Using Thresholding Techniques for Object Detection in Infrared Images

PHAM ICH QUY, MARTIN POLASEK

[*]University of Defence, Kounicova 65, 66210 Brno, the Czech Republic, E-mail: phamichquy.hvktqs@gmail.com

[†]University of Defence, Kounicova 65, 66210 Brno, the Czech Republic, E-mail: martin.polasek@unob.cz

***Abstract*— Image processing techniques play an important role in military applications. Image binarization could be understood as a process of pixel values segmentation of grayscale image into two value groups, zero as a background and 1 as a foreground. In simple humorapplication of object detection we assume that contrast distribution of foreground is uniformed and without background noise or that variation in contrast does not exist. However, in complex cases previous conditions are inappropriate as variation in contrast exists and it does include background noise, etc. This paper deals with object detection in infrared images for military application using an image binarization step. Military targets are detected in different conditions such as winter condition, summer condition, at night etc. This paper focuses on combination of two methods of image binarization. One is the global binarization method proposed by Otsu and the other one is the local adaptive threshold technique. The global binarization method is usually faster than the local adaptive method and the global method will give good results for specific weather conditions such as object detection in winter condition. In these cases, acquired images have uniform contrast distribution of foreground and background and little variation in illumination. We are looking for an effective method for object detection in infrared images in challenging conditions such as summer conditions or in an urban environment, where there is a shortage of objects of interest. In these cases, we employed local mean techniques and local variance techniques. The experiment results are presented so that we can better choose which method should be employed or what combination of these previous techniques to employ. In order to minimise computational time of local thresholding technique, we employed a combination of two previous techniques. The algorithm was tested in a Matlab environment and the tested pictures were acquired by RayCam C.A. 1884 and thermoIMAGER 160 cameras.**

Keywords- Local thresholding technique, Global thresholding technique, Object detection, Digital image processing, Infrared image, Binarization techniques, Matlab, Integral sum image,

I. INTRODUCTION

The techniques for image format conversion from grayscale to binary could be grouped into two categories: global group and local group. Global methods group – i.e. the one proposed by Otsu [1] finds a single or general threshold value for the whole image. On basis of this general threshold value the algorithm will assign each pixel in the grayscale image to foreground or background in the binary image, see [1]–[6]. These statistical methods are suitable for converting any grayscale image into binary form, but they are inappropriate for complex images, for example infrared images obtained in summer conditions and in an urban environment. To overcome these complexities, local thresholding techniques could be proposed for binarization techniques. These techniques calculate a different threshold for each pixel according to the grayscale information of the neighbouring pixels, see [7]-[14]. The hybrid techniques see, [15], [16], which combine information on global and local thresholds belong to another category.

In this paper, we focus on the binarization of a grayscale image using both thresholding techniques. Local binarization methods try to compute thresholds individually for each pixel, using information from the local neighborhood of the pixel. These methods are often slow since the computation of image features from the local neighborhood is to be done for each image pixel. This paper focuses on employing a fast approach to compute local thresholds using the technique of integral sum image [17]. Using this approach we are able to achieve binarization speed close to the global binarization methods [17]. What more, the Otsu method was used representing the other techniques in the global category. Global techniques are very fast and they give good results for typical images.

II. ALGORITHMS AND RESULTS

A. *Global threshold method*

Otsu method was used for calculating the threshold. For still images, we calculated a threshold value using the above method for each image. The images obtained in winter and spring conditions were arranged into two different groups (winter and spring group). For each group of i images we had the i threshold values of different groups. On the basis of the threshold values in each group, the mean threshold value and standard deviation was calculated. The mean threshold value is then a standard threshold value for converting grayscale image to binary image in that group. For video sequences, the Otsu method was employed to calculate the threshold value corresponding to each frame in initial n frames. On the basis of n obtained threshold values, we could determine the mean and standard deviation threshold value. The mean threshold value was then used as the standard threshold value for converting the remaining frames in binary.

The green line in Fig. 1, shows the change of threshold values for images obtained in winter conditions and their mean threshold values marked with purple color $m_1 = 0.2137$. The light green graph represents the change of the threshold value depending on the different images that were recorded in spring condition. The red line shows their average value $m_2 = 0.3829$. $\sigma_1 = 0.0874$, $\sigma_2 = 0.1320$

Fig. 2 shows the change of threshold values of the 3 different video frames. Each video is represented in a different colour.

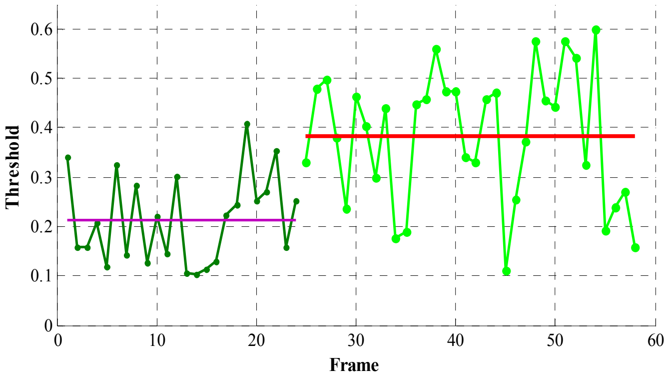

Figure 1: Threshold values according to different images

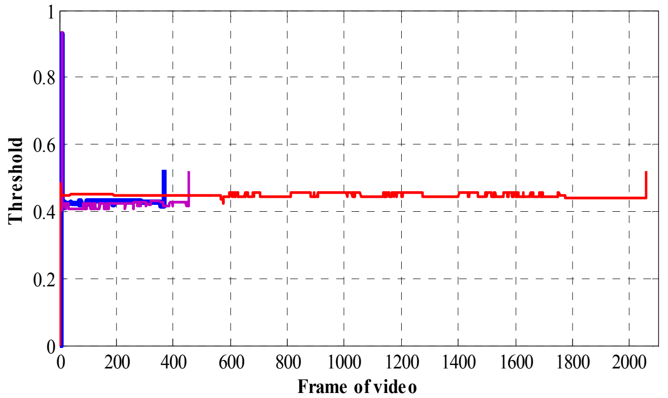

Figure 2: Threshold of frames in the video

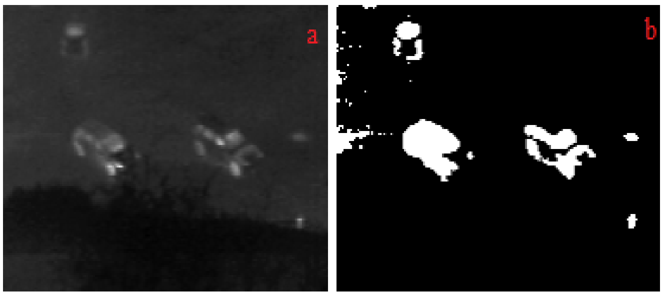

Figure 3: Detection result using Otsu method in image 1

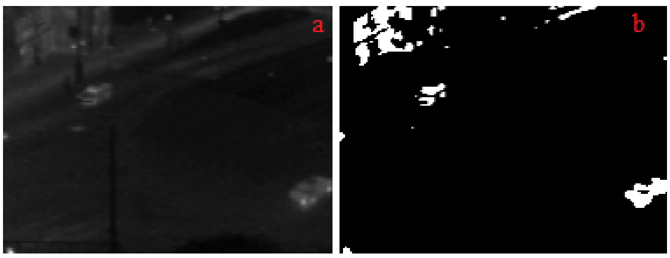

Figure 4: Detection result using Otsu method in image 2

Fig. 3, Fig. 4 and Fig. 5 show the objects that were detected by using the Otsu method. In each figure, image (a) is a grayscale image that was converted into binary image (b). The images were recorded in winter conditions. It is not difficult to detect objects of interest in each image.

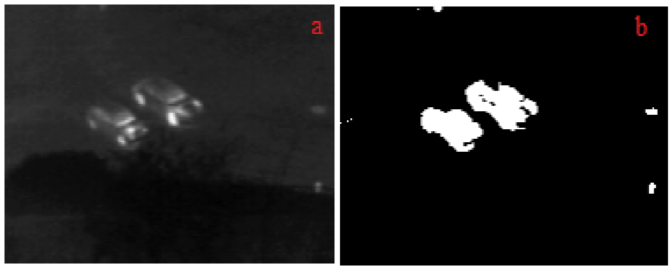

Figure 5: Detection result using Otsu method in image 3

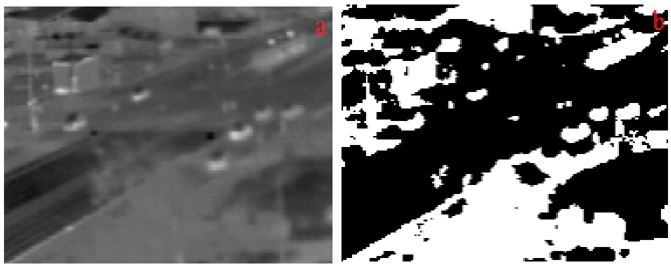

Figure 6: Detection result using Otsu method in image 4

Figure 7: Detection result using Otsu method in image 5

Fig. 6 and Fig. 7 show the results of Otsu method for converting a grayscale image into a binary image. These images were collected in spring conditions.

B. *Threshold adjustment*

Threshold adjustment is based on assuming that, after using global thresholding, we obtained some object that was too large in comparison to the size of the objects of interest. As an example in Fig. 7, we could see that using the above mean global threshold value did not seem to be effective. A larger global threshold value was needed. Global threshold value increased gradually to determine values that were not more than the sum of the mean threshold value and the standard deviation.

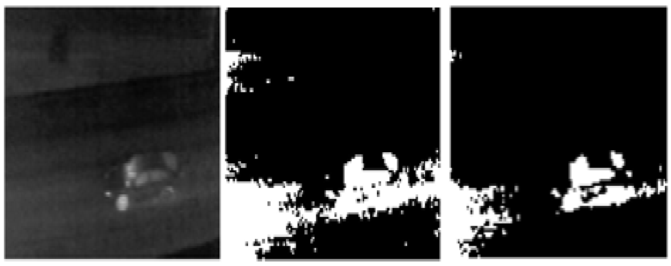

Figure 8: a) grayscale image, b) using mean threshold value, c) using threshold adjustment

C. Local adaptive threshold method

The application of adaptive thresholding techniques to local algorithms is based on the fact that for a complex environment, where the threshold value was calculated by global methods, the correct value for the entire image could not be guaranteed. However, using this value, can find the object that would be removed or hidden in the foreground. Local threshold techniques can be complicated and the total number of computational steps can be large, thus consuming more processing time. An improvement of the author for the local threshold value calculated from integral image is depicted in [17]. Afterwords, the other author added a bit nominated for calculating the integral image in order to reduce the computation time for the next step. After having integral image $g(x,y)$ the following formula can be used to calculate the local value [17].

$$T(x,y) = m(x,y)\left[1 + k\left(\frac{\partial(x,y)}{1-\partial(x,y)} - 1\right)\right] \quad (1)$$

Where $\partial(x,y) = I(x,y) - m(x,y)$ is the local mean deviation, and k is a bias. Its range is [0,1] only. $m(x,y)$ is a local mean.

The arithmetic mean $m(x,y)$ within the window of size $w\times w$ of the image I is.

$$m(x,y) = \frac{s(x,y)}{w^2} \quad (2)$$

Where $s(x,y)$ is the local sum of any window size w, and can be computed simply by using addition and subtraction operations as shown in the following equation.

$$s(x,y) = [g(x+d-1, y+d-1) + g(x-d, y-d)] - [g(x-d, y+d-1) + g(x+d-1, y-d)] \quad (3)$$

Where $d = round(w/2)$ (4)

Examples of using this technique are showed in Fig. 9 and Fig. 10. The grayscale image in Fig.9 was recorded in winter conditions, and in Fig. 10 in the spring conditions. Both for parameters k = - 0.0001, w = 7.

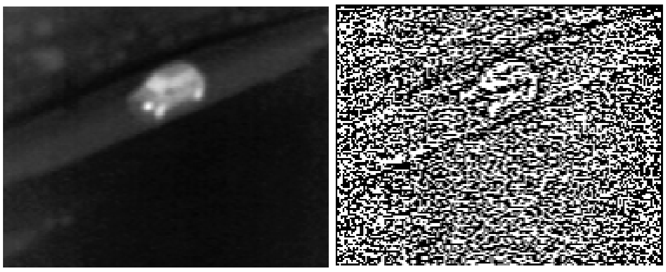

Figure 9: Grayscale image 6 (left), binary image using local thresholding (right)

As the results have shown, the object and its background are still discriminated. However, if we employ this technique to all the pixels in the images, we can get the following two problems. One is a long computational time, although this technique was improved by using the sum of integral image. The second problem is that it is difficult to separate the objects of interest from their neighbouring objects in the binary converted images. The objects of interest may still belong to other objects, if we use the wrong parameters w and k.

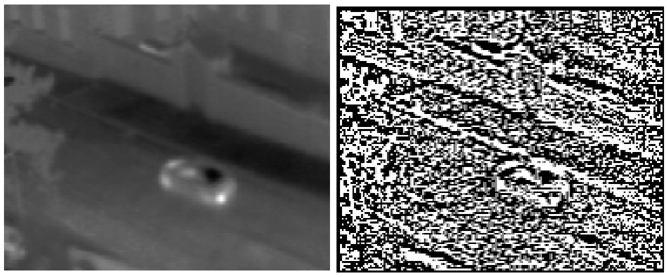

Figure 10: Grayscale image 7 (left), binary image using local thresholding (right).

D. Combined techniques

In order to employ techniques combination, we need to rely on the statistics of the objects properties in the binary image after using global thresholding. Often, in certain conditions, such as winter conditions, using global threshold technique allows exact detection of objects of interest (see again Fig.3, 4, 5). The obtained objects had most of their properties similar to the real object properties in the grayscale image, for example size and shape. In these cases, the local threshold technique didn't have to be employed, because it requires a longer computational time, it is difficult to separate objects of interest from their background and it is difficult to keep the integrity of the object's size. In the spring conditions, when the contrast between the objects of interest and their background was not as sharp, it was difficult to detect these objects (see Fig.6 and 7) when using this global threshold method. So our idea is to then switch to the local adaptive threshold method. Suppose that, after using global technique, we did not detect the desired objects and the obtained objects were large (see Fig. 7). However, these large objects could include some object of interest.

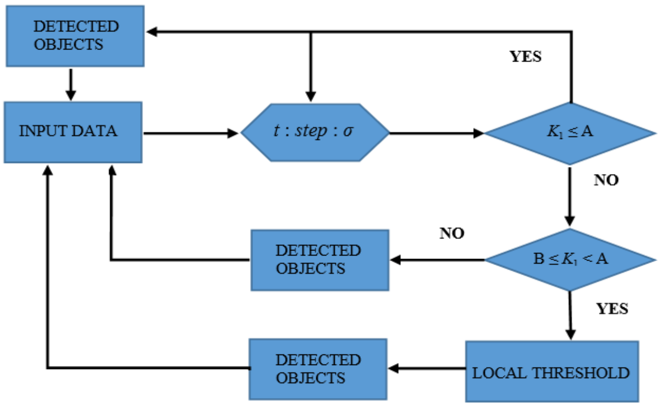

Figure 11: Combined techniques flowchart

Fig. 11 illustrates the combined method algorithm. Each method was employed based on the criteria of the number of pixels of obtained objects and the input image. The algorithm is simply described as follows: The input color (grayscale) image is converted to a binary image by using a mean global threshold value that was calculated by statistical method it means, this value was calculated earlier. After the conversion process, the process continues to determine the number of pixels of the obtained object, and then it compares the standard K_1 (the number of pixels of the largest objects) with A (the number of pixels of the input image). If $K_1 \geq A$ the process will continue by using threshold adjustment algorithm. If $K_1 < A$ the algorithm will separate the obtained objects into three groups. The first group contains objects that have the number of pixels in the interval (S_1, B). These objects of the group are thought to be the detected objects. The second group includes objects with the number of pixels smaller than S_1. These objects are considered as noise. For the objects of third group, with their number of pixels in the interval (B, A), the local thresholding algorithm will be employed. And then a threshold adjustment will be employed, in order to reduce computational time - as the example in Fig. 7. The reason is that in cases when the size of the obtained object is almost as long as the size of the input image, applying local thresholding in the step would not reduce the computational time.

In order to simulate the researched results, we divided the detection algorithm into two groups. The first group was focused on object detection in still images and the second group focused on video sequences.

1) Object detection in images: In the first example, we assumed that the size of the object of interest was in the range of (S1, B) = (350, 4000), $A = 14114$. The result is shown in Fig. 12, where the winter grayscale image (a) was converted to a binary image using the mean global threshold value m_1 = 0.2137. All obtained objects are shown in Fig.12b. In this case, the objects with the number of pixels greater than A were not found. Therefore, the threshold adjustment method was omitted.

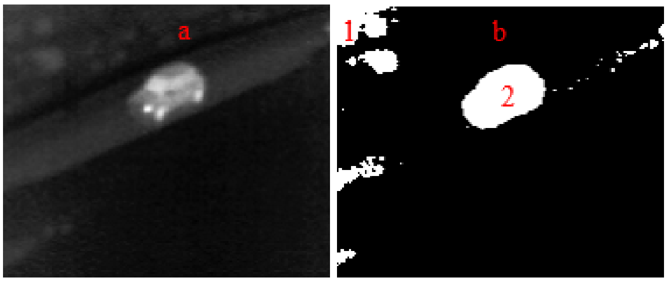

Figure 12: A grayscale image (in winter) and its binary image using global threshold method

The selection of parameters as S_1, B and A first affected the obtained number of pixels, the shape of the objects. In this selection, the third group included the object 1, and object 2 (see Fig. 12b). The objects have number of pixels in the interval (B, A) = (4000, 14114), where the local threshold process was employed. The results of using the local threshold method for objects 1 and 2 is shown in the left part of Fig. 13 (see Fig. 13). The other obtained objects were reviewed to check their sizes within the selected criteria, i.e. which group the objects belonged to. The objects belonged to the first group and to the second group. The objects in the first group are shown in the right part of Fig. 13. The final result of using the combined technique is shown in the Fig.14. The problem was that the initial number of pixels of the first objects received was 4781 pixels, the second object was 11646. After using the local adaptive technique the number of pixels of the first object was 1912 pixels and for the second object it was 4232 pixels.

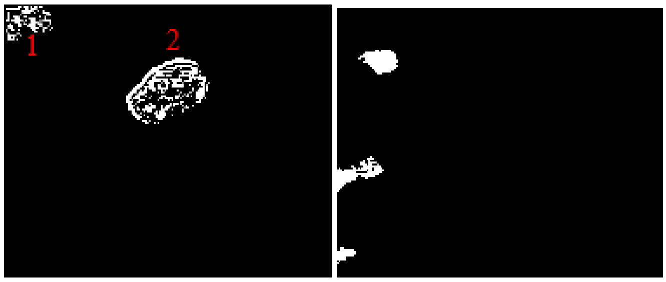

Figure 13: Obtained objects using local threshold (left), Objects of the first group (right)

Figure 14: The final result using the combined method

If we take the advantage of optimizing with the global thresholding algorithm and we want to get the best results by combining it with the local adaptive thresholding algorithm, we need to carefully choose the correct parameters S_1, B, A. In order to obtain a better shape and the correct number of pixels of the objects of interest, we only changed the parameter B, then the algorithm produced expected results as shown in Fig. 14.

In the second example, the input image was recorded in spring conditions (see Fig. 15). The grayscale image (left) was converted to binary by using mean global threshold value m_2 = 0.3829. All obtained objects are in Fig.15b. In this case, we used the same values of parameters S_1, B as above. However, the value of parameter A depended on the image size. The value A was calculated as follows.

$$A = X \cdot Y / 20 = 495 \cdot 373 / 20 = 9231$$

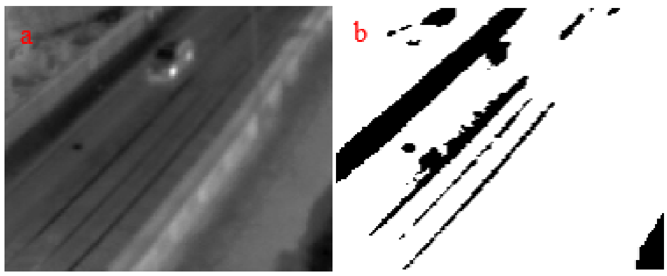

Figure 15: A grayscale image (in the spring) and its binary image using a global threshold method

Using global thesholding, we only obtained one object with the number of pixel more than A. The algorithm continued by using the threshold adjustment method. The obtained binary image is shown in Fig. 16a. The threshold adjustment method finished, we obtained 4 objects. However, the objects 1 and 4 had the number of pixels more than A, because the algorithm used the maximum value of threshold. In this case, the local threshold method was not employed (see Fig. 16b). The first group includes objects 2 and 3. The third object had 3386 pixels, and the second object had 2272 pixels. The final result of the combined method is shown in Fig. 17.

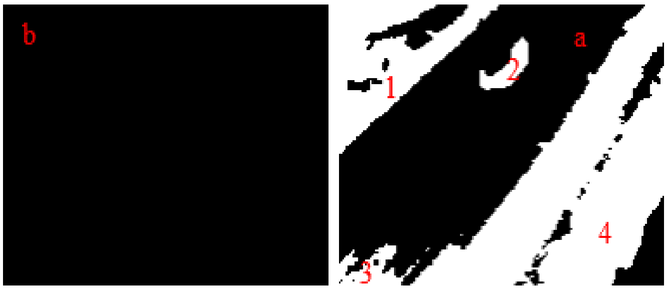

Figure 16: Result of using threshold adjustment (a), using threshold adjustment (b)

Fig. 16b shows the results obtained by using local adaptive thresholding techniques. This technique shows that the object does not exist in the group of oversized objects (groups 1 and 4). As a result, we obtained only two remaining objects as shown in Fig. 17.

Figure 17: The result of combined method

Now, we will return to the first example (winter image in Fig. 12). In order to detect the objects of interest, with their maximum realistic size and shape as close as possible to the real object, without using a local thresholding algorithm (when it is not necessary to apply the local thresholding algorithm), we needed to change the range of values in the interval (S_1, B) = (350, 12000). Therefore, all the obtained objects belonged to the first group. It wasn't necessary to employ the threshold adjustment nor the local thresholding algorithm. Fig. 18 shows the results of the combined method. In this case, the objects of interest had the biggest size.

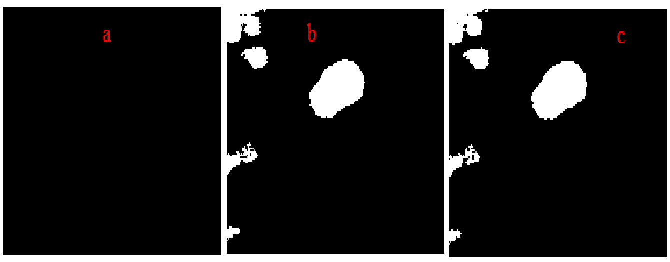

Figure 18: No objects using local thresholding (a), detected objects using global threshoding (b), detected objects using combined method(c)

For the second image obtained in the spring (see Fig. 15), the image size changed, so we kept the value A = 9232 and the detected objects size were in the range (S_1, B) = (350, 8000), from which we obtained the results shown in Fig. 19.

Figure 19: No objects using local thresholding (a), detected objects using global thresholding (b), detected objects using combined method(c)

Then, we tested the combined algorithm in the input image that contained many objects of interest. In the first case, the parameters were used as follows, (S_1, B) = (350, 4000) A = 14114. The result is shown in figure 20.

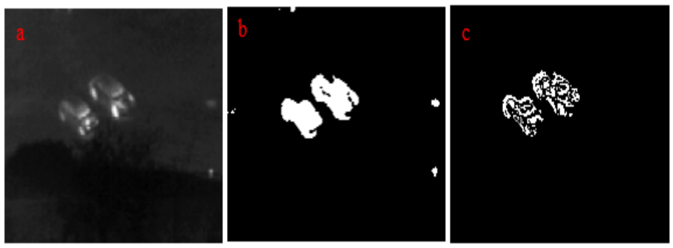

Figure 20: Grayscale image (a), detected objects using global threshoding (b), detected objects using combined method (c)

In the second case, the size of detected object was (S_1, B) = (350, 8000). We obtained results shown in Fig. 21.

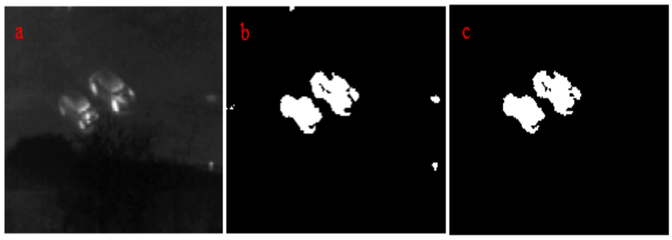

Figure 21: Grayscale image (a), detected objects using global thresholding (b), detected objects using combined method (c)

2) Object detection in video sequences: For the video sequence we employed the combined algorithm which operates the same as the combined algorithm used in still images. The difference was that we found the mean threshold value and standard deviation based on 1/3 of the initial number of frames of each video sequence. The calculated mean threshold was considered as a global threshold value for converting next grayscale frames of the video sequence to binary. The combined algorithm was implemented in next frames after obtaining a mean threshold value. The influence of both the local threshold method and threshold adjustment, to object detection, is described below.

We used the proposed parameters as follows, window size w = 3, bias coeficient k = -0.05, detected objects had the number of pixel in the interval of (S_1, B) = (40, 500), A = frame size/10. The number of steps employed by the threshold adjustment was 20.

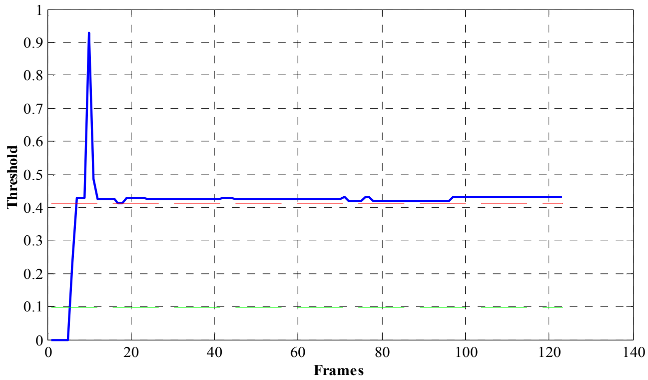

Figure 22: Threshold values depending on frame No. 1

Fig. 22 shows the change of threshold value depending on different frames. The threshold value is almost stable value from 20th frame, which is about 0.4.

In order to illustrate the results obtained in the processing of the video sequence, we present three results in 3 different frames of the video sequence, containing one object.

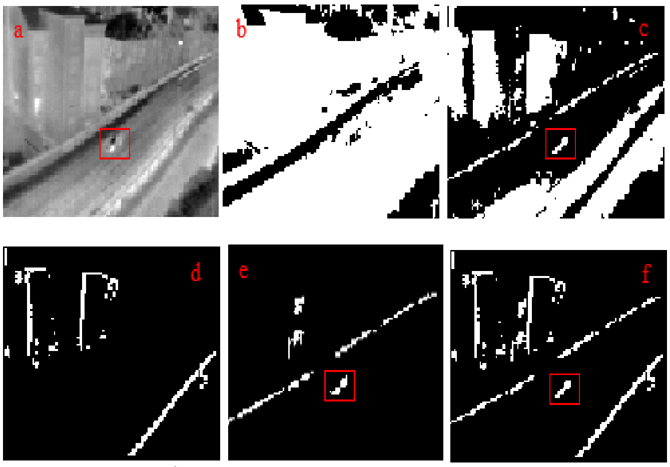

Figure 23: The 126th grayscale frame (a), binary frame using global threshold (b), binary frame using threshold adjustment (c), binary frame using local threshold (d), binary frame containing objects in given interval (S_1, B) (e), final result using combined method (f).

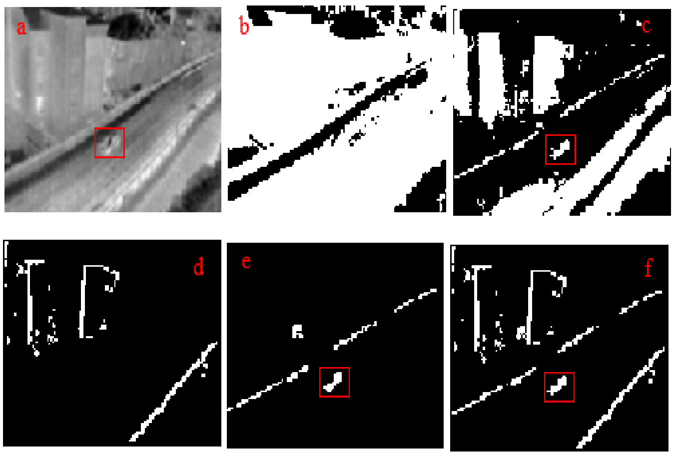

Figure 24: The 142nd grayscale frame (a), binary frame using global threshold (b), binary frame using threshold adjustment (c), binary frame using local threshold (d), binary frame containing objects in given interval (S_1, B) (e), final result using combined method (f).

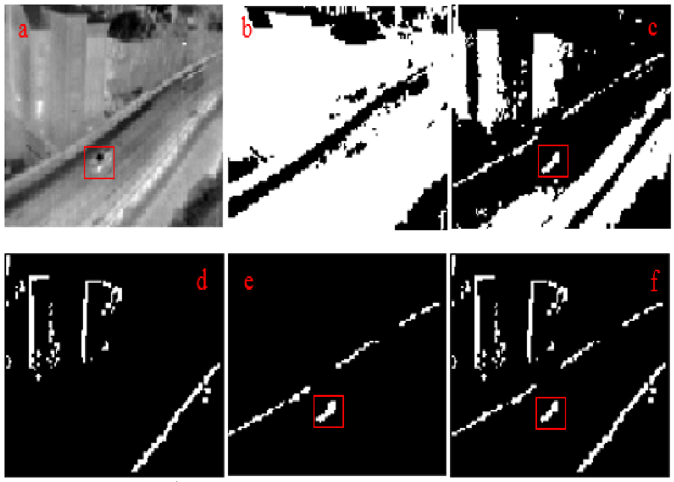

Figure 25: The 166th grayscale frame (a), binary frame using global threshold (b), binary frame using threshold adjustment (c), binary frame using local threshold (d), binary frame containing objects in given interval (S_1, B) (e), final result using combined method (f).

In all three mentioned examples (see Fig. 23b, Fig. 24b, Fig. 25b) we could see that all three binary frames obtained by the global threshold technique did not produce good results, i.e. the target location was not detected. The object of interest was detected by the threshold adjustment method and some other criteria such as the number of pixels. The process of detection could be stopped at this point, however, other objects could be hidden in other large objects (see Fig. 23c, 24c, 25c). The objects of the third group were processed by local thresholding and the obtained result is shown in Fig. 23d, 24d, 25d. The first group included the objects shown in Fig. 23e, 24e, 25e. The other objects were considered as noise. Fig. 23f, 24f, 25f in this case showed the final result using a combined technique. From this final result we easily filtered the object of interest by using other criteria such as length–width ratio standard and the number of pixels of the selected object.

Now, we wanted to keep these parameters, we only changed the value of the window size $w = 7$. When we changed only the parameter w, the results changed only in pictures *d* in Fig. 23, 24, 25. The following Fig. 26 shows the changes of pictures *d* in Fig. 23, 24 and 25.

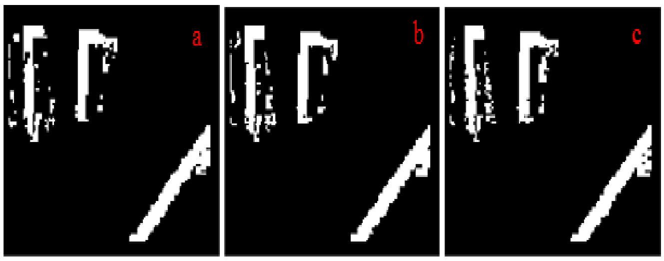

Figure 26: Changing of result in the frame 126th (a), the frame 142nd (b), and the frame 166th (c)

In both cases, the more changes of window size w increased, the larger the number of computational steps. The obtained objects in the different frames were almost identical. Their location would not change, if the camera was not moved. In both cases, the detected object could be easily removed by using the number of pixels criteria and their length - width ratio. In both cases, the object of interest was not detected by local thresholding.

The second testing of object detection in video sequence contained many objects. The mean threshold value was calculated the same way as in the above example. The graph of the threshhold values is represented in Fig. 27.

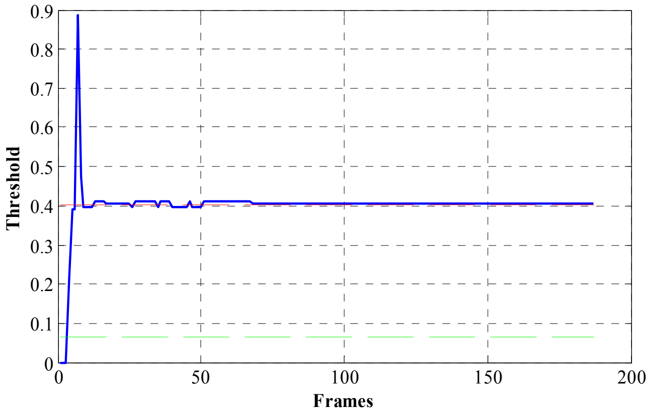

Figure 27: Threshold values depending on frame No. 2

In this case, we needed to change the number of steps employed in threshold adjustment. The results are shown in the following Fig. 28 and Fig. 29.

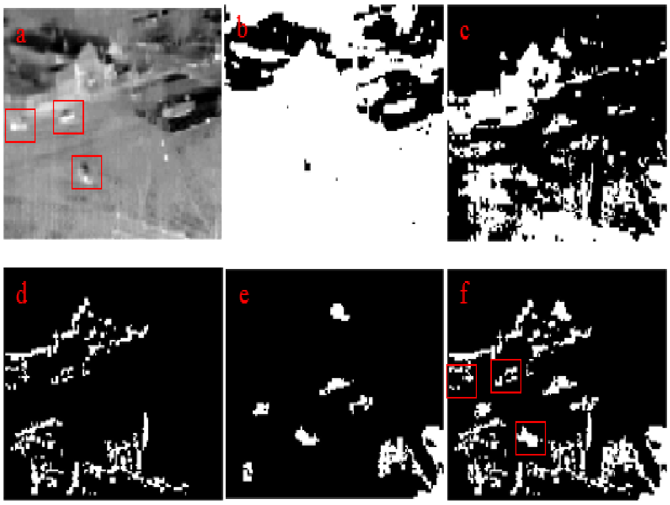

Figure 28: The 190th grayscale frame (a), binary frame using global threshold (b), binary frame using threshold adjustment (c), binary frame using local threshold (d), binary frame containing objects in given interval (S_1, B) (e), final result using combined method (f).

The change of the number of steps can be adjusted based on the criteria of the obtained object size. Basically, for two tested video sequences, the same values of parameters were kept, except the number of steps in the threshold adjustment method. Number of steps was changed to 35 steps. In this case, the objects of interest were detected by a local adaptive thresholding method.

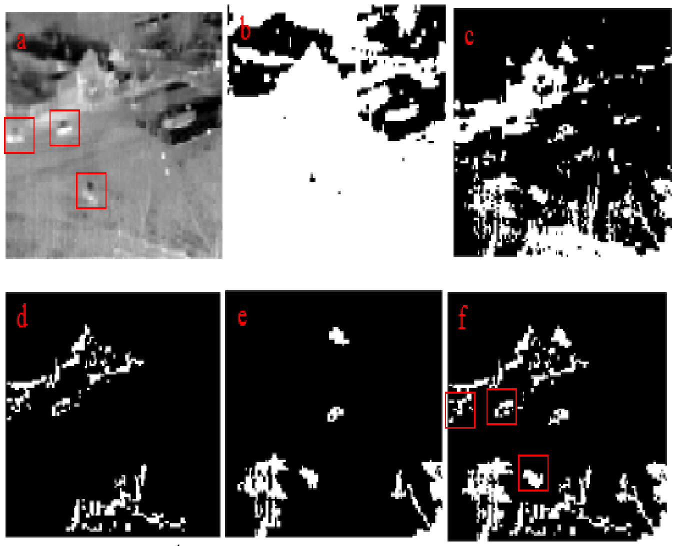

Figure 29: The 210th grayscale frame (a), binary frame using global threshold (b), binary frame using threshold adjustment (c), binary frame using local threshold (d), binary frame containing objects in given interval (S_1, B) (e), final result using combined method (f).

III. CONCLUSION

In this paper, we have shown the effectiveness of the combined thresholding method to obtain binary image with optimal results. The use of a global thresholding algorithm gives us relatively efficient results in specific conditions, for example, in winter conditions, night time conditions, where the contrast between the objects of interest and their background is enough to remove objects from their background. Global thresholding proved less effectivein conditions with low contrast, such as the conditions of sunlight, etc. In these cases, we employed a local threshold, the local adaptive thresholding algorithm was employed only for objects based on the results of global thresholding. The goal of the combined thresholding method is to reduce the computational time of using local adaptive thresholding. In order to avoid the use of the local adaptive thresholding algorithm for very large objects, the obtained objects were divided into four groups. The first group contained objects that had the number of pixel in the interval of (S_1, B). These objects of the group were thought to be detected objects. The second group included objects with the number of pixels smaller than S_1. These objects were considered as noise. For objects in the third group, i.e. with their number of pixels in the interval of (B, A), the local thresholding algorithm was employed. The fourth group contained the objects with a size bigger than A. These object were oversized. If the obtained objects, after using the global thresholding, included some object belonging to the fourth group, we automatically employed the threshold adjustment algorithm before the local threshold algorithm was used. The intermediate algorithm was to minimize the use of local adaptive threshold in cases of obtaining large objects. Also, the computational time was reduced by using the adjustment algorithm.

In the case of object detection in video sequences, parameters such as the window size, the bias coefficient and the number of steps used in the threshold adjustment method were changed. The detected objects could be fully selected on their number of pixels and length - width ratio. The use of a local adaptive thresholding method might help us to detect more objects, and also remove the objects from the groups of buildings, roads, etc.

ACKNOWLEDGMENT

The work presented in this paper has been supported by the Ministry of Defence and Ministry of Education, Youth and Sports of the Czech Republic (K206 Student research program "Implementation of modern technologies in avionics systems").

REFERENCES

[1] N. Otsu, "A threshold selection method from gray-level histograms," IEEE Trans. Systems, Man, and Cybernetics 9(1), pp. 62–66, 1979.

[2] P. Viola and M. J. Jones, "Robust real-time face detection," Int. Journal of Computer Vision 57(2), pp. 137– 154, 2004.

[3] R. Cattoni, T. Coianiz, S. Messelodi, and C. M.Modena, "Geometric layout analysis techniques for document image understanding: a review," tech. rep., IRST, Trento, Italy, 1998.

[4] F. Shafait, D. Keysers, and T. M. Breuel, "Performance comparison of six algorithms for page segmentation," in 7th IAPR Workshop on Document Analysis Systems, pp. 368–379, (Nelson, New Zealand), Feb. 2006.

[5] J. M. White and G. D. Rohrer, "Image thresholding for optical character recognition and other applications requiring character image extraction," IBM Journal of Research and Development 27, pp. 400–411, July 1983.

[6] Chi, Z., Yan, H., and Pham, T.: 'Fuzzy algorithms: with applications to image processing and pattern recognition' (World Scientific Publishing, 1996)

[7] Bernsen, J.: 'Dynamic thresholding of gray-level images'. Proc. 8th Int. Conf. on Pattern Recognition, Paris, 1986, pp. 1251–1255

[8] Chow, C.K., and Kaneko, T.: 'Automatic detection of the left ventricle from cineangiograms', Comput. Biomed. Res., 1972, 5, pp. 388–410

[9] Eikvil, L., Taxt, T., and Moen, K.: 'A fast adaptive method for binarization of document images'. Proc. ICDAR, France, 1991, pp. 435–443

[10] Mardia, K.V., and Hainsworth, T.J.: 'A spatial thresholding method for image segmentation', IEEE Trans. Pattern Anal. Mach. Intell., 1988, 10, (8), pp. 919–927

[11] Niblack, W.: 'An introduction to digital image processing' (Prentice-Hall, Englewood Cliffs, NJ, 1986), pp. 115–116

[12] Taxt, T., Flynn, P.J., and Jain, A.K.: 'Segmentation of document images', IEEE Trans. Pattern Anal. Mach. Intell., 1989, 11, (12), pp. 1322–1329

[13] Yanowitz, S.D., and Bruckstein, A.M.: 'A new method for image segmentation', Comput. Vis. Graph. Image Process., 1989, 46, (1), pp. 82–95

[14] Sauvola, J., Seppanen, T., Haapakoski, S., and Pietikainen, M.: 'Adaptive document binarization'. Proc. 4th Int. Conf. on Document Analysis and Recognition, Ulm Germany, 1997, pp. 147–152

[15] Gorman, L.O.: 'Binarization and multithresholding of document images using connectivity', CVGIP, Graph. Models Image Process., 1994, 56, (6), pp. 494–506

[16] Liu, Y., and Srihari, S.N.: 'Document image binarization based on texture features', IEEE Pattern Anal. Mach. Intell., 1997, 19, (5), pp. 540–544 [24] W. Niblack, An Introduction to Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1986.

[17] SINGH, ROY, SINGH, SINAM a Kh.Manglem SINGH. A New Local Adaptive Thresholding Technique in Binarization. In: [online]. 2011 [cit. 2014-09-25]. ISSN 1694-0814. Aviable from: http://arxiv.org/ftp/arxiv/papers/1201/1201.5227.pdf