AUTOMATYKA • 2011 • Tom 15 • Zeszyt 3

Tomasz Jaworski*, Jacek Kucharski**

Preprocessing and Clusterization of Thermal Images of Induction Heated Steel Cylinder

1. Introduction

Monitoring and control algorithms of various technological processes often require many input signals and other information if ever growing quality requirements are to be satisfied. Among such signals, in many cases very important one, is the temperature of in-process objects or moving parts. For example, in paper industry [1], the temperature of dryers and calandering steel rollers are crucial for controlling of a paper moist-temperature balance which affects overall process efficiency and final product quality. Moreover, current state of the art in paper production technology, allowing high speed product output up to 120 km/h [2], requires effective and reliable control algorithms, design of which becomes more and more complicated.

Paper drying and calandering, as the last steps that affect the final product quality, can be implemented as hot air dryers and steam heated cylinders [3] or as microwaves application [4]. However, the idea behind each of them is to sustain stable drying temperature and moisture profile of the output paper by appropriate control of power and position of heat sources.

A disparate approach to the problem of the plant heating in paper industry have been recently discussed by Kucharski, Frączyk and Urbanek on a basis of spinning steel cylinder (as roller), heated by a set of inductive heaters mounted in fixed positions just above the cylinder surface [5–8]. It was proven that the temperature of such a cylinder can be controlled by a closed loop system influencing the electrical power of each inductor while the surface temperature is measured by a set of infrared thermocouples rigidly located along the axis of the cylinder (Fig. 1a). Authors discuss dynamic properties of heat transfer with inductive heating, control models and issues of contactless temperature measurement with emissivity correction algorithms [9] for heterogeneous surfaces. Despite many advantages, some weak points of the method were also pointed out. The most important drawbacks result from the static character of both the heat sources (inductors) and the temperature

* Ph.D. Student, Technical University of Lodz, Computer Science Department, Lodz, Poland
** Technical University of Lodz, Computer Science Department, Lodz, Poland

sensors. The first one causes inevitable non-uniformity of temperature profile along cylinder axis while the latter significantly reduces information about the thermal state of the cylinder available to the control system. Therefore the next step in the development of the method is to make the inductors able to move and to replace spot temperature sensors by IR camera gathering information about the temperature of the whole cylinder surface. It obviously yields several new tasks to be realized within the control system, some of them belonging to the image processing area.

In this article authors introduce a method for processing of gray-level images acquired from thermo-vision IR camera, which portray instant surface temperature distribution. Such an approach, in connection with moveable inductors, opens a new field of research on algorithms that will utilize holistic temperature measurement for control purposes.

The organization of this paper is as follows. Section 2 describes mechanical and electrical components used in experiment to obtain thermal images. Section 3 is devoted to synchronization between different components described previously. Section 4–7 introduces the clusterization algorithm as a set of independent image processing steps in which output of preceding step can be considered as an input for. Section 8 describes proposed clustering algorithm and Section 9 presents obtained results. Finally Section 10 carries a discussion on advantages and disadvantages of presented algorithm.

a) b)

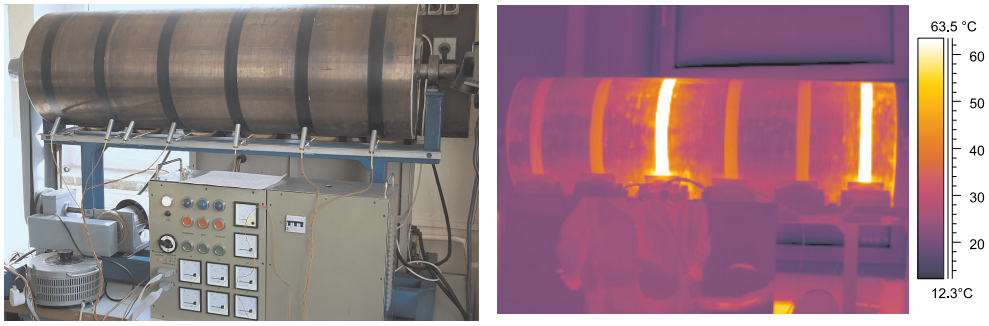

Fig. 1. The steel roller with pyrometer sensors and heating control panel (a), an image from infrared thermovision camera FLIR T335 with induction heaters visible (b)

2. Mechanical and electrical components of semi-industrial experimental setup

The central part of the experimental setup used is a steel cylinder of 60 cm diameter, 120 cm length and 5 mm thick. It is driven by an asynchronous 1,5 kW three-phase motor with SJ200 inverter (Fig. 1a) connected to a PC with RS485 line on which MODBUS protocol is employed. The surface of the cylinder is heated by a set of six inductive exciters [7]. Each of them is supplied by a power amplifier with PWM control [10] and its position is referenced by a PC controller workstation.

Instead of a spot measurement, as previously mentioned, a thermovision IR camera is used as shown in Figure 1b. It is connected to the PC workstation and positioned in such a way that it allows the system to monitor temperature distribution on visible part of cylinder's surface.

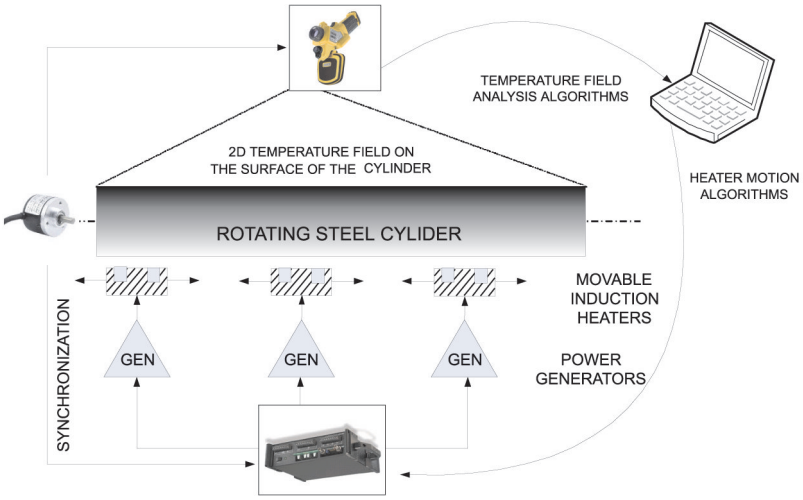

Fig. 2. The setup for thermal images acquiring system

3. Component synchronization

Due to specific character of the heated plant which is a spinning cylinder, its angular position at any given time becomes a vital information for synchronization of IR camera recordings with power dispatch and position of every heater.

To obtain current position of the cylinder, together with its velocity, an electronic synchronization device was built. It utilizes a DSP microprocessor, namely a Microchip dsPIC33FJ128MC706A with 16-bit architecture and a differential quadrature encoder attached to the shaft of the cylinder.

The device interfaces with a PC via Universal Serial Bus communication link and with IR camera with use of standard I/O digital signals. The PC receives a steady stream of information that consists of: a time stamp of every measurement, a current position of the shaft in pulses (with resolution of 2×10^4 ppr[1]) and a velocity of the shaft. Together with PC communication process, the IR camera is triggered by synchronization impulses generated by the device each time the spaced markers on the perimeter of the cylinder pass mechanical reference point[2].

[1] Pulses per revolution

[2] A mechanical reference point (or zero-angle point) is an arbitrary selected point on the cylinder's perimeter that we assume it is angle 0.

The encoder works in Index mode which results in a synchronization impulse once per revolution – a mechanical reference point. This feature will be necessary in further work on emissivity correction where the surface will be covered by an emissivity map in which each point is strictly related with specific physical area of the cylinder's surface.

4. Image processing and clusterization

The presented algorithm can be considered as a set of three independent steps of image processing executed in given order (Fig. 3). Last step is the clusterization algorithm supplied by the image with corrected geometry. Of course, not every step is required. However for the purpose of described experimental setup it is necessary.

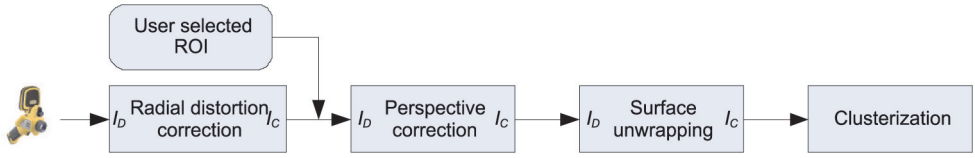

Fig. 3. An overview of the algorithm presented in the paper

All of the three processing steps have its own input and output. Input image for each of them is denoted as I_D (a distorted image). In the case of the radial correction block, it is an image directly received form IR camera. After processing the distorted input image I_D, each block exposes on its output corresponding corrected image I_C. Therefore in the case of processing pipeline depicted in Figure 3, a corrected image I_C from radial correction block is the distorted image I_D for perspective correction block. The same is the case for surface unwrapping block. Finally, an image that was processed in sequence by each of three blocks is given as an input for clustering algorithm. The final output, after the clusterization block, consists of a set of Gaussian functions Γ calculated in (8.7) which can be interpreted as a temperature clusters.

During the development of the algorithm we decided to perform all operations on selected region, **ROI** (region of interest). Since every image, irrespective of its content, should be processed by the radial correction block, a decision was made to place the ROI selection rectangle after the first step. In each of the following images, the ROI is visualized as a white rectangle in selected part of an image. The coordinates of ROI are user-selected. It is also important to mention, that during each correction process a corrected pixel that does not fit the rectangle is rejected (not written to the corrected image).

5. Correction of barrel distortion in the input image

The radial distortions are well-known phenomena accompanying almost every image processing task where the source image is acquired from a digital camera, especially with

wide-angle lenses. Radial distortion displaces every image point further or closer from the correct[3] point along radius line between the central point of the distortion and the correct point [11].

A positive displacement, when distorted point is further from the centre of distortion then the correct point is known as barrel distortion (Fig. 4a). A negative displacement, when distorted point is closer to the centre of distortion than the correct point, is known as pincushion distortion (Fig. 4b). Former is the distortion inherent in IR camera images and its idea is depicted in Figure 4c where distorted point P_D is further from the centre than the correct one P_C.

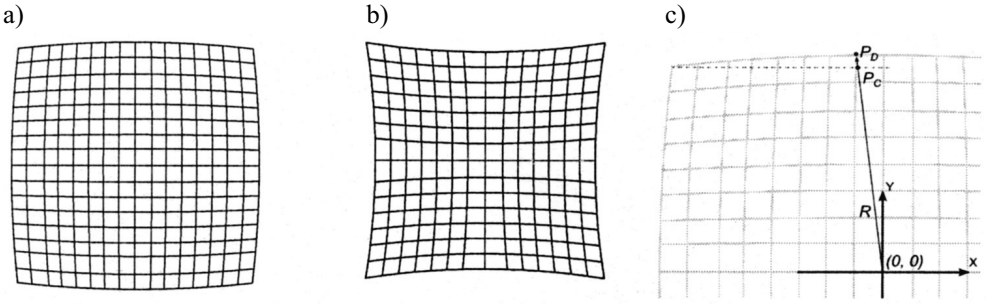

Fig. 4. Radial distortions of rectangular grid: a) barrel distortion; b) pincushion distortion, c) the idea of the barrel distortion, where distorted pixel is further then the original one

To diminish the existing barrel distortion we made the following assumptions (see Fig. 4c): point $P_D = [x_D\ y_D]^T$ is an arbitrary point in the distorted image frame I_D acquired from IR camera with resolution 640×480. Every point P_D contains a value of measured temperature, normalized for the whole image, thus in the range [0; 1]. Point $P_C = [x_C\ y_C]^T$ is a corresponding to P_D point in corrected (without distortion) image I_C, also with resolution 640×480. Both images are then translated into two-dimensional unit area; the central pixel is located at [0; 0], upper left at [–1; –1] and lower right at [1; 1].

The classical radial distortion model [12] delivers the following equation for approach that employ mapping from ideal, undistorted image coordinates (point P_C) to corresponding coordinates in distorted image (point P_D) [11]:

$$P_D = P_C \frac{1}{1 + \|P_C\|_2^2 \kappa_1 + \|P_C\|_2^4 \kappa_2} \tag{5.1}$$

Expression $\|\cdot\|_2$ given in (5.1) is the L^2–norm – a distance between P_C and center of the image. Both κ_1 and κ_2 are distortion coefficients. For convenience we assume that points

[3] Correct point in an image without distortion corresponding to one or more points in distorted (original) image

with coordinates [0; 1], [0; –1], [–1; 0], [1; 0] in distorted image I_D are mapped to the same coordinates in corrected image I_C. To achieve this, the (5.1) has to be scaled with coefficient $s = 1 + \kappa_1 + \kappa_2$ which alters the numerator of (5.1) as follows:

$$P_D = P_C \frac{1 + \kappa_1 + \kappa_2}{1 + \|P_C\|_2^2 \kappa_1 + \|P_C\|_2^4 \kappa_2} \tag{5.2}$$

As the result of such computation, we arrive at corrected image (Fig. 5b) from distorted image (Fig. 5a). Both distortion parameters κ_1 and κ_2 are determined experimentally as κ_1 is set to 1/45 and κ_2 is set 0. Second parameter was assumed 0 from the beginning as simplification and due to the fact that value of zero (reduced order of the distortion model) does not significantly affects corrected image quality [13].

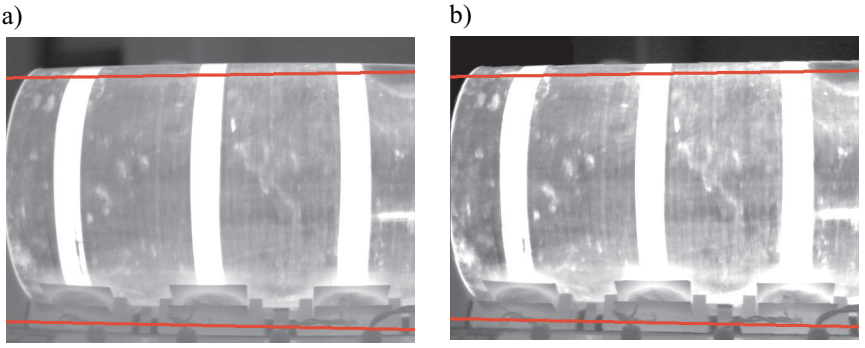

Fig. 5. A sample shot taken form IR camera: a) with visible barrel distortion; b) corrected image

The value of κ_1 was obtained by a simple iterative algorithm, that for a given value of κ_1 in range $(0; 1\rangle$ (where 0 gives no image alteration) detects the upper edge of the cylinder (Fig. 5a, b) and calculates the collinearity for detected edge points. The highest value of collinearity (best fit) selects the value of κ_1.

6. Perspective correction

Along with geometric distortions, caused by the camera lenses and quite easy to diminish, there are two effects that will also be treated as pixel coordinates distortions and dealt with as such.

First type of distortion is the perspective projection in which a three-dimensional object (in this case a horizontally placed cylinder) is projected onto a two-dimensional image surface. Due to the curvature of the cylinder's surface the horizontal mid area is closer to the camera then the rest of the visible surface. As a result the two strips placed parallely along the cylinder circumference appear to be bent outward the center of the image, as depicted in Figure 6b.

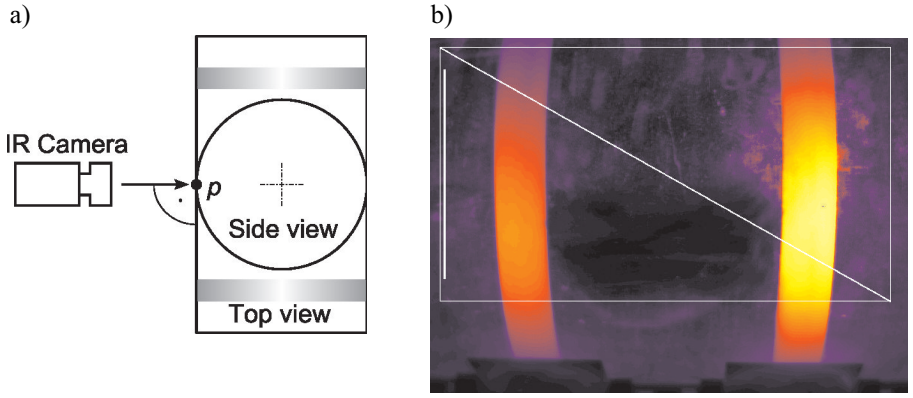

Fig. 6. Location of IR camera (a) and sample of acquired image (b). Vertical and diagonal lines are placed artificially to show the influence of correction algorithms

Since camera's view axis is positioned perpendicular horizontally and vertically to the surface tangent (as shown in Fig. 6a), an empirical model of pixel distortion was implemented to diminish the influence of perspective on the final image.

The correction algorithm operates on a set of pixels from input/distorted image I_D bounded by a ROI (the white rectangle in Fig. 6b) defined by the height (H) and width (W) on the distorted image I_D (Fig. 7a). Each pixel from ROI is denoted as distorted pixel P_D.

When IR camera is positioned as described above, point p in Figure 6 should coincide with center of the ROI rectangle (located at C_x, C_y in I_D) denoted as p'. This causes the center of ROI to be located in perpendicular tangent point p. Such compliance between the physical setup and the image is not required; nevertheless it gives the best results since cylinder is bent symmetrically to the abscissa.

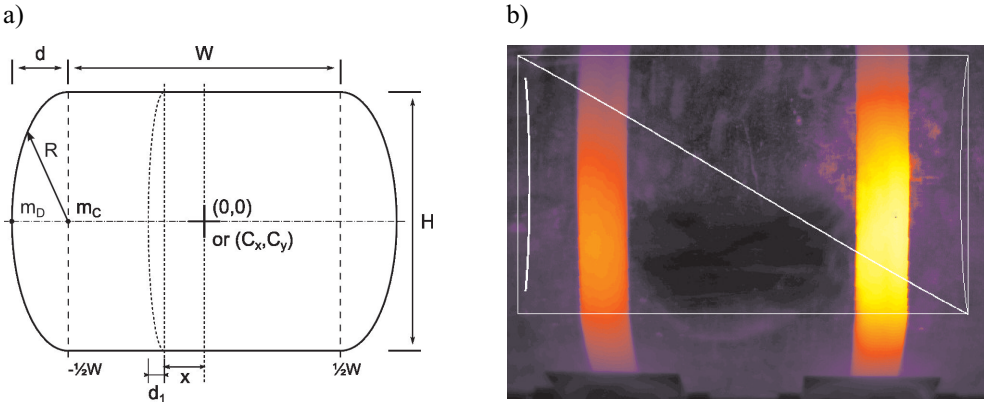

Fig. 7. Parameters of perspective correction model (a), the result of perspective correction algorithm (b)

Radius R is selected in such a way that it gains best fit to the circular edges of the cylinder or in the case depicted in Figure 6b to the outer edges of bright stripes. Coefficient d determines the strength of the distortion for the extreme values of $x \in \{-½W; ½W\}$ as shown in Figure 7a. In mid cases, d is proportionally scaled by x (6.2).

Finally, a corrected pixel P_C is obtained by translating coordinates of distorted pixel P_D as shown in (6.1). The values of x and y are relative to the position of the ROI and changes in range $\langle 1; W\rangle$ and $\langle 1; H\rangle$ respectively. In the case of $P_D(x+a_{x,y}, y)$, coordinates can point outside the window. If, however, $P_D(x+a_{x,y}, y)$ crosses the image boundaries, value of such non-existing pixel is assumed to be 0.

$$P_C(x, y) \leftarrow P_D(x + a_{x,y}, y) \tag{6.1}$$

Component $a_{x,y}$ can be interpreted as horizontal correction value that depends on distorted pixel's row and column. For instance: let (x, y) equals $(-½W, 0)$ which is denoted in Figure 7a as correct point m_C. Then $a_{x,y} = -d$ which is denoted as distorted point m_D. Value of d is depicted in Figure 7a.

$$a_{x,y} = d\,\frac{x}{0.5W}\,\frac{\sqrt{R^2 - y^2} - \sqrt{R^2 - 0.25H^2}}{R - \sqrt{R^2 - 0.25H^2}}; \quad R \geq \frac{H}{2} \tag{6.2}$$

The example image shown in Figure 6a was used for tests of the proposed correction algorithm. To make the results more readable a thick vertical line near the left edge of the image was added to the image just before the perspective correction was performed. Figure Figure 7b shows same image after correction. Bright vertical stripes were straightened and the white vertical line, located at arbitrary x value was bent accordingly to the parameters d, R and its position.

7. Surface unwrapping

Second kind of distortion is caused by the natural curvature of the surface of cylinder. Final version of the image, usable for further processing and analysis, requires the surface to be 'unwrapped'.

In contrast to described above pixel-oriented distortion corrections, the following is a row-oriented one. An image with corrected perspective according to (6.1) can be considered as an image with distorted y coordinate of each row. Again, similar coordinate conversion expression (7.1) can be used, where R_C is the set of rows in corrected image I_C and R_D is the set of rows in distorted image I_D.

$$R_C(y) \leftarrow R_D(C_y + row_y) \tag{7.1}$$

The value of row_y is an y coordinate (relative to C_y) of distorted row. Its is calculated via (7.2) and points a pixel row in distorted image which has to be placed under y coordinate in corrected image as in Figure 8.

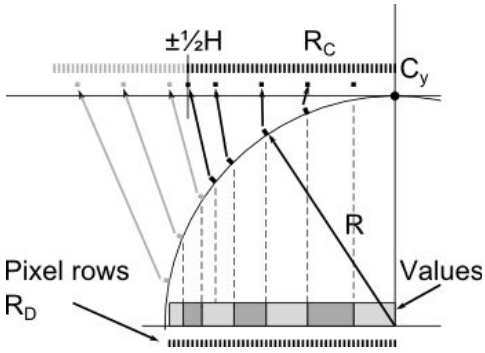

Fig. 8. Idea of unwrapping; a pixel row should be interpreted as a line of pixels perpendicular to the page surface

Value of R is the radius of the cylinder viewed by the IR camera. The 1/R coefficient in (7.2) stands for '*amount of radians per one row of pixels*'.

$$row_r = R\sin\frac{1}{R}\left(r - \frac{1}{2}H\right) \tag{7.2}$$

Finally, after applying both corrections (described in section 6 and 7) to the input image, the result is similar to that shown in Figure 7b. Of course, the input image should be previously processed by radial distortion correction routine.

a) b)

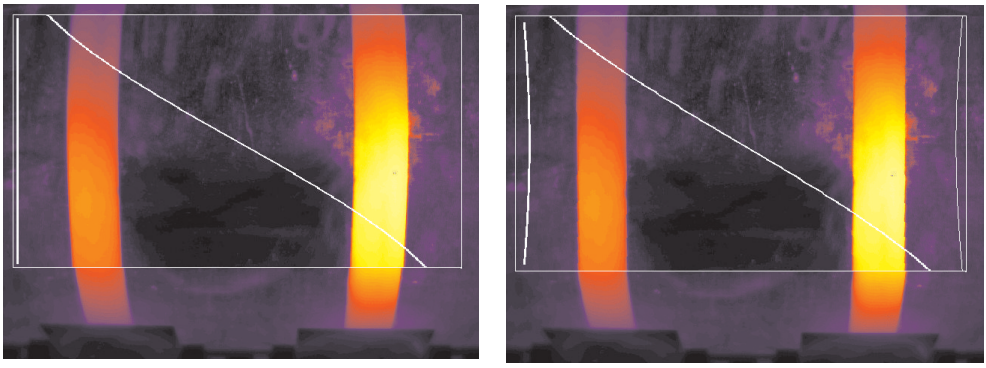

Fig. 9. Results of correction algorithms correction effect:
a) only unwrapping; b) perspective correction and unwrapping

A sample image depicted in Figure 6b is shown again in Figure 9a, but after unwrapping operation only. It should be noticed that straight diagonal line is bent accordingly to the surface curvature. The final result of applying both steps is depicted in Figure 9a. It is clearly visible, that both artificial lines are now 'corrected' according to distortion model.

8. Clusterization

The main idea behind proposed clusterization algorithm is to describe a thermal image of the steel cylinder surface as a set of fuzzy objects – a kind of 'temperature clusters'. This way of proceeding was previously mentioned by the authors in [14]. Such a set of fuzzy clusters indicating cooler and warmer areas of a surface can be used together with inferring methods based on fuzzy spatial relations between fuzzy objects [14] to arrive at robust algorithms which enable to control both position of movable heaters and amount of power dispatched by them on the surface of steel roller, as depicted in Figure 2.

For this purpose the Mountain Method (MM) [15] was explored. It was introduced by Yager and Filev with further modifications [16]. The method was originally intended as a preprocessing method for obtaining initial number of clusters for FCM (Fuzzy C-means) algorithm, but it can also be used as a stand alone clustering algorithm. In this paper authors have utilized the idea of calculating modified Mountain Function for the input image. As a result a mountain surface is obtained based on which the proposed algorithm iteratively extracts temperature clusters from the original temperature image. The algorithm is described step-by-step in further text.

The main idea of MM is to create a grid of nodes in a domain space of points constituting input data set. Then a potential in each node N_i is calculated by so called Mountain Function (MF) given by (8.1).

$$MF(N_i) = \sum_{k=1}^{n} e^{-\alpha d(x_k, N_i)} \tag{8.1}$$

The N_i is i-th grid node, n is the number of points in data set, x_k is the k-th point from given data set and α is a constant chosen empirically whose value depends on the application of Mountain Method; it can be interpreted as a construction coefficient. The function $d(\cdot,\cdot)$ is the Euclidean distance.

It is important to mention, that there are two approaches to idea of the grid. First one – the basic one assumes grid nodes declared as points in a space with the same dimensionality as the data points. Latter, discussed in [17] assumes data points as grid nodes, hence the potential calculated by (8.1) is not in grid nodes but in the data points. This 'data points as grid nodes' approach was utilized in the presented algorithm.

After mountain function (8.1) is built, MM algorithm destructs MF as follows: A grid node N_i^* with highest value of $MF(N_i^*)$ is selected. Coordinates of such a node are memorized as a cluster initial position and the mountain function is destroyed as follows:

$$MF^s(N_i) = MF^{s-1}(N_i) - MF^{s-1}(N_i^*) \sum_{k=1}^{n} e^{-\beta d(N_i^*, N_i)} \tag{8.2}$$

Variable s determines current iteration and β is a constant chosen empirically – a destruction coefficient. The sequence of steps is carried as long as termination condition occurs (max number of clusters reached or ratio of $MF^{s-1}(N_i^*)$ to $MF^s(N_i^*)$ is less then δ).

Our proposal modifies the construction equation (8.1) in order to obtain a function called Mountain Surface (MS) (8.3). The MS function is defined on a normalized image I of width (W), height (H) and a set $\boldsymbol{P}$, containing coordinates of pixels . Coordinates of each pixel are denoted as P_{xy}, where $P_{xy} \in \boldsymbol{P}$; $x \in \langle 1; W\rangle$ and $y \in \langle 1; H\rangle$. Expression $I(x,y)$ denotes measured temperature at point x, y (at P_{xy}) and $I(x,y) \in \langle 0; 1\rangle$.

$$MS(P_{xy}) = \sum_{u=1}^{W} \sum_{v=1}^{H} I(u,v) e^{-\alpha d(P_{uv}, P_{xy})} \tag{8.3}$$

Main modification is done by adding a multiplication of exponent by $I(u, v)$. It was necessary, since we assumed to use a grid with same number of nodes and their coordinates as the image I. Lack of coefficient $I(u, v)$ resulted in a flat surface since α is constant through the whole process.

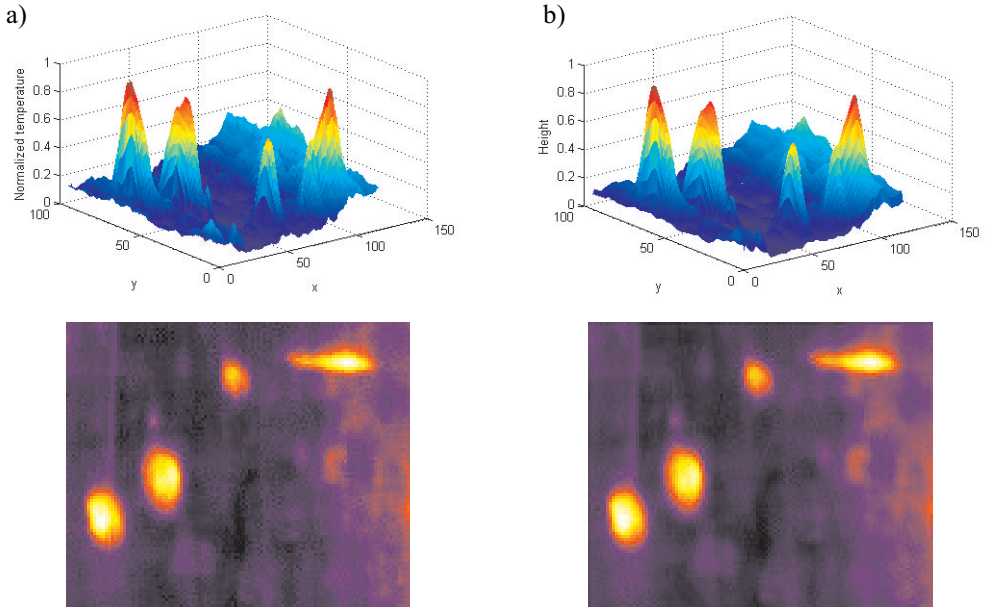

Fig. 10. An sample image from IR camera: a) original image with normalized temperatures; b) corresponding mountain surface with heights

Result of (8.3) can be observed in Figure 10. It is clearly visible, that the original image (Fig. 10a) contains noise inherent to temperature measurement. In this case, Mountain Function also works as a digital filter that smoothes source image.

After a mountain surface is built from an input image, our method begins extracting clusters, one by one in the following, iterative manner:

1) Assume iteration counter $s = 1$; MS^1 = previously calculated mountain surface and $I^1 = I$ (the input image from which the *MS* was acquired).
2) Find highest value of mountain surface MS^S with (8.4). Let assume that the point P_0 holds found x, y coordinates of the value and the value A itself.

$$(P_0, A) = \max_{P_0} MS^S(P_0) \tag{8.4}$$

Classical approach with two nested loops, iterating one by x and other by y, can be used.

3) Minimize goal function (8.5) where I^S is image in iteration s, P_0 is the point of highest mountain surface value, found in previous step; W and H are width and height of the image.

$$G(\{A,\sigma,\Delta P,k\})\big|_{I^S,P_0} = \sum_{u=1;v=1}^{W;H}\left(I^S(u,v) - Ae^{-\frac{1}{\sigma^2}\|P_0+\Delta P-(u,v)\|_2^2} - k\right)^2 \tag{8.5}$$

Optimization parameters: $\{A, \sigma, \Delta P, k\}$ are responsible for obtaining best fit of Gaussian kernel at point P_0. A is the initial normalized peak temperature that can be changed during optimization process, σ is the width of Gaussian function, and ΔP allows to eventually correct position of the Gaussian bell-shape on the image. The purpose of component k is to diminish static error due to different elevation of image and fitted function. Without passing k though as an optimization parameter, the minimization algorithm converges to meaningless parameters or does not converge at all. Finally, value of k can be ignored.

As the initial values for optimization parameters we proposed: {A from step 2; $\sigma = 1$; $\Delta P = (0, 0)$; $k = 0$}. During optimization, minimization algorithm was bound by arbitrary selected artificial boundaries: $\{A^{min} = 0$; $\sigma^{\min} = 10^{-3}$; $k^{\min}=0$; $\Delta P^{\min}=(-20, -20)\}$ and $\{A^{\max} = 10$; $\sigma^{\max} = 10^2$; $k^{\max} = 10$; $\Delta P^{\max} = (0, 0)\}$ however there was no case of optimization parameter to be equal to boundary value.

After optimization process is finished, calculate corrected version of P_0 as follows:

$$P = P_0 - \Delta P \tag{8.6}$$

4) Calculate image Γ of Gaussian function as in (8.7) and with same size as the I.

$$\underset{x\in\langle 1;W\rangle;y\in\langle 1;H\rangle}{\Gamma(x,y)} = Ae^{-\frac{1}{\sigma^2}\|P-(x,y)\|_2^2} \tag{8.7}$$

Obtained image Γ should be extracted from both: image I^S and mountain surface MS^S as in (8.8). After calculating (8.8) increment the iteration counter s.

$$\begin{aligned} I^{S+1} &= I^S - \Gamma \\ MS^{S+1} &= MS^S - \Gamma \end{aligned} \tag{8.8}$$

5) Values of $\{P, A, \sigma\}$ constitute now a cluster of temperature – a fuzzy object which can be stored for further calculations. In following part of the text images reconstructed from clusters will be shown.
6) Stop condition: as in classical iterative algorithms, stop condition can be derived from arriving at given number of clusters or by ratio between images I from two consecutive steps (8.9)

$$\frac{I^S}{I^{S-1}} < \delta \tag{8.9}$$

where δ is a user-selected stop parameter. If the given stop condition is not met, the algorithm proceeds to step 2.

9. Results

Following section shows the results of proposed clustering algorithm. Results are presented as a reconstruction of given sample input images from produced clusters. The reconstructed image is obtained by calculating a sum of Γ images acquired from selected number of clusters with equation (8.7).

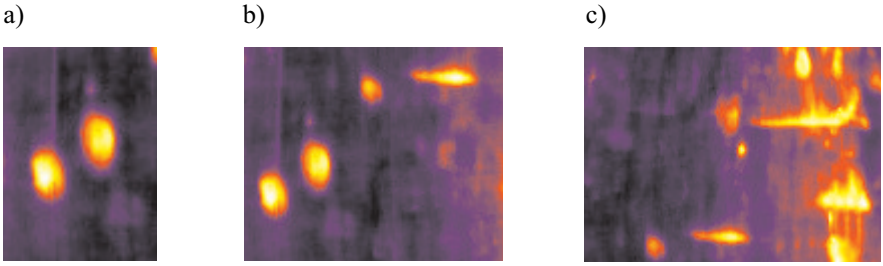

Fig. 11. Sample images: a) sample 1 (82×59); b) sample 2 (96×116); c) sample 3 (108×149)

There were three samples tested, as shown in Figure 11. Resolution of each of them is given in their subscriptions. Results as reconstructions from 25, 50, 75 and 100 clusters (obtained form samples in Fig. 11) are depicted in Figures 12, 13 and 14.

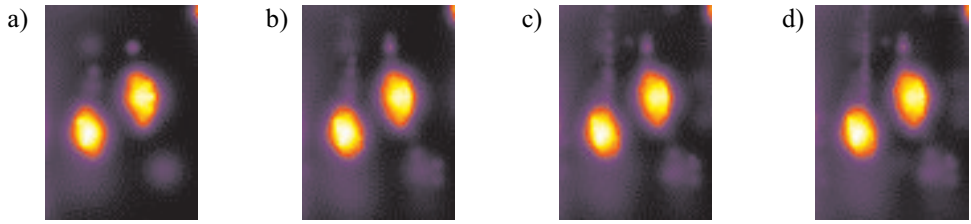

Fig. 12. Sample 1 reconstructed from: a) 25 clusters; b) 50 clusters; c) 75 clusters; d) 100 clusters

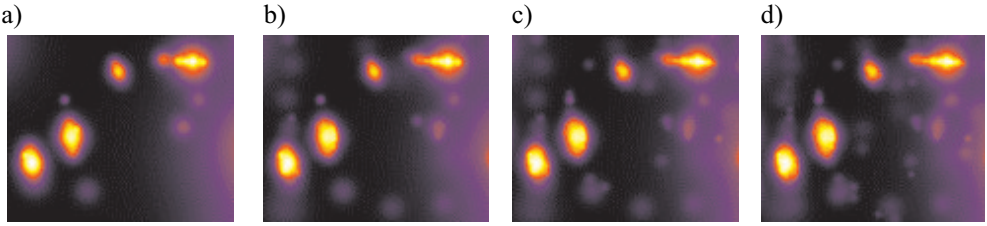

Fig. 13. Sample 2 reconstructed from: a) 25 clusters; b) 50 clusters; c) 75 clusters; d) 100 clusters

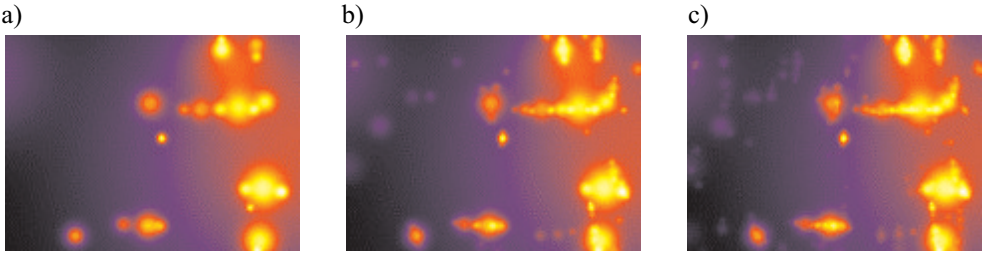

Fig. 14. Sample 3 reconstructed from: a) 25 clusters; b) 50 clusters; c) 75 clusters; d) 175 clusters

Along with three arbitrary selected samples, presented algorithm was also tested on full-size IR image depicted in Figure 15.

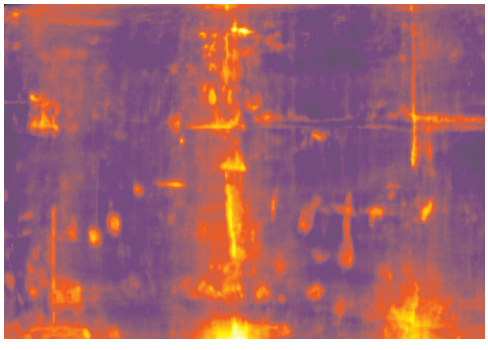

Fig. 15. Full-size image in resolution 332×479 pixels

Results of reconstruction of image from Figure 15 is presented along with a residual image after decomposing into 100, 500 and 1000 clusters (Fig. 16). The residual image is denoted as I^{S+1} in equation (8.8).

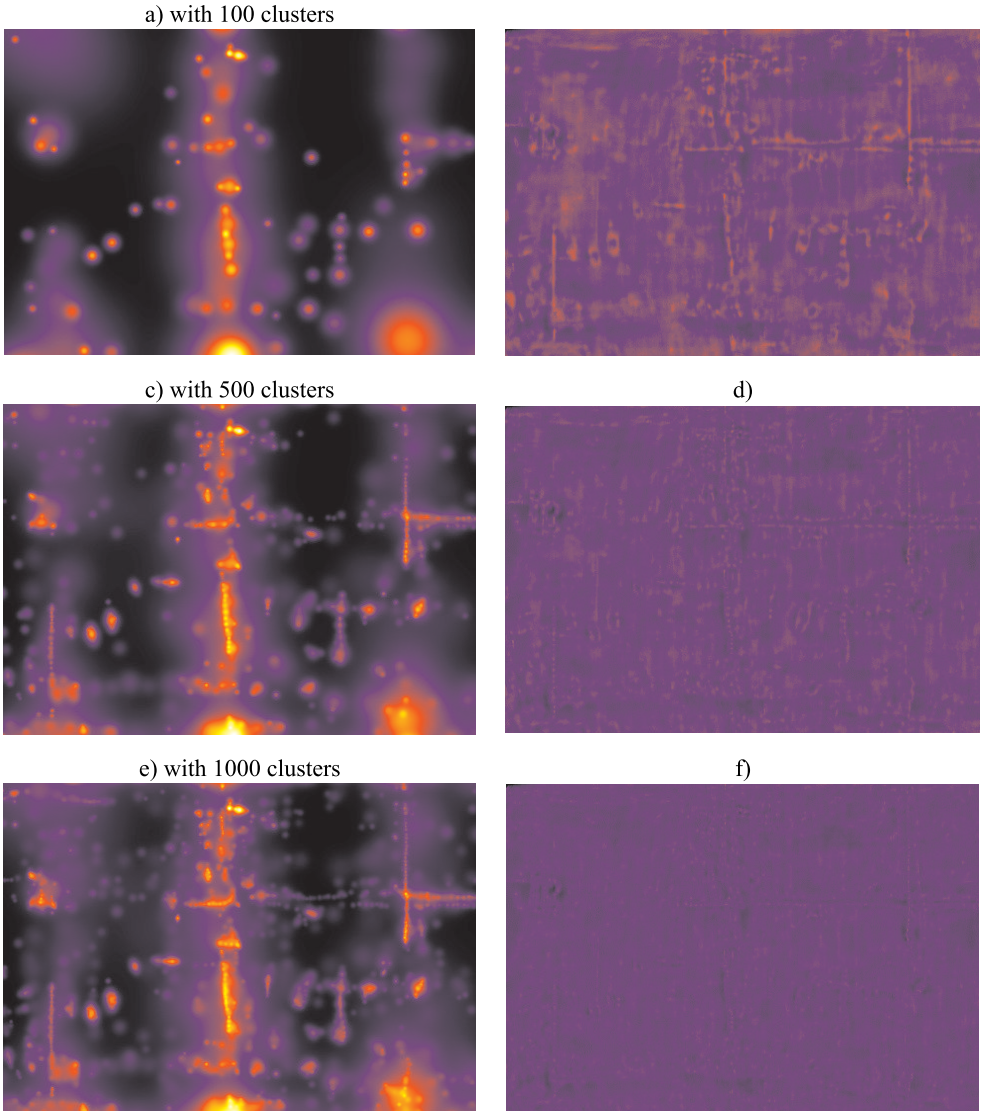

Fig. 16. Reconstructed images and residuals

Following chart, depicted in Figure 17, shows a difference in percents between original image (samples 1–3 and full image) and their reconstructions. The difference assessment

index was calculated from original image I and its reconstruction R_I as in (9.1) where p is the number of a pixel, $I(p)$ and $R_I(p)$ are the values of pixel p:

$$Index(I, R_I) = \sqrt{\frac{\sum_{p=1}^{W \cdot H} \left(I(p) - R_I(p)\right)^2}{W \cdot H}} \cdot 100\% \tag{9.1}$$

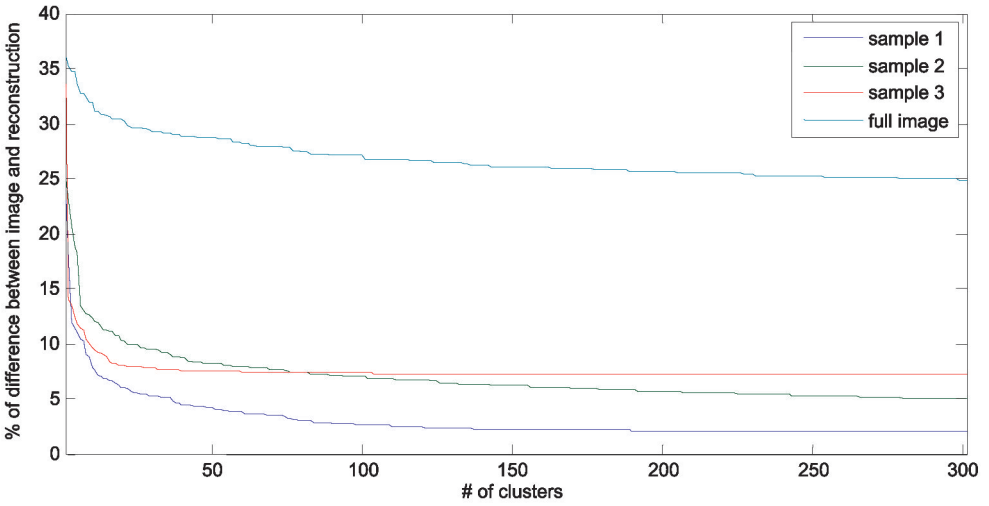

Fig. 17. Percentage distance between image and its reconstruction

With reference to stop condition in described algorithm, it is visible that e.g. for sample 3 clustering algorithm can be finished at level of 100 clusters, for sample 1 at level of 200 clusters, since samples 2 and full image should require more iterations.

10. Conclusions and further research plans

The idea of presented algorithm is to describe a thermal image in terms of unlabeled clusters interpreted as warmer areas. The choice of Gaussian functions is justified by the nature of temperature distribution generated by a heat point source [18]. Moreover, as Gaussian is one of basic membership functions in the fuzzy logic theory this choice corresponds to further research plans aiming at interpreting obtained clusters as fuzzy objects.

Clustering algorithm is divided into two parts: First involves correction methods for three kinds of distortions: a radial pixel distortion caused by optical block of IR camera, perspective distortion and curvature of the measured object – a steel roller. Second part is the clustering algorithm itself, which is an iterative algorithm with simple minimization

routine for acquiring a best fit of cluster into input image. Calculated cluster is subtracted from both image and mountain surface and then algorithm performs next iteration, until the stop condition are met.

Next step of our research will be devoted to combine both: temperature clusters as fuzzy objects and assessing methods for spatial relations between these fuzzy objects into fuzzy rules for inferring about state of the heat distribution on the surface of steel cylinder. Finally, a method based on fuzzy logic for controlling heat dispatching and heaters position will be developed.

References

[1] Linnonmaa P., Todorovic A., Roman N., Donigian D., *Calandering thermal and moisture effects on lwc paper with precipitated calcium carbonate coating applied by metered size press*. TAPPI Metered Size Press Forum, 2002.

[2] Konrad Olejnik K.P., *Rozwój procesu suszenia wstęgi papierniczej.* [in:] X Sympozjum Suszarnictwa, Łódź, 17–19 września 2003.

[3] Slatteke O., *Modeling and Control of the Paper Machine Drying Section*. Department of Automatic Control, Lund University, 2006.

[4] Sander A., Bolf N., Prlic Kardum J., *Research on dynamics and drying time in microwave paper drying*. Chem. Biochem. Eng., vol. 17 (2), 2003, 159–164.

[5] Frączyk A., Urbanek P., Kucharski J., *Algorytm sterowania ruchem wzbudnika i mocą zapewniający równomierny rozkład temperatury nagrzewanego indukcyjnie obracającego się walca stalowego*. Automatyka (półrocznik AGH), vol. 13/3 2009, 1075–1083.

[6] Frączyk A., Urbanek P., Kucharski J., *Correction algorithms in computerized power control system for induction heating*. Automatyka (półrocznik AGH), vol. 14/3, 2010, 423–427.

[7] Urbanek P., *Induction heating of the rotating cylinders of desired temperature distribution along cylinder axis*. PhD thesis, Technical University of Lodz, 1998.

[8] Urbanek P., Kucharski J., *Modelling of dynamic properties of induction heated rotating steel cylinder*. Proc. of the 13th IEEE/IFAC International Conference on Methods and Models in Automation and Robotics, 2007.

[9] Urbanek P., Kucharski J., *Algorytm korekcji zmienności emisyjności powierzchni nagrzewanego indukcyjnie wirującego walca*. [in:] Zeszyty Naukowe Politechniki Świętokrzyskiej, vol. 5, 147–154, Konferencja Modelowanie i Sterowanie Procesów Elektrotermicznych, 2007.

[10] Frączyk A., Kucharski J., Urbanek P., *Komputerowy system stabilizacji mocy grzejnej w układzie nagrzewania indukcyjnego obracającego się walca*. [in:] Zeszyty Naukowe Politechniki Świętokrzyskiej, vol. 15, 39–46, Modelowanie i Sterowanie procesów Elektrotermicznych. Kielce, 22–24 września 2010 r.

[11] Park J., Byun S.-C., Lee B.-U., *Lens distortion correction using ideal image coordinates*. IEEE Transactions on Consumer Electronics, vol. 55, August 2009, 987 –991.

[12] Ahmed M., Farag A., *Nonmetric calibration of camera lens distortion: differential methods and robust estimation*. IEEE Transactions on Image Processing, vol. 14, August 2005, 1215–1230.

[13] Cyganek B., *Komputerowe przetwarzanie obrazów trójwymiarowych*. Exit, 2002.

[14] Jaworski T., Kucharski J., *The use of fuzzy logic for description of spatial relations between objects*. Automatyka (półrocznik AGH), vol. 14/3/1, 2010, 563–580.

[15] Yager R., Filev D., *Approximate clustering via the mountain method*. IEEE Transactions on Systems, Man and Cybernetics, vol. 24, Aug. 1994, 1279–1284.
[16] Shieh H.-L., Chang P.-L., *A novel validity index for fuzzy clustering algorithm*. [in:] 2010 International Symposium on Computer Communication Control and Automation (3CA), vol. 2, May 2010, 326–329.
[17] Chiu S., *Fuzzy model identification based on cluster estimation*. Journal of Intelligent Fuzzy Systems, vol. 2, 1994, 267–278.
[18] Holman J.P., *Heat transfer*. McGraw-Hill, 1986.