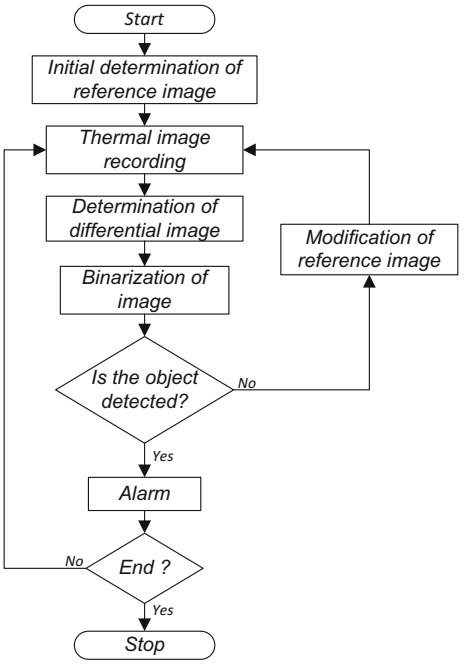


**Fig. 20** Simplified diagram of the algorithm to detect objects in the infrared image

$$f^*(m,n) = \frac{1}{K}\sum_{k=1}^{K} f_k(m,n). \tag{4}$$

where $f_k(m,n)$, $f^*(m,n)$—pixel value respectively of the source image and the resulting image placed in m–th row and n–th column in k–th picture in the sequence.

After determination of the reference image, algorithm goes to determination of the different image $\Delta f$ for each newly recorded image. A differential image is calculated according to the equation:

$$\Delta f(m,n) = \left| f(m,n) - f^*(m,n) \right|. \tag{5}$$

The next operation is to check whether the determined value of a differential image for each pixel falls into the detection threshold. If the calculated value of the differential image is greater than a predetermined detection threshold L then the object is detected. If the value of the differential image is below the detection threshold, the detection did not occurs and the recorded image is used to adaptively modify the reference image.

Probability of detection ($P_w$) and probability of false alarm ($P_f$) was determined in experiment along with their relationship to threshold level. Minor threshold level cause the background elements in the image to be classified as the detected object. With high detection threshold some elements of an object are not detected what

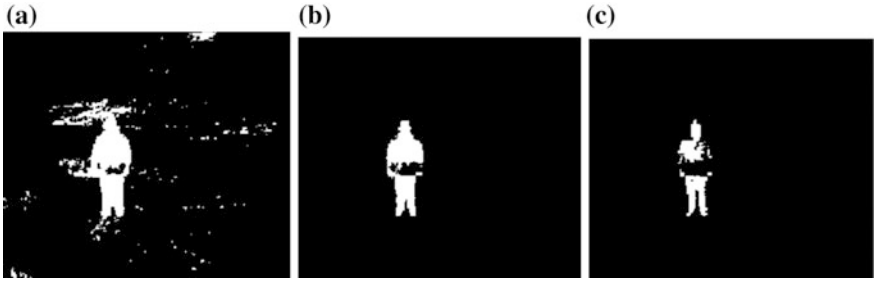


**Fig. 21** Objects detection results for threshold values of 1.0 °C (**a**), 3.3 °C (**b**), 6.0 °C (**c**)

**Table 1** Summary of the probability of object detection ($P_w$) and false alarm probability ($P_f$) at different distances from the object and different values of the proportionality coefficient $\alpha_P$

| Proportionality coefficient | Distance 160 m | | Distance 50 m | | Distance 20 m | |
|---|---|---|---|---|---|---|
| | $P_f$ | $P_w$ | $P_f$ | $P_w$ | $P_f$ | $P_w$ |
| 0.50 | 0.063 | 0.888 | 0.385 | 0.794 | 0.110 | 1 |
| 0.55 | 0.063 | 0.833 | 0.308 | 0.692 | 0.110 | 1 |
| 0.60 | 0.063 | 0.770 | 0.077 | 0.666 | 0.110 | 1 |
| 0.65 | 0 | 0.580 | 0 | 0.590 | 0 | 1 |

causes that interesting object is distorted and blurred. Exemplary results of detection for different values of the detection threshold are shown in Fig. 21.

The most important factor influencing the high probability of detection achieved with simultaneous low probability of false alarm is the right choice of detection threshold. To get the best performance the detection threshold L is adaptively determined according to the following formula:

$$L = \alpha_P \cdot \Delta f_{MAX} \tag{6}$$

where: $\alpha_P$—proportionality coefficient, $\Delta f_{MAX}$—maximum value of the pixel in differential image.

Proper selection of the proportionality factor is essential to correctly identify the potential location of an object, and extract more of its components and thereby reduce the number of false alarms. The results of research carried out for the set of the objects and different proportionality factor is shown in Table 1.

## 5.2 *Objects Recognition by Means of Radial Shape Function*

One of the object recognition methods is an algorithm using the so-called. radial shape function [12]. This method has some similar properties to the human vision

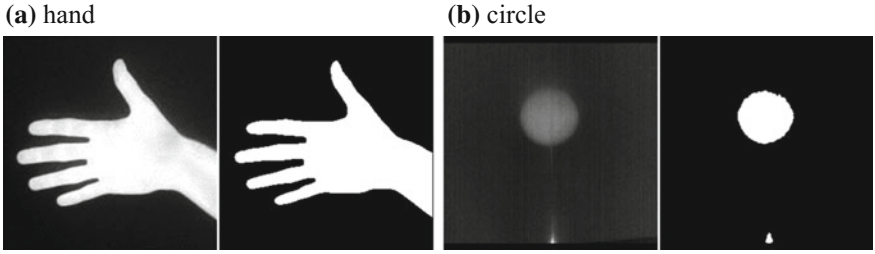


**Fig. 22** Thermovision image (on *left*) binarized and filtered result (on *right*) for two objects of different shape

and pattern recognition capabilities. The fundamental advantages of the method include immunity to distance, location and size change of the recognized object. The effectiveness of shape recognition is independent of the shape rotation. Moreover, this method has a relatively high recognition rate with low computational demands. This makes if a good candidate to embed in infrared cameras for image analysis in real-time.

The first operation in is binarization carried out in such a way that the detected object is represented by a value of one (white) and the background by zero (black). After binarization some distortions may occur specially on the edges of objects. In order to eliminate the noise there has been erosion operation applied. Sample results of such preprocessing are shown in Fig. 22.

The main part of the algorithm involves determination of a function describing the shape of the object called the radial shape function (RSF). First stage of RSF determination consists of describing the object in polar coordinates. Center of polar coordinate system is set to the statistical center of gravity of the object. According to the theory of statistical moments coordinates of the center of gravity (centroid) can be described as [12]:

$$x = \frac{M_{1,0}}{M_{0,0}}, \quad y = \frac{M_{0,1}}{M_{0,0}}$$

where:

$$M_{p,q} = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} x^p \cdot y^q \cdot f(x,y),$$

$f(x,y)$—binary image value at the point (x, y).

After calculating the position of the center of the coordinate system, the radial shape function $f_{RSF}$ is being calculated. This is done by determining the coordinates of points belonging to the edges of the object in polar coordinates according to equation:

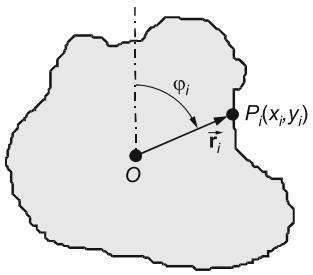


**Fig. 23** Method of determining the polar coordinates for successive points on the objects edge

$$f_{RSF}(\varphi) = |\vec{\mathbf{r}}_i| = \sqrt{x_i^2 + y_i^2} = \frac{x_i}{\cos \varphi_i} \tag{7}$$

where: $(x_i, y_i)$ are the coordinates of successive points on the edge of the object in relation to the center of gravity, while $\varphi$ is the angle between point on the edge and the vertical axis. Determined in this way radial shape function expresses the distance between the center of gravity of the object and its edge as a function of angle. The described method of radial shape function determination is illustrated in Fig. 23.

In the next step of the object recognition algorithm the previously determined radial shape function is being low pass filtered. This operation is introduced to eliminate noise caused by the irregular edges of the object and noise. Then the radial shape function is subjected to a discrete Fourier transform [12] according to the equation:

$$F_{RSF}(\Phi_k) = \left| \sum_{n=0}^{N-1} f_{RSF}(\varphi_n) e^{-j2\pi kn/N} \right| \tag{8}$$

The obtained spectrum of radial shape function $F_{RSK}$ is normalized to the mean value which can be written using the equation:

$$F^*_{RSF}(\Phi_k) = \frac{F_{RSF}(\Phi_k)}{F_{RSF}(0)} \tag{9}$$

Based on the obtained in the above manner normalized spectrum the similarity score is determined. Similarity score $\omega_k$ is defined as sum of differences between individual spectral components $F^*_{RSF}$ and the spectrum of reference object $W_k$:

$$\omega_k = \sum_i \left| F^*_{RSFi} - W_{ki} \right| \tag{10}$$

After determining the similarity scores for all reference vectors, system is taking the decision to recognize the object. Classification decision is made by selecting object pattern with smallest similarity score. In the Table 2 there are values of the

**Table 2** The values of similarity scores for selected objects obtained during the algorithm testing

| Pattern | Object on the picture | | | |
|---|---|---|---|---|
| | Circle | Hand | Square | Triangle |
| Circle | **0.37258** | 6.1515 | 0.68999 | 2.7068 |
| Hand | 4.6372 | **0.90317** | 3.0745 | 2.6486 |
| Square | 0.85206 | 5.338 | **0.6428** | 2.6556 |
| Triangle | 3.3157 | 3.3243 | 2.3643 | **0.8632** |

similarities shown, obtained for the chosen objects. Similarity scores that determined the recognition of the object has been highlighted in the table.

## *5.3 Object Tracking Algorithm*

After successful target detection and recognition, the system can track the movement of the object of interest. Object tracking is an image processing procedure of finding chosen object on the following frame using knowledge about its position in previous frames. This means that the object tracking method differs from previously described image processing techniques by the fact that they analyze multiple frames in time. Among many tracking algorithms one can distinguish methods with different levels of complexity, different computational demands and varying execution time. During the development of the method we have focuses on the tracking algorithms, which enables real time operation and can be implemented in a low power digital signal processor or FPGA. Having all this in mind the Sum-of-the-Squared Differences algorithm was chosen [13].

Gradient based methods like Sum-of-Squared-Differences [14, 15] localize targets by analyzing differences between consequent frames. Finding target movement is performed by searching minimum of cost function in space and time. Cost function in this approach is a sum of squared differences. Sum of squared differences coefficient is a measure of difference between two fragments of images. Both fragments of images should have equal size. Assuming, that the two fragments are $(2h + 1)$ by $(2h + 1)$ in size and that they centers have the coordinates $(x, y)$ and $(u, v)$ respectively, the SSD coefficient can be calculated according to the following relation:

$$SSD = \sum_{\substack{i \in (-h,h), \\ j \in (-h,h)}} \left\{ [f_{k-1}(x+i, y+j) - f_k(u+i, v+j)]^2 \right\} \tag{11}$$

$i, j$—point coordinates with respect to the centers of compared fragments.

If we assume that the tracked object is present on the $f_{n-1}$ frame and is centered around $(x, y)$ coordinates, then finding this object on the consecutive frame means, that the point $(u, v)$ is to be found, for which the SSD coefficient has minimal value.

Then the point $(u, v)$ becomes the center of the tracked object on consecutive $f_n$ frame. The search for the minimal value of SSD coefficient is performed in the neighborhood of the past location of tracked object. The general idea of finding an object using the Sum-of-the-Squared Differences method is presented in Fig. 24.

Fragment containing found object became a new model for further search in following frame. Replacing old model with the fragment of picture containing a newly found one is called the model update. In traditional version of algorithm the model update is made every frame.

This approach can lead to some undesired effects. In case where object tracking will fail for one frame, algorithm will forget the object. After that, tracking of this object will not be possible, because the proper model has become obsolete automatically. That is why SSD algorithm has low long time reliability. Noise in one frame only, can cause the whole algorithm to fail in further frames. That is why there is a need to develop some new special routines to make this algorithm immune to partial or full occlusions, noise and changes of objects appearance. To distinct reliable position estimation from noisy one the special SSDVar coefficient was developed:

$$SSDVar = \sum_{i \in \langle -h,h \rangle, j \in \langle -h,h \rangle} \left[ (SSD[i,j] - \min(SSD[i,j]))^2 \right] \tag{12}$$

This coefficient can be used to evaluate the quality of objects localization. High SSDVar indicates that the minimum of SSD coefficient was clear. Low SSDVar indicates that the estimated localization is not so reliable. This coefficient was used to determinate the model update procedure. When the SSDVar parameter was higher than arbitrary threshold, model update procedure is made like in traditional version of SSD algorithm. When SSDVar is lower than the threshold, the model update procedure is skipped. This prevents the algorithm to forget the object model when the new estimation of object localization is unreliable.

Exemplary plots of SSD coefficient map for disrupted and non-disrupted object is shown in Fig. 25.

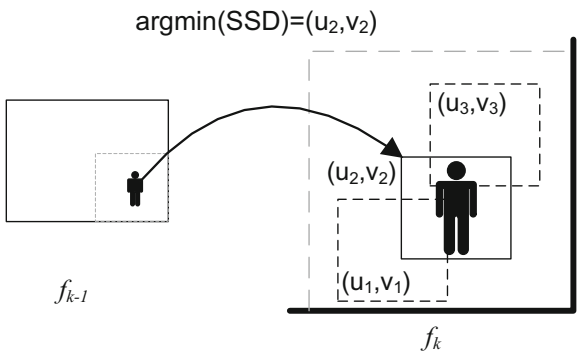


**Fig. 24** Localizing the object by finding the area for which the SSD coefficient is the smallest

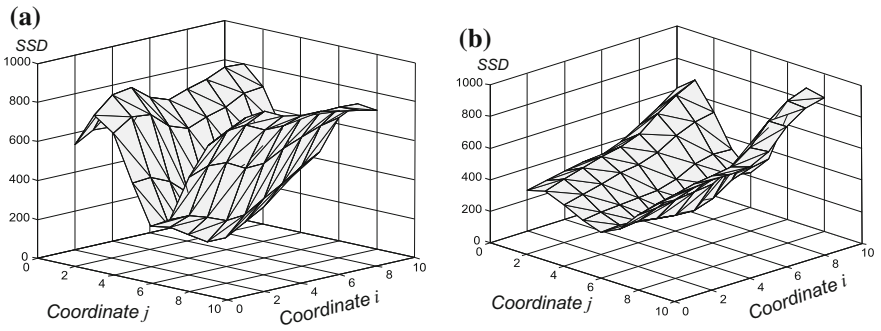


**Fig. 25** Charts of SSD values in the neighborhood of the object when the object is undisturbed (**a**) and, when disturbed (**b**)

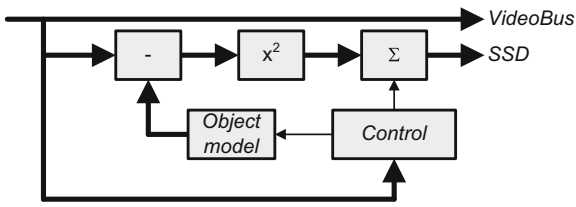


**Fig. 26** The block diagram of the SSD coefficient calculation module

**Fig. 27** Obtained trajectory of objects using developed tracking method in the infrared image

Object tracking algorithm according to the proposed method is quite computationally complex and requires relatively high computing power. Furthermore, it is desirable that tracking module introduced the shortest possible delay in the signal processing path. One way to speed up the operation of the system is to provide specialized hardware module performing the calculation for the proposed algorithm. That is why the data processing algorithms have been implemented and tested in the FPGA structure. This approach provides high performance thanks to multiple acceleration techniques like pipelining and parallelism. A simplified diagram of a module for calculating the SSD coefficient for the data stream is presented in Fig. 26.

The results of obtained trajectory of objects using developed tracking method in the infrared image is shown in Fig. 27.

## 6 Conclusion

The article presents the general structure of a thermal imaging camera and shows the image processing operations performed in the observation of infrared cameras that are used in security systems, detection and recognition. Provided examples of processing operations for thermal images and their implementations in the system addresses specifics of infrared image processing that differs from commonly used visual image processing. High processing power demands for image processing techniques used in real-time infrared cameras demands usage of hardware accelerated modules especially when infrared camera has to be used in security and control applications, where data latency is a critical issue.

Presented by the authors developed methods for detecting and tracking objects on the thermographic images has been a subject to experimental verification. Based on these results, it turned out that a detection method is characterized by a high probability of detection at a relatively low probability of false alarm. Results of operation were dependent on applied internal parameters so the adaptive functions were described like adaptive threshold of detection. Developed method of object tracking allows effective determination of the trajectory of moving objects in thermographic images. Method was particularly robust to low image resolution typical for infrared cameras and blurred edges of objects. When implemented in a digital system based on the structure of the programmable FPGA both methods are performed in real time. The combination of both methods in one system allows for effective support for decision-making.

## References

1. Gonzalez, R. C., & Woods, R. E. (2002). *Digital image processing* (2nd ed.). Prentice-Hall.
2. Holst, G. C. (1998). *Testing and evaluating of infrared imaging systems*. SPIE Optical Engineering Press.
3. Sosnowski, T., Orżanowski, T., Kastek, M., & Chmielewski, K. (2007). Digital image processing system for thermal cameras. In *Advanced Infrared Technology and Applications AITA 9*, Leon, 8–12.10.2007.
4. Sosnowski, T., Bieszczad, G., Kastek, M., & Madura, H. (2010). Digital image processing in high resolution infrared camera with use of programmable logic device. *Proceedings of SPIE, 7838,* 78380U.
5. Bieszczad, G., Sosnowski, T., Madura, H., Kastek, M., & Bareła, J. (2010). Adaptable infrared image processing module implemented in FPGA. *Proceedings of SPIE, 7660,* 76603Z.
6. Bieszczad, G., Sosnowski, T., Madura, H., Kastek, M., & Bareła, J. (2011). Image processing module for high-speed thermal camera with cooled detector. *Proc. SPIE, 8012,* 80120L.
7. Orżanowski, T., Madura, H., Kastek, M., & Sosnowski, T. (2008). Nonuniformity correction algorithm for microbolometer infrared focal plane array. In M. Strojnik (Ed.), *Advanced infrared technology and applications 2007* (pp. 263–269). Leon: Mexico.

8. Krupiński, M., Bieszczad, G., Sosnowski, T., Madura, H., & Gogler, S. (2014). Non-uniformity correction in microbolometer array with temperature influence compensation. *Metrology and Measurement Systems, XXI*(4), 709–718.
9. Dulski, R., Powalisz, P., Kastek, M., & Trzaskawka, P. (2010). Enhancing image quality produced by IR cameras. *Proceedings of SPIE, 7834,* 783415.
10. Dulski, R., Madura, H., Piatkowski, T., & Sosnowski, T. (2007). Analysis of a thermal scene using computer simulations. *Infrared Physics and Technology, 49*(3), 257–260.
11. Accetta, J. S., & Shumaker, D. L. (1993). *The infrared and electro-optical systems handbook*. Bellingham, WA: Ann Arbor MI and SPIE Press.
12. Bieszczad, G. (2005). Metoda rozpoznawania wzorców w obrazie graficznym, VI Międzynarodowa Konferencja Elektroniki I Telekomunikacji Studentów i Młodych Pracowników Nauki, SECON 2005, 8–9.11.2005 Warszawa.
13. Bieszczad, G., & Sosnowski, T. (2008). Real-time mean-shift based tracker for thermal vision. In *Quantitative InfraRed and Thermography QIRT 2008 Conference*.
14. Hager, G., & Belhumeur, P. (1998). Efficient region tracking with parametric models of geometry and illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*(10), 1025–1039.
15. Venkatesh Babu, R., Patrick, P., & Patrick, B. (2007). Robust tracking with motion estimation and local Kernel-based color modeling. *Image and Vision Computing, 25*, 1205–1216.