

Evaluation of hardware implementation of the infrared image enhancement algorithm

by R. Dulski*, T. Sosnowski*, T. Piątkowski* and S. Milewski**

* Institute of Optoelectronics, Military University of Technology
 ul. gen. Sylwestra Kaliskiego 2, 00-908 Warsaw, Poland, rdulski@wat.edu.pl
 ** Navigation and Naval Weapons Division, Polish Naval Academy,
 ul. Śmidowicza 69, 81-103 Gdynia, Poland, s.milewski@amw.gdynia.pl

Abstract

Infrared images are a specific source of information. Perception and interpretation of the infrared image greatly depends on thermal properties of observed object and surrounding scenery. The most effective solution for enhancing infrared images are digital image processing mechanisms implemented in the camera or applied after image registration. The paper evaluates effectiveness of a hardware implementation of the infrared image enhancing algorithm in a digital data processing system. The algorithm is based on plateau histogram equalization and was finally implemented in FPGA. The algorithm allows real time improvement of the image quality and thus increases the effectiveness of object detection.

1. Introduction

IR image interpretation is greatly influenced by the properties of observed object and surrounding scenery. Obviously not only objective factors play important role in perceiving the image but also subjective, human-specific perception of visual information. Thus, even optimal settings of the camera (focus, temperature range, camera placement) do not guarantee successful detection [2, 3, 10]. The solution to above mentioned problem could be algorithms based on the methods of digital image processing. They can be implemented in the camera or applied after image registration. The algorithms should provide intelligent, dynamic brightness control applied not only across entire image but also selectively to specific areas in order to maintain optimal visualization of observed scenery. Auto-adaptation to the changing conditions will ensure optimal, from observer's point of view, output image.

Thermal image is generally a visualization of IR radiation, emitted by the observed object and its surroundings (background). The output image from a thermal camera differs significantly from visual image (Fig. 1). Consequently, it is difficult to obtain proper visualization of a thermal scene in a form easy to comprehend by a human observer.



Fig. 1. Comparison of thermal (left) and visual (right) images.

Process of image enhancement does not increase the amount of information in the image. All of the information is already there, but by enhancing the image its certain features are modified thus making image easier to interpret visually and some details become discernible. The theoretical background for thermal image enhancement is digital image processing theory. That does not mean, however, that procedures that work well for visual images can be directly adapted to thermal image processing. Differences between visual and thermal images make such direct approach usually ineffective. It is necessary then to carefully choose the applied methods and modify them taking into account specific properties of thermal images [1, 2, 3].

Thermal image is created by converting the thermal radiation that reaches IR detector into visual range perceived by a human eye. The level of incident IR radiation at the detector depends on many factors and emissivity is one of most important [1, 2, 3]. It is a property of a real object characterizing its ability to radiate energy. The object's emissivity ε in full spectrum can be defined as a ratio of exitance (emissive power) of object's surface $M_{obj}(T)$ and surface of a blackbody $M_{bb}(T)$, both at the same temperature T :

$$\varepsilon = \frac{M_{obj}(T)}{M_{bb}(T)}. \quad (1)$$

Emissivity is a function of chemical properties of an object, parameters of its surface and depends on the temperature, wavelength λ , angle of observation and sometimes on time, in case of rapid temperature changes. For opaque objects, according to Kirchhoff's law the following relation can be written:

$$\varepsilon_\lambda + r_\lambda = 1. \quad (2)$$

This expression describes the relation between emissivity ε_λ and reflectance r_λ . High emissivity means that given object reflects very little of ambient radiation. This effect is extremely important from image creation point of view and directly determines how the object can be visualised (observed) in IR, where luminosity of each pixel depends strongly on both self-emission and reflected radiation coming from neighbour heat sources. The exitance of the object $M_{obj}(T)$ in full spectrum can be defined using Stefan-Boltzmann Law:

$$M_{obj}(T) = \varepsilon \int M_{obj,\lambda}(T) d\lambda = \varepsilon \sigma T^4 \quad (3)$$

The value of object's exitance $M_{obj}(T)$ is equal (with accuracy to a constant) to object's radiant power W_{obj} at the camera detector. As it was mentioned above additional sources of IR radiation have to be considered while analysing total radiant power W_{tot} on camera's detector. The radiant power of a background W_{refl} can very diverse (clouded sky, sea surface, forest, etc.). The radiant power of an atmosphere W_{atm} itself also contributes to the total signal recorded by thermal camera and attenuates signal from observed object according to its transmission properties τ [1, 3]. In consequence the object's radiant power W_{obj} and the same the object's image creation process should be described as:

$$W_{obj} = \frac{W_{tot} - (1 - \varepsilon)\tau W_{refl} - (1 - \tau)W_{atm}}{\varepsilon\tau} \quad (4)$$

Another property that also impairs the quality of thermal image is emissivity dependence on the angle of observation. It is described by Lambert's cosine law, saying that the radiant intensity observed from a surface is directly proportional to the cosine of the angle between the observer's line of sight and the surface normal. As a result, the edges in thermal image become blurred.

In thermal cameras the output brightness of image pixel is derived from corresponding IR detector output. In the process the quantization occurs, which reduces the memory requirements and necessary processing power. In case of thermal cameras equipped with automatic contrast control such approach may result in a loss of information that can be extracted from observed scenery. Typical example is a "blinding effect" when a very hot (or very cold) object is present in the camera's field of view. If such an object is large enough, it will not be treated as noise and removed during image processing. Instead, the camera will adapt its settings to the object's temperature, thus the other important elements of the observed scene can be practically indistinguishable.

The following main features of thermal images can be distinguished [2, 11]:

- low resolution and low contrast (when compared to visual images),
- not clearly visible edges (diffused),
- low image sharpness (when compared to visual images),
- visible noise due to non-uniformity of array detectors,
- occurrence of pulse disturbances,
- color palette or grayscale representation different than human eye is used to see.

The features of thermal images sometimes make effective observation difficult or even impossible. What is more the "blinding effect" for example, may originate not from natural phenomenon but could be intentional actions to disturb observation. They sometimes make effective observation difficult or even impossible. What is more the "blinding effect" for example, may originate not from natural phenomenon but could be intentional actions to disturb observation.

The features of thermal images described above should be very similar in MWIR and LWIR spectral bands. We expect NIR spectral band images will have virtually the same features though such images can have some features of visual images as well.

2. Algorithm concept

Process of image enhancement does not, of course, increase the amount of information in the image. All of the information is already there, but by enhancing the image its certain features are modified thus making visual image easier to interpret and some details become discernible. Image enhancement techniques are mostly heuristic ones. They

include several image transformations, like gray levels distribution, contrast enhancement, noise reduction, edge detection and sharpening, spatial filtration, interpolation and many others.

The theoretical background for thermal image enhancement is, naturally, digital image processing theory. That does not mean, however, that certain procedure that work well for visual images can be directly adapted to thermal image processing. Differences between visual and thermal images described earlier make such direct approach ineffective. It is necessary then to carefully choose the applied methods and modify them [2, 4, 6].

The four stages of the developed algorithm are closely related to the previously defined features of thermal images [2]:

- noise removal,
- contrast enhancement,
- edge detection and sharpening,
- optimization of greyscale palette.

It should be mentioned that the sequence of image enhancing operations is important. If the sequence of operations is wrongly defined, the results may be far from expected and assumed level of image enhancement will not be achieved.

Noise reduction was applied at the initial stage of thermal image enhancement algorithm. The noise removal procedure reduces the number of pixels that have brightness levels significantly different than its close neighbourhood. Such pixels would likely be emphasized by subsequent procedure in the algorithm (e.g. identified as edges at the edge detection step) thus creating unwanted artefacts. Edge sharpening, in turn, is most effective when performed at this step of image processing, when the brightness levels are optimized and noise removed. Final step of proposed algorithm consists in greyscale palette optimization in order to obtain the image which resembles the natural one, meaning visible image.

The algorithm was so designed to effectively improve the quality of regular, properly recorded thermal images as well as those of degraded quality due to operator's error.

Noise filter (intended to reduce mainly pulse noise) uses contextual transformations. The applied filter is basically an adaptive median filter. Its operation depends on the brightness values of pixels surrounding actually transformed pixel of an image. If the brightness of the analyzed pixel is above maximal (or below minimal) brightness value found in the neighbour pixels, then the mean value of neighbour pixel brightness is assigned to it; otherwise the pixel brightness value remains unchanged. Such solution does not generate the problem of blurred edges, which is an inherent feature of contextual transformations. The operation of applied filter is adaptive because depends on the brightness distribution around analyzed pixel. As a result the areas free of noise remain unchanged by the filter and the shapes of edges remain also unaltered. The proposed filter is a reasonable compromise between efficiency of noise removal and amount of edge blurring, both important in terms of image enhancement process. The presence of the noise affects the image quality and total effectiveness of image enhancement, because remaining noise would be further processed during the next procedures and increased in magnitude. Edge sharpness, in turn, determines the detail discrimination level in the output image, so significant image blurring due to noise reduction would affect the effectiveness of edge sharpening in the following steps of the algorithm.

The main stage in tested algorithm is contrast enhancement. It is especially important stage, because, as it was mentioned before, thermal images are usually low-contrast ones. Most effective and most frequently used method of contrast enhancement is histogram equalization [5]. This method is effective when applied to visual images, but unfortunately when it comes to thermal images it improves rather background contrast not the contrast of an object. The utilized method of contrast enhancement is based on plateau histogram equalization [6]. It has all the advantages of typical histogram equalization but does not strengthen the contrast excessively. This is achieved by applying plateau threshold, which is determined automatically on the basis of image histogram properties. This approach makes it possible to enhance contrast in thermal images using quite limited computing power. The first step of the procedure is naturally histogram calculation. When it is done, the threshold value is determined. The proper choice of a threshold value greatly affects the results of contrast enhancement for thermal images. Local maximum values for a given image histogram must be known to calculate the threshold value. Only non-zero values of brightness function are taken into account and temporary histogram is created. In this data set local maximum values are searched for. In the next step median value is calculated for all local maximum values of this new temporary histogram and the calculated value is rounded to the nearest integral number, which is the final threshold value. In the next step this threshold is applied for the transformation of the source histogram. Every pixel count for the given gray level $P(k)$ is tested according to the following relation:

$$P_T(k) = \begin{cases} P(k), & P(k) \leq T \\ T, & P(k) > T \end{cases} \quad (5)$$

where k is the gray level and assumes the following values: $0 \leq k \leq 255$.

If the pixel count for a given gray level k is greater than threshold value, then the pixel count is limited to the threshold value, otherwise the pixel count for a given gray level k remains unchanged. New values of pixel count for gray levels are represented by $P_T(k)$ set of values. Final equalized histogram of transformed image (with contrast enhancement) is described by the $D_T(k)$ set of pixel counts for particular gray levels. $D_T(k)$ values of pixel count for each gray level k is given by the following equation:

$$D_T(k) = \left[\frac{255 \cdot F_T(k)}{F_T(255)} \right], \quad (6)$$

where $F_T(k)$ is accumulated value of pixel count for a given gray level k , calculated according to the following equation:

$$F_T(k) = \sum_{j=0}^k P_T(j), \quad 0 \leq k \leq 255, \quad (7)$$

The third step of the algorithm is edge sharpening procedure. It is usually performed by applying high-pass filter or image convolution with dedicated Laplace operator. Approach used by authors is different and consists of three main stages: low-pass filtering of source image, subtracting resulting image from original image, adding calculating difference image to the original image. This approach is sufficiently effective for the majority of thermal images and the computational complexity is significantly reduced.

The last step is the algorithm is optimization of the image palette. Thermal images are usually presented in "white-hot" standard, where the brighter pixels means higher temperature and dark shades indicate lower temperature regions. For the human eye the reverse representation of some IR images can be more natural [2, 10, 12].

3. Hardware implementation

Electronic unit of a thermal camera consists of three main modules [7, 8]: FPA module, control and digital image processing module and display module. General, simplified block diagram of electronic circuitry of a thermal camera is presented in Fig. 2.

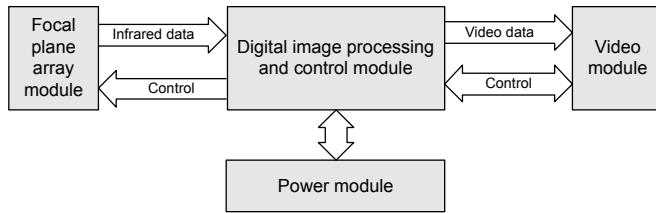


Fig. 2. Basic functional blocks of a thermal camera

FPA module contains mainly the detector array and A/D converter, transforming analogue signals form the array into digital form. Digital output is then fed to the digital image processing and control module, which is responsible for advanced signal processing by complex algorithms [7, 8, 9]. The resulting thermal image is transmitted to the display module, being usually a device for image presentation, like CRT or LCD display.

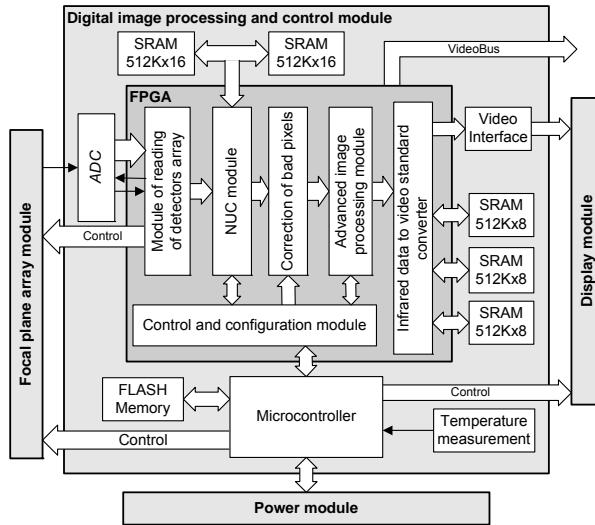


Fig. 3. Functional block diagram of digital image processing and control module with FPGA device

Main issue in hardware implementation of complex procedures for image data processing (like image enhancing algorithms) is high computational cost. As a result implementation of complex algorithms using general purpose processors and software usually does not bring satisfactory results. Complex algorithms for real time image processing are often executed by specially designed ASIC integrated circuits. Such architecture is designed for a specific application so the development time is rather long and expensive. Furthermore its functionality and flexibility is usually limited. Alternative solution can be based on reconfigurable FPGA circuits [7]. It is flexible and cost-effective approach and yet FPGA-based system is capable of complex, real time image processing. Additionally it offers favorable price to performance ratio.

Bearing those in mind the digital image processing and control module was designed and constructed around two main integrated circuits: a FPGA device and a microcontroller. Programmable FPGA device performs image data processing, which requires considerable computing power. It generates the control signals for array readout, performs NUC correction and bad pixel mapping, generates the control signals for display module and finally executes complex image processing algorithms. Microprocessor, in turn, performs remaining camera control functions and other tasks requiring low computational load. Block diagram of digital image processing and control module is shown in Fig. 3.

3.1. VideoBus link

VideoBus link for transferring the image data was implemented in a FPGA device. All image processing modules implemented a FPGA use VideoBus to exchange data, which is schematically presented in Fig. 4.

The application of VideoBus made it possible to alter the order of image processing operations without intervention in the implemented modules. Dynamic control of image processing is also possible because the modules themselves are so designed that they can become transparent and not modify the image data. Such concept greatly improves the flexibility of the design of image processing system. The VideoBus is equipped with diagnostic connector for the testing of each step performed by image processing module.

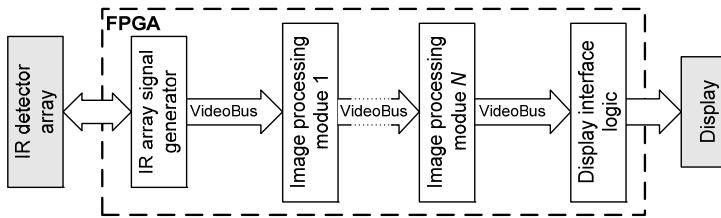


Fig. 4. Block diagram of image processing modules implemented in FPGA device

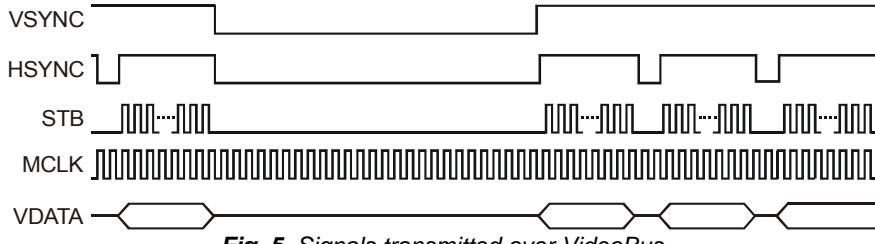


Fig. 5. Signals transmitted over VideoBus

VideoBus image data link consists of 14-bit data bus (VDATA), vertical (VSYNC) and horizontal (HSYNC) synchronization signals and strobe signal (STB). VSYNC signal assumes the high state through the transfer of a whole frame, whereas HSYNC signal goes high during the readout of consecutive rows. Change of STB from low to high state (rising edge) indicate that the valid (stable) data are available on the VDATA bus. The valid data are kept over the VDATA bus as long as the STB signal is in high state. VSYNC and HSYNC signals assume the high state before the first rising edge of STB signal. The timing sequence of VideoBus signals is shown in Fig. 5.

3.2. Point operations module

Point operations are the most common method used in the initial stage of image processing. It is basic and simultaneously the simplest image processing method. The idea of point operations is shown in Fig. 6.

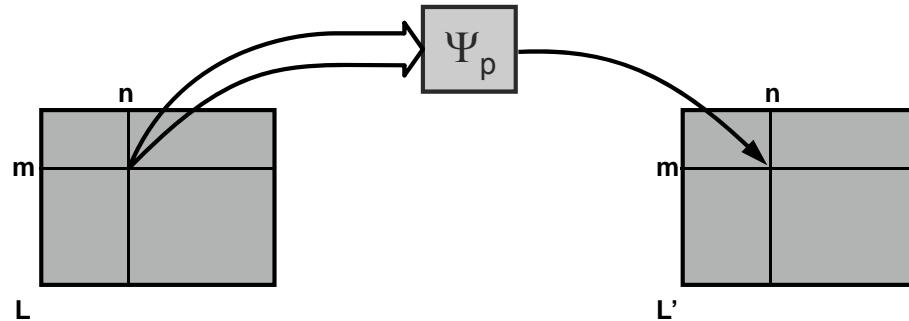


Fig. 6. The idea of point operations

Hardware implementation of point operations usually involve the usage of look-up-table (LUT) image transformations. The pixel value in LUT image processing is recalculated using the data table stored in memory. This procedure is graphically presented in Fig. 7.

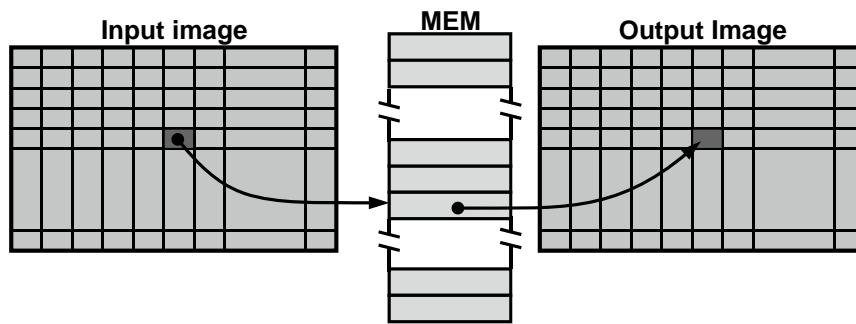


Fig. 7. The idea of LUT operations

The memory used for LUT operations (MEM in Fig. 7) can be ROM or RAM type. Storing the LUT table in ROM, however, increases the speed of LUT operations and lowers the resource usage. In such solution the LUT table is stored during the programming of FPGA device, which means that any change in the LUT requires re-programming of FPGA. In turn, by using RAM memory for LUT its content can be changed during the normal operation of the device without the need to re-program the core system (i.e. FPGA). The applied FPGA device has considerable amount of RAM memory, and as a result LUT data can be stored there, not using logic resources or external memory chip. LUT operations were described in VHDL language and implemented in FPGA device.

3.3. Convolution module

Convolution module is a functional block (described in VHDL language and implemented in FPGA device) which performs linear neighborhood operations. It was assumed to include this block in the data processing pipeline, which means that its input and output is VideoBus. The operation of this module consists in replacing the actual pixel value with a new one, calculated from its neighborhood. In other words the new pixel value is a normalized sum of an initial pixel value and its neighborhood, multiplied by weighing coefficients. The idea of neighborhood operations is presented in Fig. 8.

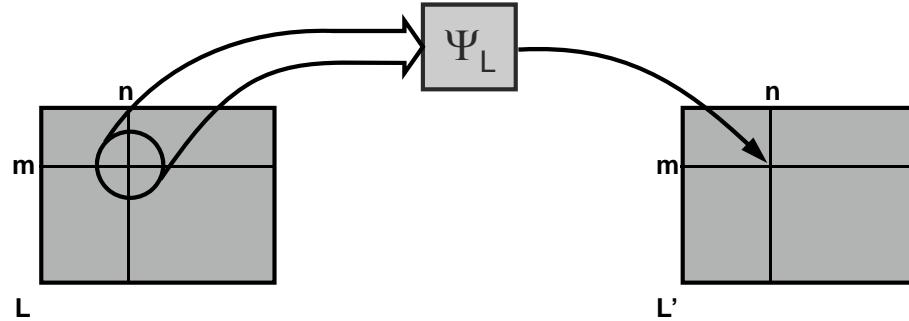


Fig. 8. Scheme of neighborhood operations

Convolution is an associative operation, which means that the following relation can be written:

$$(f * g) * h = f * (g * h). \quad (8)$$

The above relation means that by choosing the appropriate coefficients it is possible to divide large filter masks into smaller ones and perform the operation in several passes. For example, filtering with 5×5 mask can be replaced by two-pass filtering with 3×3 mask. As a result relatively simple hardware implementation can be used for different mask size in neighborhood operations.

Real-time convolution module must have delay lines (buffers) for continuous data flow and simultaneous access to the whole neighborhood of analyzed pixel and sufficient number of arithmetic units. The number of required buffers and arithmetic units depends on the size of the chosen mask. For example in case of 3×3 , 5×5 and 7×7 masks it is required to use 2, 4 and 6 buffers, 9, 25 and 49 multipliers and 4, 6 and 8 adders, respectively (it is theoretically possible to use fewer adders, but it increases the overall latency).

Convolution module for 3×3 mask size (shown in Fig. 9) was described in VHDL language and implemented in FPGA device. Total delay caused by neighborhood operations performed by this module equals the time of two image lines plus four pixels.

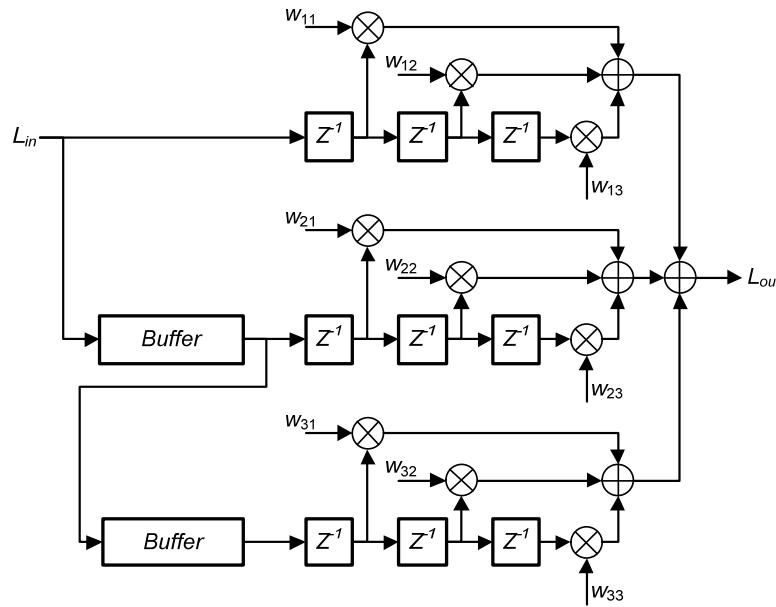


Fig. 9. Block diagram of convolution module using 3×3 mask

3.4. Histogram calculation module

The module for histogram calculation is one of more complex, even though the procedure itself does not alter the input image (it remains unchanged). Histogram calculation is however, important for many steps of image processing, like histogram equalization or binarization. The idea of operation of histogram calculation module is quite simple and consists in counting the number of instances of particular pixel values in the image. But the realization of incrementing operation requires three main steps: reading the value of a memory cell, incrementing it by one and then store the new value in memory. The pixel brightness is defined by the address of a memory cell in which the number of detected instances is stored. It should be mentioned that the module must sequentially perform three operations in single calculation cycle. Those operations cannot be performed in parallel and they have to be executed one by one. As a result fast memory of considerable size must be used. Simplified block diagram of histogram calculation module is presented in Fig. 10.

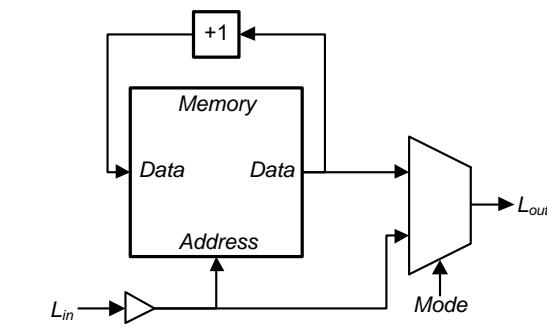


Fig. 10. Block diagram of histogram calculation module

Apart from the blocks shown in the above picture, there is also a control block required. It generates control signals for asynchronous memory read and write operations and data processing. The designed circuit is informed about incoming image data by a special synchronization signal. Additionally, when the control circuit detects the end of image data transfer, it changes the operation mode and initiates sending the histogram data to the external modules. To do so, the control module have to generate several control signals for the external data bus used for the transmission of histogram data. After that, the histogram memory is cleared and the overflow correction circuit is reset. Finally the module is set into a ready state and awaits the next image.

All the components performing the histogram calculation have been described in VHDL language. The whole module was implemented in a FPGA device and it uses external asynchronous SRAM memory, located on the PCB. In order to test the performance of histogram calculation, several other modules were implemented, responsible for the generation of test images and visualization of results. Module used for the generation of test images can output different test patterns, consisting of horizontal, vertical and skew strips with gradually changing brightness. Additionally the certain amount of noise could be added, provided by pseudo-random number generator. Of the test signals, the one with skew strips was the most interesting for its cognitive value. In this case no adjacent pixels have the same brightness level, thus the change in pixel brightness takes place at each clock signal of the VideoBus. This case is perfectly suited to evaluate the performance of histogram calculation module, because it requires highest processing speed.

The operation of the module was verified by comparing the calculated histogram with the one derived by special software, written for this task in MATLAB environment. Series of tests were conducted in order to thoroughly evaluate the module, using different test images with various complexity.

The designed modules operated properly in all the tests. There were no errors recorded in histogram calculations from the image measuring 640x480 pixels, with 16 bit brightness depth. Furthermore the module successfully calculated histogram data from 640x480 video stream at 60 frames per second, which means data rate nearly 300 Mbit/s. Data conversion at such rate is a serious task even for the high-end desktop processors. The histogram calculated for the last analyzed frame is ready just one clock pulse after the frame transmission is finished. Such a short delay is practically impossible to achieve in a classic microprocessor-based system. It can be concluded from simulation tests, that the module can process the 640x480 video stream up to the frame rate of 130 fps, thus being suitable for very demanding applications. The entire module implemented in FPGA occupies about 150 logic cells, which is only a small part of available resources.

4. Results and conclusions

The image enhancement algorithm for thermal cameras presented in this paper and included in the digital image processing system of an IR camera is fast and flexible. It is capable of real-time operation without introducing significant delays. It should be mentioned that for continuous data processing the delays introduced by the system should be constant, which means that they cannot accumulate in time. To achieve this, the advanced implementation methods must be applied, including parallelization of image processing operations. Considering the speed and parallelization requirements the hardware implementation was realized using ALTERA's EP2C35F672 FPGA device. This chip provides the required efficiency at relatively low power consumption. The selected circuit has a sufficient number of terminals (I/O ports) and logic cells. Additionally it has 483 840 bits of RAM memory, four PLL circuits for signal generation and 35 embedded multipliers. It should be emphasized that the presence of embedded multipliers allows for the programming of functional blocks inside FPGA, which are capable of high speed arithmetic operation with relatively low usage of resources.

The simulations and laboratory experiments proved the correct operation of the designed system for digital image processing and control in executing the sophisticated image enhancement algorithm. The timing analysis of the modules implemented in FPGA device also confirmed that the image enhancement algorithm implemented in the system conforms to the initial design requirements, namely the real time processing of 640x480 video stream.

One of the more important tests was the subjective evaluation of the results of thermal image enhancement algorithm. The judgment was based on direct visual comparison between original and processed images, taken for different objects and observation conditions. The processed images were perceived by a human observer as better ones. Sample images before and after image processing by the implemented algorithm are presented in Fig. 11.



Fig. 11. Thermal images: original (left) and processed by image enhancement algorithm (right)

REFERENCES

- [1] Accetta J. S., Shumaker D. L., "The Infrared and Electro-Optical Systems Handbook", Ann Arbor MI and SPIE Press' Bellingham, WA, 1993.
- [2] Dulski R., Powalisz P., Kastek M., Trzaskawka P., "Enhancing image quality produced by IR cameras", Proc. of SPIE, Vol. 7834, 783415, 2010.
- [3] Holst, G. C., "Testing and Evaluating of Infrared Imaging Systems", SPIE Optical Engineering Press, 1998.
- [4] Cheng H. D., Shi X. J., "A simple and effective histogram equalization approach to image enhancement", Digital Signal Processing 14, pp. 158-170, 2004.
- [5] Weith-Glushkod S., Salvaggio C., "Quantitative Analysis of Infrared Contrast Enhancement", Proc. of SPIE Vol. 6543, 65430S, 2007.
- [6] Bing-Jian W., Shang-qian L., Qing L., Hui-xin Z., "A real-time contrast enhancement algorithm for infrared images based on plateau histogram", Infrared Physics & Technology, 48, pp. 77-82, 2006.
- [7] Sosnowski T., Biesczad G., Kastek M., Madura H., "Digital image processing in high resolution infrared camera with use of programmable logic device", Proceedings of SPIE Vol. 7838, 78380U, 2010.
- [8] Biesczad G., Sosnowski T., Madura H., Kastek M., Barela J. "Adaptable infrared image processing module implemented in FPGA", Proceedings of SPIE Vol. 7660, 76603Z, 2010.
- [9] Biesczad G., Sosnowski T., Madura H., Kastek M., Barela J., "Image processing module for high-speed thermal camera with cooled detector", Proc. SPIE 8012, 80120L, 2011.
- [10] Dulski R., Madura H., Piatkowski T., Sosnowski T., "Analysis of a thermal scene using computer simulations", Infrared Physics and Technology, 49 (3 SPEC. ISS.), pp. 257-260, (2007).
- [11] Dulski R., Madura H., Niedziela T., Sikorski Z., "Theoretical model of thermodetection system", Optica Applicata, 30 (2-3), pp. 451-453, (2000).
- [12] Dulski R., Sosnowski T., Polakowski H. "A method for modelling IR images of sky and clouds", Infrared Physics and Technology, 54 (2), pp. 53-60, (2011).