Електрична кола, електрични системи и обрада сигнала

Electric circuits and systems and signal processing

Binary Mask based Crowd Counting analysis using Multi-Column Convolutional Neural Network

Lara Kašca and Ana Gavrovska, *Member, IEEE*

***Abstract*—Crowd counting has attracted significant attention recent years since it is valuable to estimate the number of people in a crowd in numerous applications, especially the ones related to video surveillance. Artificial intelligence, especially convolutional neural networks, became a part of such applications. In this paper, multi-column convolution neural network implementation has been analyzed, where the output is density map. The number of people is estimated as a sum of the map. In this paper experimental analysis using binary mask based postprocessing and ShanghaiTech dataset is performed. The obtained results seem promising in dealing with unwanted texture details related to irrelevant regions as in the case of greenery.**

***Index Terms*— Image processing, frame, crowd counting, density map, computer vision, MCNN, binary mask.**

I. Introduction

CROWD counting using a single image or a frame has been popular over the years [1-4]. It implies rather approximate techniques, where instead of determining the exact number of people in the crowds, estimation techniques can be applied. The need to develop these techniques arose due to the desire to find out how many people are at public gatherings such as rallies, traffic jams, disasters, protests or those mass events where this information could not be obtained based on the number of tickets sold. It can be considered useful for behavioral analysis when counting is performed in specific moments.

Counting all heads in a crowd image may be difficult or impossible. This is the reason why modern crowd counting concept relies on using a grid in order to segment the whole picture. The first step after counting the people manually in a few segments is to calculate the average number of people per segment after initial analysis. Then, the average number of people is multiplied by the total number of segments. This technique is called Jacobs' technique after journalism professor Herbert A. Jacobs at the University of California, Berkley. In the 1960s he applied this for counting students who protest the Vietnam War [1]. Also, he described a light and a dense crowd by cases having one person per 10 and 4.5 square feet area, meaning approximately 0.93 and 0.42 square meters area, respectively [1]. A slightly more sophisticated approach is to do an extrapolation adapted to the current part of the image, because not all parts are the same. However, the solutions are not linear and the mentioned approaches will not always give a satisfactory result. In 1995, one of the largest protests in American history, called the Million Man March, was held in Washington to raise public awareness of the position of African Americans in America. To this day, scientists are dealing with this event and how many people were in Presidential Park at the time, and estimates range between 400,000 and 1.1 million, which is a very large range giving inaccurate information [2].

Advances in technology today speak of much more elegant and impressively precise methods and calculations involving computer vision. Recent advances favorize convolutional neural network (CNN) in crowd counting [5-6]. In this paper we analyze some of the disadvantages of using a multi-column CNN. The experimental results presented in this paper shows the possibility to employ binary masks for CNN based refinement of the people counter which is in accordance to recent experiments with contextual representation [7-9].

The paper is organized as follows. After the introduction, a brief review regarding crowd counting methods is given. In Section II general usage of CNN is explained, as well as its application in crowd counting. Details regarding the simulation using binary mask approach and multi-column CNN performed in this paper are explained in Section III. The experimental results followed by discussion are presented in Section IV. Finally, in Section V conclusions are given.

II. Crowd counting and CNN

A. Crowd counting

Current methods from the literature related to crowd counting can be classified into one of the following four groups: detection-based methods, regression-based methods, density estimation-based methods and CNN-based methods [3-11]. In previous research on the CNN based topic, there are two main approaches. The first one is to pass an input image through the network architecture, and to give an estimation of the number of people as an output result. The second approach, used here, is to forward an image to the input of a network, and to get output density maps that needs to be processed further in order to get the final result. The reason

Lara Kašca is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mail: larakasca@yahoo.com).

Ana Gavrovska is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mails: anaga777@gmail.com; anaga777@etf.rs).

why this approach may seem better is that it provides more information than just one numerical value. The density map provides data on the density of different parts of analyzed image. Ground truth density map (GTDM) can be calculated using k nearest neighbor (KNN) method, and the corresponding neural network output is estimated density map (ESDM). An example of GTDM is presented in Fig. 1.

Fig. 1. Input image (left) and corresponding ground truth density map (right).

Density map used as a reference is determined using delta function δ at locations of interest convoluted with Gaussian kernel G in order to obtain a continuous representation for each image pixel x:

$$GTDM(x) = \sum_{i=1}^{N} \delta(x - x_i) * G\sigma_i(x). \quad (1)$$

where x_i represents image pixel of interest where a person's head is found, N is the total number of people, and σ_i is variance proportional to mean distance value $d_{i,mean}$ for each x_i ($0.3*d_{i,mean}$). The mean distance, $d_{i,mean}$, is the average value of distances between each pixel x_i and its k nearest neighbors.

B. Convolutional Neural Network architecture

Convolutional neural network (CNN) is a neural network mostly used for visual data, such as images. As the name suggests, it uses the convolution operation in layers, usually in the first layers of a deep architecture. Input image is filtered (convoluted) by multiple filters, where CNN has the ability to represent images in more appropriate form having in mind relevant features. What can be applied in addition to the usual convolution is padding the input due to filtering application, or one can use strided convolutions that have the opposite effect to reduce the output. Convolutional layers are applied to learn features like edges using smaller segments of input data.

In addition to convolutional layers, there is also a pooling layer that is applied after the convolutional layer. It has the task of choosing max, sum or average value, and only passes that one value in order to further reduce the relevant data. The third type of layers that are optional are fully connected layers that usually take the last matrix output from convolutional-pooling layer sequence, and flattens it into a row or a vector. It can further pass it to the classic neural network. Fully connected layer often enables learning correlations between previous image representations called feature maps. Back-propagation is used for training via a number of iterations or epochs using available input and corresponding reference or target. The advantage of CNN application is a smaller number of parameters for determining and sharing, i.e. less feature engineering than in a standard NN concept.

In order to overcome the issue of filter size, instead of single-column, one may apply parallel CNN architecture with final layer for merging the results of each CNN branch. This can be particularly valuable for different person's head size in a crowd counting challenge. In Fig. 2, multi-column CNN (MCNN) architecture is illustrated. Moreover, binary image or binary mask (BM) may be used in pre- and/or post-processing to improve the estimation of the number of people [9]. Here, in Fig. 2, a post-processing block is presented, where final result is a density map where the mask removes irrelevant details from the estimated map.

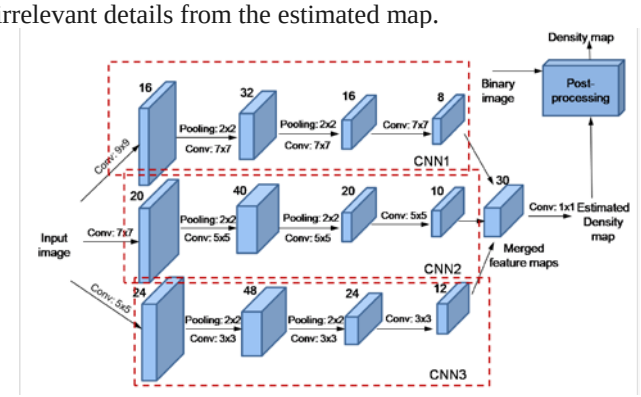


Fig. 2. Three CNNs making MCNN architecture with post-processing.

III. Simulation

This paper experiments with ShanghaiTech dataset of crowded images consisted of two subsets A and B [6]. The subset A contains 300 train and 182 test images, while the subset B has 400 train and 316 test images. In Fig. 3 the train examples are presented. The subset A is made up of images downloaded from the internet of densely distributed people, while they are less dense crowds found in subset B. The dataset statistics is presented in Table I. In addition to ShanghaiTech set, we also performed manual labeling of people in subset C for the purpose of testing, where crowd images were downloaded from internet.

Fig. 3. Examples of images from the train sets of subset A and B.

TABLE I
THE DATASET STATISTICS

Subset (resolution)	Number of images	Min - max number of people (average)	Total number of people
A (Various)	482	33-3139 (501.4)	241677
B (768x1024)	716	9-578 (123.6)	88488
C (Various)	4	60-817 (322)	1288

The architecture in this paper is a multi-column CNN consisted of 3 parallel CNNs. It uses max pooling layers which apply to 2x2 regions and ReLU as an activation function. Images used in training are not used while testing. The network output is estimated density map (ESDM), where GTDM from (1) is used as input. The training is performed according to Euclidean distance between the maps.

A. Experimental crowd analysis

In the first phase of experimental analysis, MCNN is trained. Two subsets, A and B, are trained for single-image estimation. Metrics, like mean absolute error (MAE) and the

mean square error (MSE), can be used for the training:

$$MAE = \frac{1}{M}\sum_{m=1}^{M}\left|p_m - p_{m,est}\right|, MSE = \sqrt{\frac{1}{M}\sum_{m=1}^{M}\left|p_m - p_{m,est}\right|} \quad (2)$$

where M is the number of images, p_m is the number of people in m-th image and $p_{m,est}$ is the estimated number of people in the same image. Here, the training is performed using MAE. CNN implementation and crowd analysis are performed using Python 3.7 and Pytorch [12], where for visualization purposes visdom is applied [13]. Used configuration is Intel i5 8400, Nvidia GeForce GTX 1050 Ti, cuda 10.1. For MCNN evaluation relative ratio is calculated:

$$R[\%] = \left|p_m - p_{m,est}\right| / p_m, \quad (3)$$

using A, B and C images. The number of people, p_m, can be considered in two ways, as a number of manual labels or as a sum of GTDM. The first way is used in the first phase, and the second way is used in the following phase.

In the second phase binary masks (BMs) for GTDM and ESDM are obtained by thresholding T (T=0 for GTDM and T=0.01 for ESDM). The thresholded ESDM can be considered as MCNN mask. In this paper, the proposed method for BM calculation for the post-processing consists of five steps: original image resizing, median residual calculation, extreme removal, local representation (histogram, histogram of gradients (HOG), sum) comparison, and morphological processing. In the first step, input image is resized according to ESDM, or MCNN output. The difference between the input and its median filtered version is calculated to obtain the median residual image. The next step is extreme removal, where the extreme details are considered to be the ones with the most highest values of difference and relative difference between the input and the filtered input. The fourth step consists of calculation of histogram, HOG and array of sums per rows and columns for the pixel neighborhood of size 5x5 (block size). Euclidean distances between the normalized histograms, the HOG and the sum representations of the blocks, are calculated and thresholded with 30%, 60% and 50-75% of the maximum values, respectively. Then, the intersection I of the three images is found, where only pixels with dense concentration within blocks are kept and used as seeds in the intersection image for region growing segmentation [14], obtaining binary image I_g. The final step is dilatation of I_g, where structuring element is square of size 15, obtaining BM1. With additional dilatation of size 10, the mask denoted as BM2 is obtained. In order to get the estimated number (est. num.) of people, the sum of density map is calculated according to the positions described by the binary mask, like BM1 or BM2. Also, the relative ratio according to (3) is calculated.

IV. Experimental results

In Fig. 4 MAE values through epochs are presented for training and testing in the case of A and B subset. Best performance is obtained for epochs 1715 and 939, respectively, where for further experiments the model based on A subset is used. The layout of the visdom window during training is shown in Fig. 5, where GTDM and ESDM for an image are presented. In Table II a comparison between obtained MAE based results and the results obtained in [6] is given showing slight variation. Besides ShanghaiTech dataset additional inputs are made according to manual labeling as shown in Fig. 6. In Table III calculated relative ratio results are presented for A, B and C image samples.

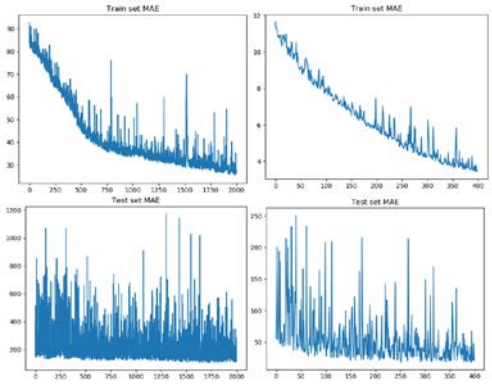


Fig. 4. *MAE values through epochs (on the left for subset A, on the right for subset B).*

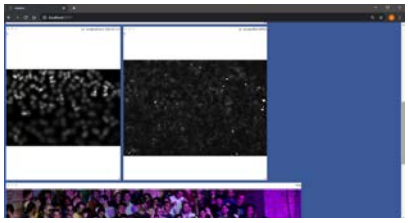


Fig. 5. Visdom window with GTDM and ESDM visualization.

Fig. 6. Original (left) and manual labeling (right) of image1, Ctest taken from [15].

TABLE II
MAE for multi-column CNN

Subset/ Experiment	MAE (MSE) [6]	Obtained MAE in this paper
A	110.2 (173.2)	108
B	26.4 (41.3)	18.5

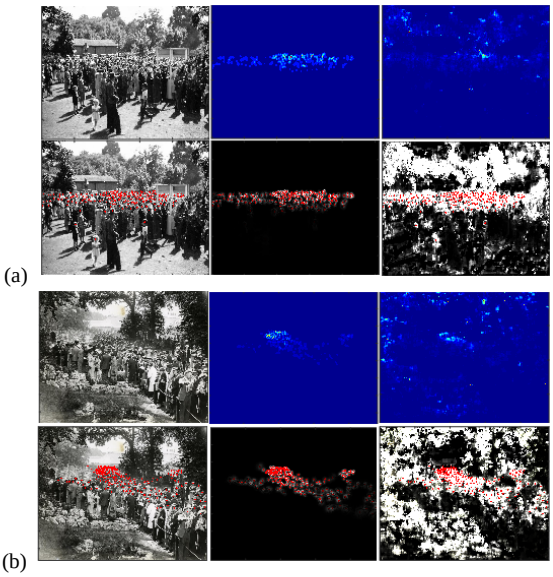
TABLE III
Relative ratio results for MCNN

Image No. (class)	Number of labels	MCNN count	Ratio R [%]
Image0,Atrain	1546	1420	8.2
Image0,Atest	172	598	**247.7**
Image3,Atest	211	654	**210.0**
Image5,Atest	431	435	0.9
Image5,Btest	57	92	61.4
Image24,Btest	70	64	8.6
Image1,Ctest	304	273	10.2
Image3,Ctest	817	557	31.8

It can be observed in Table III that MCNN count results are satisfying. Example of the successful density representations and superimposed maps and labels is shown in Fig. 7. On the other hand there are cases where significant difference between number of labels and ESDM exists, meaning relative ratio even higher than 200% as shown in Table III. These examples are presented in Fig.8. The more complex content produces large differences, and the errors here are due to texture related to greenary regions. This is presented by white pixels in lower right representations in Fig.8(a) and Fig.8(b), where red points are superimposed labels related to people.

The results from the second phase and the calculation of BM1 are illustrated in Fig. 9 (a)-(b). The obtained results are shown in Table III, showing the significance of semantic approach. Here, BM1 gives up to 13% relative ratio, where BM2 shows that there is possibility for firther improvements in some cases according to selection of structuring element.

Fig. 7. Image 5 from test A representations: original (labels:431), GTDM (sum:428.6), ESDM (sum:435.1) (upper row) and the representations with labels superimposed, respectively (lower row).



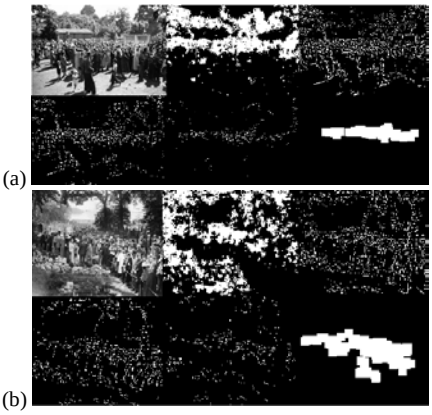
(a) (b)

Fig. 8. The representations for: (a) image0, Atest (labels:172, GTDM-sum:171.8, ESDM-sum:598.7); (b) image3, Atest (labels:211, GTDM-sum:210.4, ESDM-sum:654.5)

The parallel networks are needed because they enable detection of objects of different sizes. Thus, three filters of sizes 3x3, 5x5 and 7x7 are applied to detect heads of lower and higher size regardless of distance from the camera being used. Higher number of parallel branches may improve results in some cases having in mind the spatial resolution of an image or object of interest.

The networks merged in a parallel architecture may have their own goals in order to deal with intra-class detections (classes of individuals with similar object sizes) and inter-class issues where the background details resemble objects of interest, i.e. individuals. The parallel structures may be even considered to set goals to deal with background false detections [16], and to train what should not belong to a foreground.

The practical purpose of such parallel networks is not only to make more accurate detections for surveillance and similar tasks, but also to further exploit the parallelization strategy. Namely, if similar processes occur in different branches one may have an improved insight into system scalability and dependency between minor challenges representing the current limitations. Here, model-parallelism is applied where the similar structure is applied with training on the same data. Data-parallelism is also an available solution, where one can analyze subsets of big data across different branches or channels [17]. Computing performance improvement and memory distribution are also possible in communications using scalable architectures and different channels.



(a) (b)

Fig. 9. Input, ESDM mask, median residual, extreme removal, local representation, and morphological result for: (a) image0 and (b) image3.

TABLE III
RELATIVE RATIO RESULTS FOR DIFFERENT BINARY MASKS

Binary mask (BM)	image0 - est. num. (R [%])	image3 - est. num. (R [%])
Ground BM	171.8 (-)	210.4 (-)
Estimated BM	598.7 (248.5%)	654.5 (211.1%)
Proposed BM1	**184.2 (7.2%)**	**183.7 (12.7%)**
Proposed BM2	215.2 (25.3%)	**212.3 (0.9%)**

V. CONCLUSION

CNN as a concept has great potential to obtain automatic estimation of the people number in a crowd. Here, a parallel network architecture with post-processing step is used for detection task. The binary mask (BM1) based approach shows the possibility to refine results, or to even show when the results of CNN is not satisfying.

Future analysis should be made towards contextual deep learning solutions related to more semantic segmentation

based methods. More complex architecture could be made for making accurate binary mask estimations, where the concept of parallelism may provide scalable solutions with improved computing performance and memory distribution.

ACKNOWLEDGMENT

This research is supported and partially funded by Serbian Ministry of Education, Science and Technological Development.

REFERENCES

[1] J. Weiss, "How reports can estimate the number of people in a crowd," Int. Journalists' Network, 2013. https://ijnet.org/en/story/how-reporters-can-estimate-number-people-crowd (*last accessed* 12.07.2020.)

[2] Million Man March, https://www.britannica.com/event/Million-Man-March (*last accessed* 12.07.2020.)

[3] S. Ali, and M. Shah, "A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-6, 2007.

[4] K. Chen, C.C. Loy, S. Gong, and T. Xiang, "Feature mining for localised crowd counting," in *Proc. Brit. Mach. Vis. Conf.* - BMVC, vol. 1, no. 2, p. 1-3, Sept. 2012.

[5] C. Wang, H. Zhang, L. Yang, S. Liu, and X. Cao, "Deep people counting in extremely dense crowds," in *Proc. of the 23rd ACM international conference on Multimedia*, pp. 1299-1302, 2015.

[6] Y. Zhang, Z. Desen C. Siqin, G. Shenghua Gao, and M. Yi, "Single-image crowd counting via multi-column convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 589-597, 2016.

[7] S. Huang, X. Li, Z. Zhang, F. Wu, S. Gao, R. Ji, and J. Han, "Body structure aware deep crowd counting," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp.1049-1059, 2017.

[8] W. Liu, M. Salzmann, and P. Fua, "Context-aware crowd counting," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition* - CVPR, pp. 5099-5108, 2019.

[9] S. Jiang, et. al, "Mask-aware networks for crowd counting," *IEEE Trans. on Circuits and Systems for Video Technology*, pp. 1-10, 2019.

[10] H. Bai, S. Wen, and S.-H. G. Chan, "Crowd counting on images with scale variation and isolated clusters," in *Proc. of the IEEE International Conference on Computer Vision Workshops* - ICCVW, pp. 1-10. 2019.

[11] J. Wan and A. Chan, "Adaptive density map generation for crowd counting," in, pp. 1130–1139, 2019. in *Proc. of the IEEE Int. Conf. on Computer Vision* - ICCV , pp. 1130-1139, 2019.

[12] Pytorch, https://pytorch.org/ (*last accessed* 03.06.2020.)

[13] Visdom https://pypi.org/project/visdom/0.1.7/ (*last accessed* 03.06.2020.)

[14] R.C. Gonzalez, and R.E. Woods, *Digital image processing*, 2007.

[15] Sample https://stampaday.wordpress.com/2017/10/13/the-death-of-king-bhumibol-thailand-post-special-sheet-2016/ (*last accessed* 12.07.2020.)

[16] B.Y. Lin and C.S. Chen, "Two parallel deep convolutional neural networks for pedestrian detection," In *Proc. of the IEEE International Conference on Image and Vision Computing New Zealand* (*IVCNZ*), pp. 1-6, November, 2015.

[17] S. Lee, D. Jha, A. Agrawal, A. Choudhary, and W.K. Liao, "Parallel deep convolutional neural network training by exploiting the overlapping of computation and communication," In *Proc. of the 24th IEEE International Conference on High Performance Computing* (*HiPC*), pp. 183-192, December, 2017.

Distance metric comparison for people monitoring across multi-camera views using ternary encoding

Katarina Popović and Ana Gavrovska, *Member, IEEE*

***Abstract*— Person monitoring is based on re-identification process, where it is important to establish correspondence between the person representations in different frames. The feature vectors may use ternary encoding like in local maxima occurrence representation, where color and texture features are implemented. In this paper three different distance metrics are compared after ternary encoded feature computation for people's pattern recognition. Moreover, since a person is usually a moving object, two different resizing approaches are distinguished. The color, the texture and the combined (color-texture) feature vectors are analyzed for the purpose of differentiation between image sets corresponding to different persons obtained across six camera views. The results show how the differentiation in multi-camera network can be affected by the choice of resize technique and distance metric.**

***Index Terms*—Video, computer vision, video surveillance, distance metric, resize, re-identification.**

I. Introduction

Person re-identification is challenging and a very important task in video surveillance. It implies detection followed by identification of the same person in different frames [1-3]. CCTV (Closed-circuit television) or video surveillance networks commonly rely on several cameras for such monitoring across overlapping and non-overlapping camera views.

One of the most relevant issues in pedestrian monitoring is to deal with resolution and picture quality. Surveillance cameras are usually placed in high spots, and this results in a low resolution representation of pedestrians. Moreover, pedestrians are often detected oriented with their side or back towards the camera. In these situations biometric characterization via facial characteristics is not reliable. Inadequate camera coverage, different illumination conditions, occlusions and similar may also affect the person re-identification.

Over the time different features for person re-identification have been explored [1-3]. Descriptors for the identification can be based on color, shape or texture. Also, the feature vectors can be derived from global and local approaches [2]. One of the most popular methods for re-identification is based on ternary encoded color and texture features [3]. Besides feature engineering, adaptive metric learning [4] and convolutional neural networks [5-6] have been employed to perform the re-identification. Nevertheless, resizing of the region of interest (ROI) representing a person is often used [2]. The resizing is usually performed in a fixed manner, meaning the input (ROI) for the re-identification procedure is set to a predefined resolution.

In this paper the goal is to analyze different resizing that may affect people differentiation across multi-camera views, even if color seems to be enough to perform the differentiation. Three types of feature vectors are applied - color, texture and LOMO (local maximal occurrence) based one. The LOMO represents the combination of the color and the texture feature vector. The feature vectors are obtained using ternary encoding [3]. Here, experimental analysis is performed using multi-camera views, where two different resizing techniques are applied. Also, three common distance metrics (Euclidean, City-block, Minkowski) are tested.

The paper is organized as follows. After the introduction, a brief explanation regarding color and texture descriptors is given in Section II. The details of the simulation performed in this paper are explained in Section III. The obtained experimental results followed by discussion are presented in Section IV. Finally, the conclusion is given in Section V.

II. Multi-camera monitoring using color and texture features

A. Multi-camera views in pedestrian monitoring

Multi-camera surveillance is a field close to computer vision. It is well-known that this enabled the development of intelligent video surveillance systems. Mostly, a person is monitored in an environment covered by multiple cameras. In Fig. 1 an illustration of two-camera monitoring is presented.

The position of cameras may vary and this is the reason why methodologies for re-identification should be tested in a multi-camera environment. The pedestrians being monitored are moving objects, and they may not been caught by each of the camera properly. They can be oriented towards a camera with their back or side, or they can even not be caught due to occlusions. They can carry bags, walk a dog or change their clothes. These are only some of the reasons why intelligent monitoring of a person is not an easy task.

Katarina Popović is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mail: katarina.popovic994@gmail.com).

Ana Gavrovska is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mails: anaga777@gmail.com; anaga777@etf.rs).

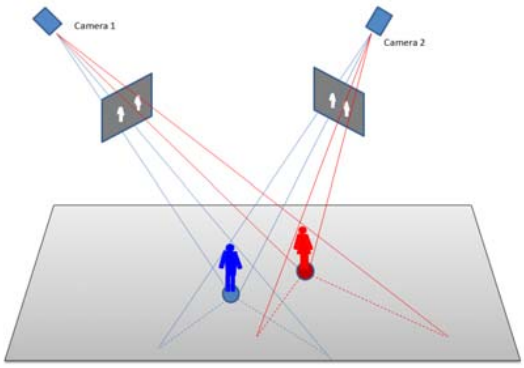


Fig. 1. Multi-camera monitoring.

Re-identification of a person can be performed using a single camera or multiple cameras. For example, it means that if a camera is switched from Camera 1 to Camera 2, a person should be identified with the same ID before and after the camera switching according to the person characteristics. For example, these characteristics may correspond to color or texture features of the person's clothes.

B. Color and texture feature vectors

The most relevant features for identification tasks seem to be color and texture [1-3]. Here, local maximal occurrence (LOMO) is used as one of the image content representations which combines both color and texture [3]. It is characterized by scale-invariance property which is particularly relevant in video surveillance tasks due to motion of an object/person being monitored. For the re-identification only the region of interest (ROI) is being analyzed. In this case ROI represents the bounding box around the person being checked.

The ROI can be interpreted as a pattern in a pattern recognition task. If a reference pattern, ROI_{ref}, is available one may compare it to a ROI obtained as a result of image segmentation from a different frame regardless the camera. The comparison verifies whether the matching exists or not. An example of ROI_{ref} is presented in Fig.2. It corresponds to one camera, while ROI1 and ROI2 correspond to another camera. Matching is expected between ROI_{ref} and ROI1.

The two ROIs (or the images in further text) have to be represented as adequate feature vectors. One common way to do this is by using blocks or patches of the two images. For images in Fig.2 resizing is performed according to maximum number of rows and columns (resizing to 88x33 pixels), as well as filtering with Retinex filter [3].

The first step after preprocessing is mapping from RGB (R - Red, G - Green, B - Blue) to HSV (H - Hue, S - Saturation, V - Value) color space. After the HSV mapping, the values from three channels (H, S and V) at pixel position (i, j) are transformed into one matrix value using (1):

$$I_{HSV,q}(i,j) = H(i,j)\cdot 8^0 + S(i,j)\cdot 8^1 + V(i,j)\cdot 8^2. \qquad (1)$$

The next step is partitioning, where matrix $I_{HSV,q}$ is divided into blocks or patches. The size of blocks is 10x10, where overlapping is 50%. If an image is of size 88x33 it can be presented by 80 blocks (16x5 block representation). Each block or patch is then represented as a column vector of 100 values forming color based patches of 100x80 pixels size. The column vectors belonging to three images (ROI_{ref}, ROI1 and ROI2) are then concatenated, forming color based patches of size 100x240 pixels. This is illustrated in Fig. 3. The color patches of 100x80 per image are then transformed to sparse histogram representation of 512x80, where 512 represents total number of bits per channel (8bit x 8bit x 8bit). So, for each column vector representing block, histogram is computed. After computing histogram, image pooling (downsampling) is performed (e.g. from image 88x33 pixels image of 44x16 pixels is obtained). The process is repeated for the number of scales (in this case maximum number of scales is 3), making the final representation through the patches by concatenating the color patches for each scale. This represents the calculated color feature vector of length 11776 per image. The steps in data reshaping for color based approach are briefly illustrated in Table I, for better understanding.

Similarly, texture based feature vector for each image is calculated. SILTP (Scale-Invariant Local Ternary Pattern) is applied for texture characterization.

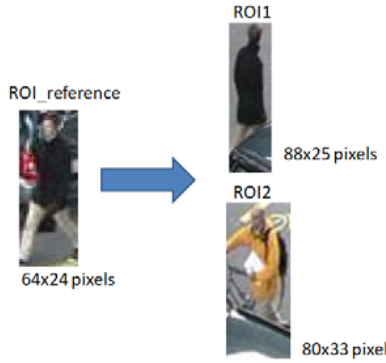


Fig. 2. An illustration of comparison between ROI_{ref} and two ROIs (ROI1 and ROI2), where the correct matching should be ROI1.

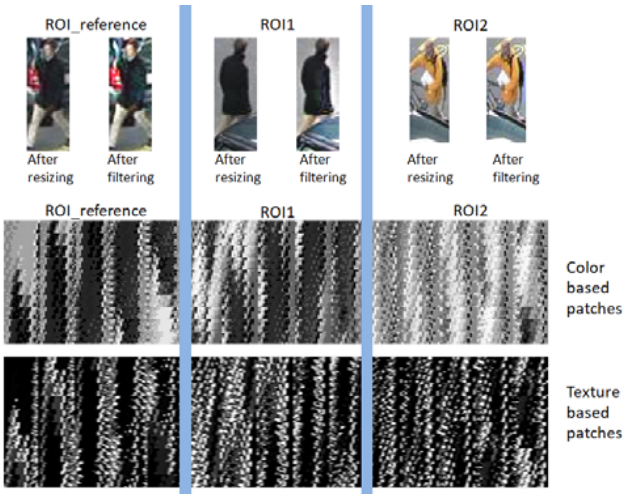


Fig. 3. Color and texture based patches for the pattern recognition task. The blue margins denote borders between data which correspond to three images (ROI_{ref}, ROI1, ROI2).

TABLE I
RESHAPING OF COLOR PATCHES

No.	Description	Data size
1	Image size for scale 1	88x33
2	Patches for three images	100x240 (100x80x3)
3	Sparse histogram representation	512x240
4	Reshaping according to blocks per rows and columns	512x16x5x3
5	Sum per columns	512x16x1x3
6	**Feature vector length for scale 1**	**8192x3**
7	Pooling/Image size for scale 2	44x16
8	Patches for three images	100x42 (100x16x3)
9	Sparse histogram representation	512x42
10	Reshaping according to blocks per rows and columns	512x7x2x3
11	Sum per columns	512x7x1x3
12	**Feature vector length for scale 2**	**3584x3**
13	Pooling/Image size for scale 3	22x8
14	**Total length of color feature vector x 3 images**	11776x3

First step in SILTP implementation is mapping RGB images to grayscale versions. SILTP originated from LBP (Local Binary Pattern), where the LBP approach is well known in pattern recognition [7]. When using LBP each image is divided into blocks usually of size 3x3. Pixel located in the centre of a block is then compared with its neighbors based on pixel grayscale value. If a neighboring pixel value is larger than or equal to the pixel value located in the block centre, it is set to one, otherwise it is set to zero. SILTP uses two bits for encoding. The pixel encoding is defined as:

$$s = \begin{cases} 01, I_n > (1+\alpha) I_c \\ 10, I_n < (1-\alpha) I_c \\ 00, otherwise \end{cases}, \quad (2)$$

where I_n is a neighboring pixel, α is scale factor indicating comparing range, and I_c is central pixel (here α=0.3). Comparison of LBP and SILTP is shown in Fig. 4. Both LBP and SILTP are robust to scale invariance of pixel values. Unlike LBP, SILTP is also robust against noise.

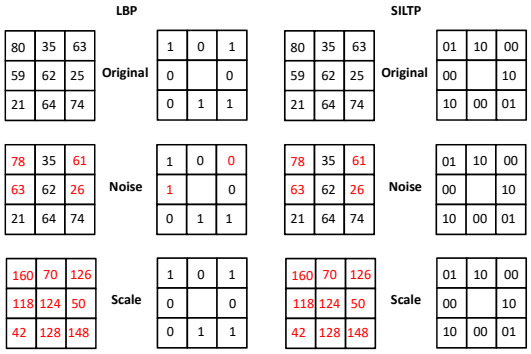


Fig.4. LBP and SILTP comparison

Here, each image is divided into blocks of size 10x10 pixels. These column-wise blocks are represented as patches, and SILTP patches are shown in Fig.3. For every block histogram is computed. As in the color based approach, downsampling is performed. This process repeats for the number of scales and the number of values set for radius parameter in SILTP. This is how texture descriptors are obtained, which makes the texture feature vector.

Color based feature vector is calculated using ternary encoding (1). Also, texture based feature vector is calculated using ternary encoding (2). The local maximal occurrence (LOMO) feature vector for number of scales is formed by concatenating color and texture vectors into one vector. The LOMO is color and texture based feature vector.

III. SIMULATION

In this paper for the purpose of testing we examined multi-camera view dataset [8]. The dataset is consisted of frames from six camera views (labeled as camera views 0-5), and annotations. Six calibrated digital video cameras are used with 25 fps covering an area of 22mx22m. The frames are of 360x288 pixels, with available bonding boxes for different types of objects, like persons or pedestrians, cars and buses. The six views of a scene at two successive frames are represented in Fig.5 (frame no. 124) and Fig.6 (frame no. 125) as two examples, with a pedestrian denoted with green boxes and cars denoted with yellow boxes. Examples of pedestrians are shown in Fig.7.

Fig. 5. Six camera views of a scene at a certain moment - example 1 (frame no. 124; green box - pedestrian, yellow box -car).

Fig. 6. Six camera views of a scene at a certain moment - example 2 (frame no. 125; green box - pedestrian, yellow box -car).

Fig. 7. Illustration of different tested pedestrians from the dataset, where two camera views are presented for each person.

In this paper only pedestrians have been considered [9]. Since the OOI (Object Of Interest) is moving, the corresponding boxes within a camera view are of different sizes, as shown in Fig.8, for three successive frames (noted here as i, i+1 and i+1). Another common issue that appears here is the occlusion as one can notice. In order to compare images, they are usually resized to a fixed resolution [2]. Nevertheless, the size may affect the results. Here two techniques are compared, where resizing the object is performed according to maximum (max) and minimum (min) resolution of bounding box of the same object across the frames for the identification task. Resizing for the object from Fig.8 is presented in Fig.9.

Fig. 8. Original box sizes for the man with a bicycle for three successive frames (camera view 0).

Fig. 9. Camera view 0 and the man with a bicycle –resized according to (a) min and (b) max technique.

A person can be re-identified across the frames if adequate clustering exists. Moreover, here the clustering is analyzed using six camera views due to possible unavailability of one of the cameras. Images are resized according to maximum and minimum number of pixels for both axes (max and min resizing). The color, texture and combined color and texture based feature vectors are computed. The feature vectors are calculated for three scales.

For the purpose of experiment three metrics are applied to measure distance between the color, texture and combined color and texture (LOMO) based feature vectors. The three distance metrics are calculated as:

$$d_{Euclidean}(x,y)=\left(\sum_{i=1}^{m}(x_i-y_i)^2\right)^{1/2}, \quad (3)$$

$$d_{City-block}(x,y)=\sum_{i=1}^{m}|x_i-y_i|, \quad (4)$$

$$d_{Minkowski}(x,y)=\left(\sum_{i=1}^{m}|x_i-y_i|^r\right)^{1/r}, \quad (5)$$

representing Euclidean, City-block (Manhattan) and Minkowski metric, respectively, where x and y are the feature vectors, and where m represents the number of descriptors (in (5) r=3 is selected).

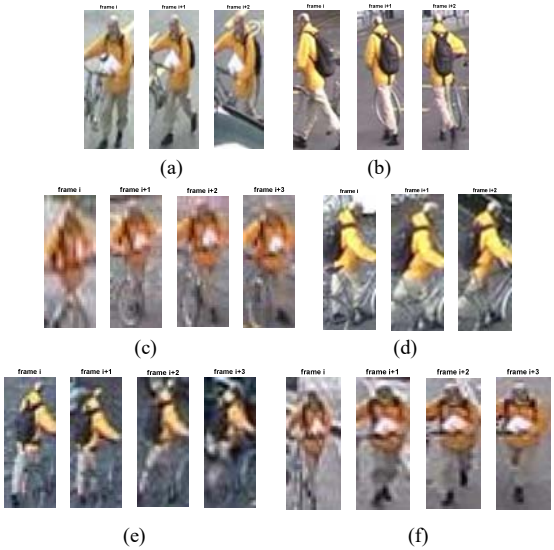
Distance is measured from the first image in camera view 0, and normalized by feature maximum value. After computing distances threshold is selected as mean value of distance metrics calculated for color, texture and LOMO feature vectors. The threshold is used for differentiating between the groups of multi-view camera images/frames belonging to different persons. Error is calculated as a number of misclassified images, divided by the total number of images, where the result is expressed in percentages.

IV. Experimental results

Experimental results are presented for the cases where obvious differentiation are expected. For the larger resolution (39x121 pixels) total of 40 images were used and for smaller resolution (14x33 pixels) there were 37 images. These are the images from six camera views which represent different groups: person 1 (lady in white) and person 2 (man with bicycle), shown in Fig.10 and Fig.11, respectively.

(a) (b) (c)

Fig. 10. Lady in white - resized using max technique: (a) Camera view 0 (4 frames), (b) Camera view 1 (4 frames), (c) Camera view 5 (3 frames).



(a) (b) (c) (d) (e) (f)

Fig. 11. Man with the bicycle - resized according to max technique for: (a) Camera view 0, (b) Camera view 1, (c) Camera view 2, (d) Camera view 3, (e) Camera view 4, (f) Camera view 5.

Color seems to be an obvious solution in most of re-identification and tracking algorithms, but the question is how the results can vary in such cases after resizing. In Fig.12 Euclidean distances are presented for color versus texture case, as well as for combined color and texture (LOMO) feature vector after max resizing. Two groups (colored as blue and red in Fig.12) represent distances for person 1 and person 2, respectively. Each point within the groups is denoted as camA_B, where A is a camera view and B is an image number from that camera view.

If min approach is applied in order to resize images, as in Fig.13, the Euclidean distance gives different results for max approach. Here, color is not as discriminative as texture.

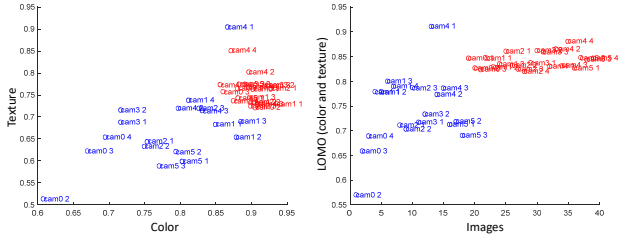


Fig. 12. Color versus texture feature vector distances (left) and LOMO feature vector distances for two groups (right) - Euclidean distance, max resize.

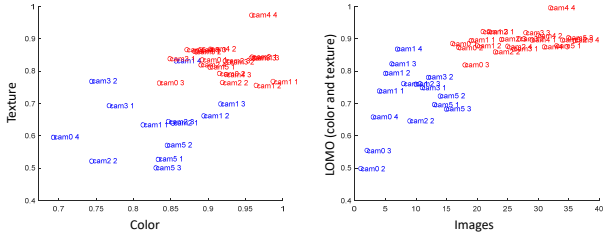


Fig. 13. Color versus texture feature vector distances (left) and LOMO feature vector distances for two groups (right) - Euclidean distance, min resize.

The color versus texture, and the color and texture representations are given for City-blok metric in Fig.14 and Fig.15. The representatations for Minkowski distance are shown in Fig.16 and Fig.17.

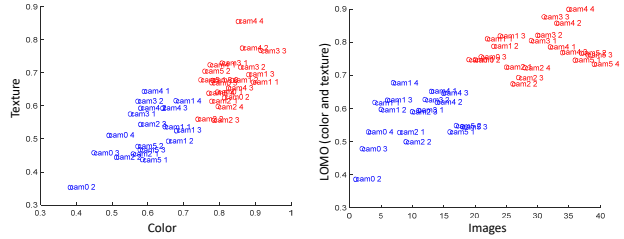


Fig. 14. Color versus texture feature vector distances (left) and LOMO feature vector distances for two groups (right) - City-block distance, max resize.

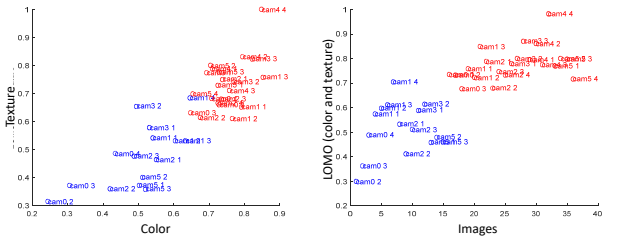


Fig. 15. Color versus texture feature vector distances (left) and LOMO feature vector distances for two groups (right) - City-block distance, min resize.

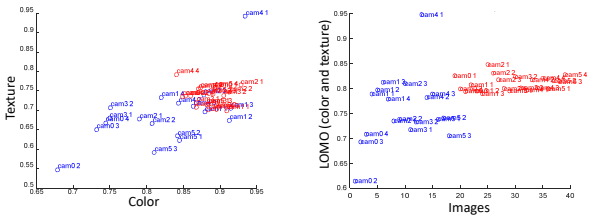


Fig. 16. Color versus texture feature vector distances (left) and LOMO feature vector distances for two groups (right) - Minkowski distance, max resize.

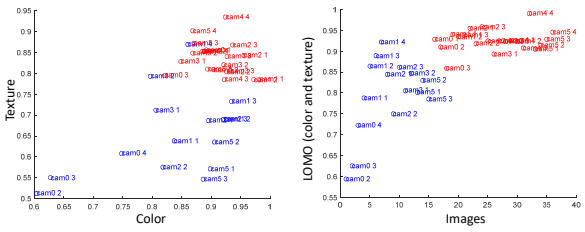


Fig. 17. Color versus texture feature vector distances (left) and LOMO feature vector distances for two groups (right) - Minkowski distance, min resize.

The percentage of misclassified images is calculated and presented in Table II, for each distance metric and resize technique.

TABLE II
PERCENTAGE OF MISCLASSIFIED IMAGES

Distance metric (resize technique)\ Feature vector (F.v.)	LOMO (Color+Texture) F.v.	Color F.v.	Texture F.v.
Euclidean (max)	**2.5%**	7.5%	10%
Euclidean (min)	5.4%	10.8%	5.4%
City-block (max)	12.5%	**0%**	15%
City-block (min)	8%	**2.7%**	5.4%
Minkowski (max)	20%	20%	22.5%
Minkowski (min)	16.21%	29.7%	5.4%

The values which represent the cases that showed the lowest misclassification percentages are bolded. One may see that for min resize approach in most cases texture feature gave better results than color and LOMO feature. Exception is City-block metric where color feature gave better results for both resize techniques. For max resize technique, LOMO and color feature show better results than texture feature. If LOMO feature vectors are chosen, Euclidean distance metric is the best solution. Nevertheless, color based differentiation and city-block distance may give even better results. Similar conclusions are drawn when testing other persons within the dataset. For the case of comparison illustrated in Fig. 2 the results are presented in Fig.18, showing the best solution (the lowest metric value) for color based feature vector and city-block distance. The results seem not be affected to a great extent for different resizing techniques in the case of color and city-block distance.

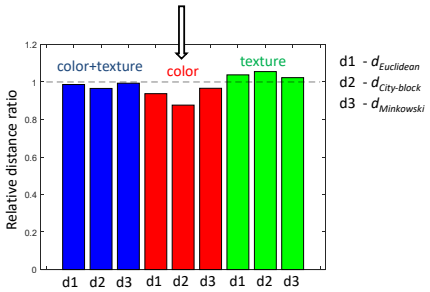


Fig. 18. Comparison results between the distances for max resize in the case presented in Fig.2.

V. CONCLUSION

In this paper we performed experimental analysis based on multi-camera view, and using different resize approaches and three distance metrics (Euclidean, City-block and

Minkowski). Distance is measured between state-of-the-art ternary encoded features like: HSV color, SILTP texture and combined LOMO features. Here, our focus was on the cases where color should enable obvious differentiation. The lowest percentage error is showed for City-block distance metric and color feature vector, as well as max resizing. Euclidean distance is the right choice when it comes to LOMO feature vector. From the obtained results one may conclude that the choice of resizing technique and distance metric affects the differentiation between the multi-camera view sets belonging to different persons even in the cases where color is expected to solve the re-identification task.

Future work will be oriented towards further experiments with distance metrics in re-identification tasks, especially the ones where color feature is not crucial for person re-identification.

ACKNOWLEDGMENT

This research is supported and funded by Serbian Ministry of Education, Science and Technological Development.

REFERENCES

[1] Y.J. Cho, S.A. Kim, J.H. Park, K. Lee, and K.J. Yoon, "Joint person re-identification and camera network topology inference in multiple cameras," *Computer Vision and Image Understanding*, *180*, pp. 34-46, 2019.

[2] F. Yang, et al., "Attention driven person re-identification," *Pattern Recognition*, 86, pp. 143-155, 2019.

[3] L. Shengcai, Y. Hu, X. Zhu, and S.Z. Li, "Person re-identification by local maximal occurrence representation and metric learning," In *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 2197-2206, 2015.

[4] W. Li, R. Zhao, and X. Wang. "Human reidentification with transferred metric learning," *Asian conference on computer vision*, Springer, Berlin, Heidelberg, pp. 1-14, 2012.

[5] H. Wu, M. Xin, W. Fang, H.M. Hu, and Z. Hu, "Multi-level feature network with multi-loss for person re-identification," *IEEE Access*, *7*, pp.91052-91062, 2019.

[6] Z. Zhang, H. Zhang, and S. Liu, "Coarse-fine convolutional neural network for person re-identification in camera sensor networks," *IEEE Access*, *7*, pp.65186-65194, 2019.

[7] G. Zhao and M. Pietikainen, "Local binary pattern descriptors for dynamic texture recognition," In *18th International Conference on Pattern Recognition (ICPR'06)*, IEEE; Vol. 2, pp. 211-214, 2006.

[8] G. Roig, Gemma, X. Boix, H.B. Shitrit, and P. Fua, "Conditional random fields for multi-camera object detection," In *Proc. of the Int. Conf. on Computer Vision*, IEEE, pp. 563-570, 2011.

[9] R. Zhu, J. Fang, S. Li, Q. Wang, H. Xu, J. Xue, and H. Yu, "Vehicle re-identification in tunnel scenes via synergistically cascade forests," *Neurocomputing*, *381*, pp.227-239, 2020.

Constant quality mode 4k video comparison using AV1 reference tool

Milan Milivojevic, Dragi Dujkovic, and Ana Gavrovska, *Member, IEEE*

***Abstract*— A variety of new coding tools has been developed and are expected to be developed in the near future in order to achieve high-quality video transmission. One of the next-generation video encoding formats is AV1, as the first compression format developed by the Alliance for Open Media, where AV1 is recognized as VP9 successor. In this paper 4k low frame rate video sequences are compared for different constant quality (CQ) values using reference tool libaom. The obtained results are also compared to VP9 and HEVC codecs. Slow AV1 coding is performed using libaom, in order to analyze differences between different CQ settings. The results show compression performance using 2-norm evaluation and time needed for coding.**

***Index Terms*— Video coding, 4k video, constant quality, AV1, libaom, VP9, HEVC.**

I. Introduction

Novel video technologies and standards are inevitable nowadays. The known fact is that most of the telecommunication traffic is related to video. Namely, it is estimated that by 2022, approximately about 82% of global internet will be dedicated to video content [1]. So, the general focus is on video delivery, such as in the case of OTT (Over-the-Top) streaming, wired or wireless communication or TV broadcasting. Particularly important are the coding formats for internet applications and efficient delivery over internet. It is obvious that OTT providers (Netflix, Hulu, etc.) and other immersive media content based industries are interested in developing innovative solutions in video encoding and compression.

In [2] three main video technology trends in 2020 are pointed out. The first one is to increase expectations of viewers by providing higher QoE (Quality of Experience), besides QoS (Quality of Service). Thus, higher quality video should be delivered to consumers, but this should be done in efficient manner. Consequently, the fast delivery of the solutions is the second trend. Moreover, the trend for media content is to find its way quickly to the market, and this should make both services and producing assets more efficient. Finally, the third trend is to improve the return of investments by advanced controlling and managing costs, risks and application complexity.

Even though, H.264 is an older video format from about 2004, it is still in massive usage despite the fact there is more recent one. The recent one is H.265, or HEVC (High Efficiency Video Coding), from about 2013, which gives better performance [3]. Besides HEVC, another UHD (Ultra High Definition) codec has become popular over the years. Open and royalty-free Google's solution like VP9 was widely adopted by different platforms, such as Youtube. Nowadays, AV1 is considered as VP9 successor, and a new open royalty-free format for video coding developed by the AOMedia (Alliance for Open Media) [4-6]. Its first release was in 2018, and, since then, different solutions has been designed for video transmission and video services over internet.

In recent work [7] 4k video traffic has been analyzed using different prediction models, where the sequences were encoded with H.265/HEVC, whereas in [8] variability of the traffic was analyzed. In this paper, analysis of the 4K video traffic is performed using 4k AV1 based on AV1 reference software called libaom. The results are compared to VP9 and HEVC for 4k video from CQ (constant quality or CRF – constant rate factor), coding time and 2-norm traffic standpoint.

The paper is organized as follows. After the introduction, AV1 format is briefly explained in Section II. The simulation and materials used in this paper are explained in Section III. Then, in Section IV the experimental results followed by discussion are given in order to evaluate the compression performance. Finally, in Section V conclusions and future work are mentioned.

II. AV1 video format

From AV1 (AOMedia Video 1), much is expected [3-6]. There is a need for efficient compression standards, and future implementations should balance software and hardware possibilities. Firstly, software experimental analysis is of great importance for realizing complex schemes and architectures. In the case of AV1, there are open implementations. It is royalty-free solution for video coding. The general AV1 architecture is illustrate in Fig.1.

As in every new generation of video coding format there are differences in coding structure, and the gains are obtained using different tools [4]. Motion vector prediction is improved in spatio-temporal domain, where eight main intra prediction directional modes are used. Also, superblocks are introduced.

Milan Milivojevic is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mail: milansmilivojevic@gmail.com).

Dragi Dujkovic is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mail: dragi@etf.rs).

Ana Gavrovska is with the School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11020 Belgrade, Serbia (e-mails: anaga777@gmail.com; anaga777@etf.rs).

So, one may use blocks of 128x128 pixels and 64x64 pixels. Higher precision is expected (ten or twelve bits), besides eight bit depth. For transform domain rectangular DCT (Discrete Cosine Transform) and asymmetric DST (Discrete Sine Transform) are used in AV1, while new quantization parameters and filtering techniques are adopted [4-6].

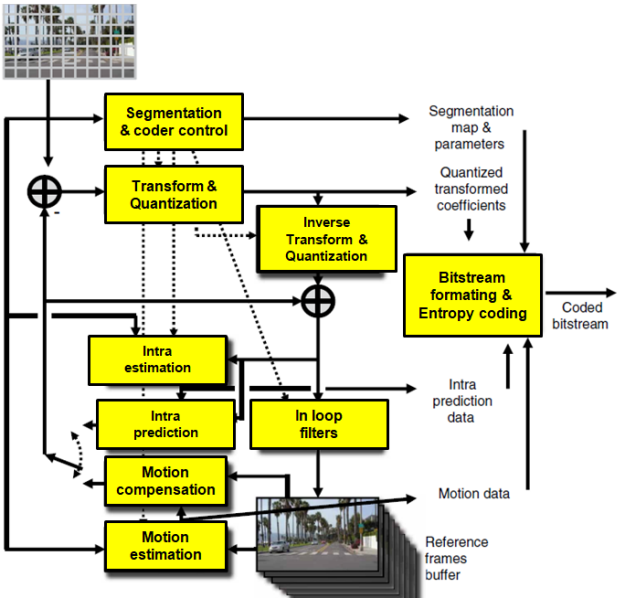


Fig.1. Illustration of general AV1 architecture [4].

Recursive partitioning of superblocks of 128x128 pixels introduced in AV1 is illustrated in Fig.2. Handling hierarchical and recursive techniques and much more in AV1 format led to developing different versions optimized for various purposes. Libaom was introduced as reference AV1 codec (coder-encoder), and enabled the main insight into the AV1 possibilities. It made progress in fast delivering new solution to the public.

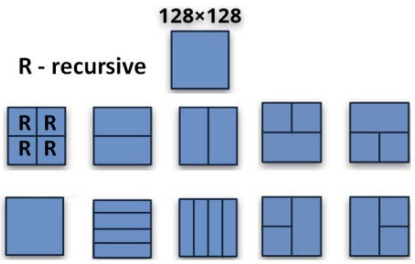


Fig.2. Recursive partitioning of superblock of 128x128 pixels introduced in AV1.

There are already different solutions of AV1 available for specific aims besides the reference software. Typical example is another AV1 codec dav1d for low CPU (Central Processing Unit) processing. Others variations are also available (rav1e, svt-av1). The idea is to develop solutions for video coding for optimized high-performance tasks [5].

In [5] it is claimed that AV1-libaom performs better than HEVC. Even about 43% improvement is reported from PSNR (Peak Signal-to-Noise ratio). AV1 is compared to HEVC since its general architecture looks alike.

III. Simulation

Sequences in mp4 format of 4k resolution are used for the experiment. The illustration of the tested content is shown in Fig. 3. Details about the 4k data are given in Table I. The sequences are of lower frame rate (LFR), so even higher length than common ITU 10 seconds can be used to observe the packet variability (here thirty seconds). Having in mind the spatial resolution like 4k (DCI – Digital Cinema Initiatives or UHD – Ultra High Definition), as well as time resolution, i.e. frame rate, the time needed for testing should be reasonable, and the length is often decreased by video trimming (e.g. to five seconds in [3]). Moreover, one should have in mind higher bit depth based content expected to be a part of future common traffic. Initial attempts when choosing relatively reasonable sequence length showed very slow procedure of applying the reference tool (libaom) in the LFR 8 bit cases, which are considered in this paper.

Materials are prepared according to Constant Quality (CQ) or crf factor. For each sequence four CQ/crf values are used: CQ-20, CQ-24, CQ-30, CQ-34. It was not possible to apply CQ-10 or CQ-40 in the experiment.

The analysis is performed on 64bit Windows 10 Pro Intel(R) Core(TM) i5-8500 CPU, 3GHz, 8GB using ffmpeg 4.3.1 [9].

TABLE I
4K DATA DETAILS

Source (abbrevation)	Resolution	Frame rate
Big Buck Bunny (bbb)	3840 x 2160	30 fps
Tears of Steel (tos)	3840 x 1714	24 fps

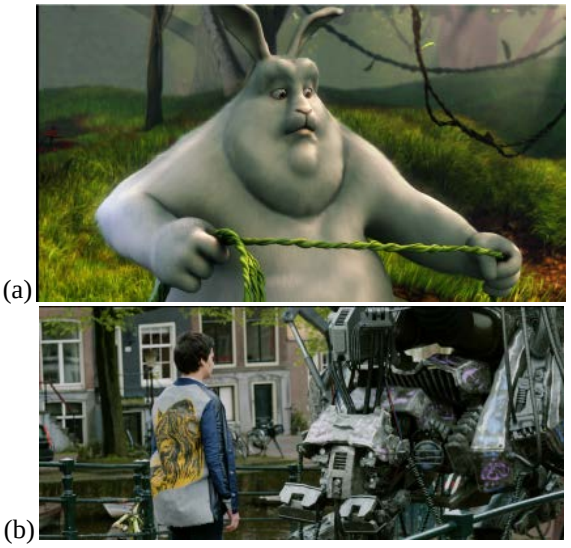


Fig. 3. Video sequences: (a) Big Buck Bunny (bbb file), and (b) Tears of Steel (tos file).

By using two files eight xml sequences are obtained (four xml sequences are generated for each crf value). In the first part of the analysis, time is measured in minutes while coding due to slow libaom performance. In the second phase, each

sequence is represented by the vector magnitude, regardless of the frame type. The magnitude or 2-norm (Euclidean norm) is calculated as:

$$norm(x) = \|x\| = \left(\sum_k |x_k|^2 \right)^{1/2}, \quad (1)$$

where x is an array corresponding to obtained sequence of frames, and k is frame index. The relative ratio for time, $t(x)$, and 2-norm values, $norm(x)$, are calculated for the comparison reasons. The p ratios are defined as:

$$p^{time}_{crf(N)} = t(x)_{crf(N)} / t(x)_{crf(N_{ref})}, \quad (2)$$

$$p^{norm}_{crf(N)} = norm(x)_{crf(N)} / norm(x)_{crf(N_{ref})}, \quad (3)$$

Where N is current and N_{ref} is reference crf value (here N_{ref} = 20). The above procedure is repeated for VP9 and HEVC video format.

IV. Experimental Results

In Table II the results of using AV1-libaom are presented. Namely, time needed for slow coding is presented for each CQ/crf value. Time spent for coding using standard VP9 and HEVC codecs, are presented in Table III and Table IV, respectively.

TABLE II
TIME FOR CODING TO AV1-LIBAOM FORMAT

No.	1	2	3	4
Coding time [min]	CQ -20	CQ -24	CQ -30	CQ -34
bbb	276	260	249	214
tos	510	417	266	196

TABLE III
TIME FOR CODING TO VP9 FORMAT

No.	1	2	3	4
Coding time [min]	CQ -20	CQ -24	CQ -30	CQ -34
bbb	6	6	5	5
tos	12	10	7	6

TABLE IV
TIME FOR CODING TO HEVC FORMAT

No.	1	2	3	4
Coding time [min]	CQ -20	CQ -24	CQ -30	CQ -34
bbb	10	9	8	7
tos	13	8	7	5

For the comparison, relative coding time is calculated and presented in Fig.4 and Fig.5 for bbb and tos sequences, respectively. The obtained sequences for AV1-libaom coding are presented in Fig.6 and Fig.7, for bbb and tos, respectively.

The new codec requires long time to perform coding. Testing shows that relative time difference for higher quality (between crf20 and crf24) resembles VP9 standard for both bbb and tos sequences. On the other hand, relative time difference needed for lower coding quality between crf30 and crf34 seems similar. Around 10% is the difference between the two highest crf values.

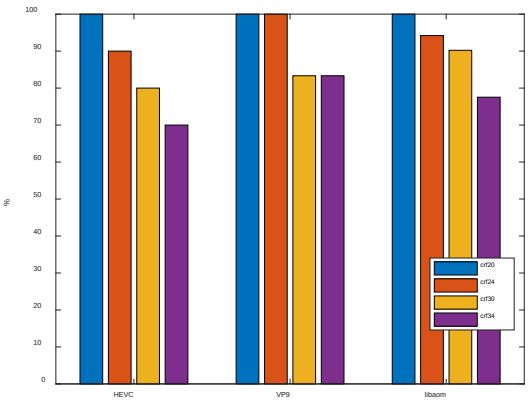


Fig. 4. Relative coding time for bbb sequences.

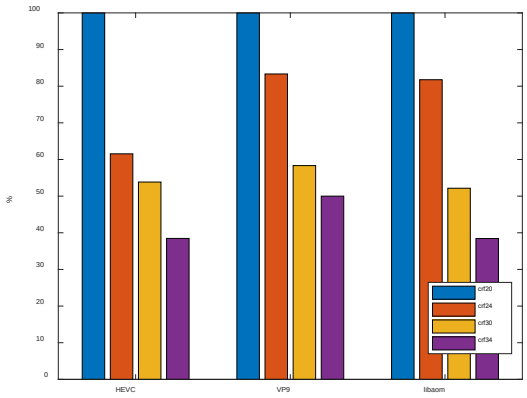


Fig. 5. Relative coding time for tos sequences.

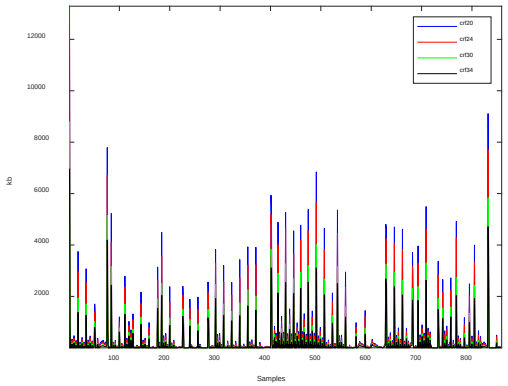


Fig. 6. Obtained sequences using reference AV1-libaom for bbb video.

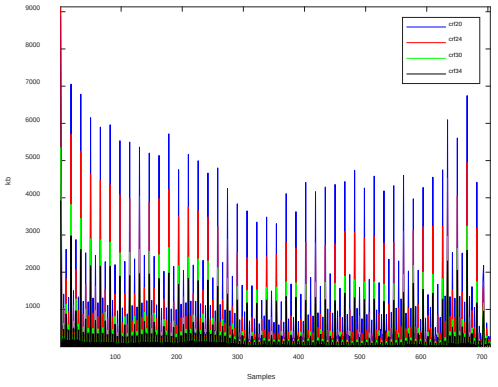


Fig. 7. Obtained sequences using reference AV1-libaom tool for tos video.

It is well known that AV1 generally suppresses VP9 by about 30% [3-6]. In Fig.8 and Fig.9 relative 2-norm values are shown for bbb and tos sequences, respectively. Lower values for HEVC are expected since for HEVC bidirectional frames exist. In Fig.10 obtained predicted frames are shown for libaom and VP9 in the case where crf equals 24.

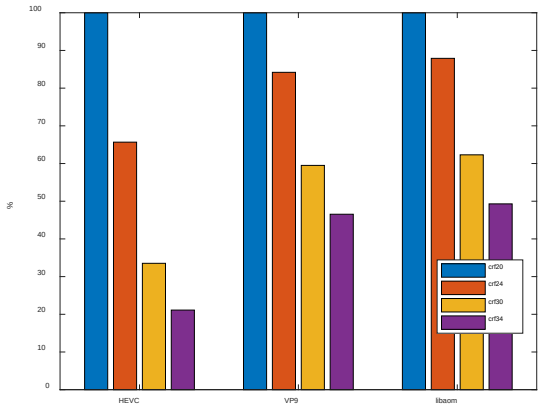


Fig. 8. Relative 2-norm values for bbb sequences.

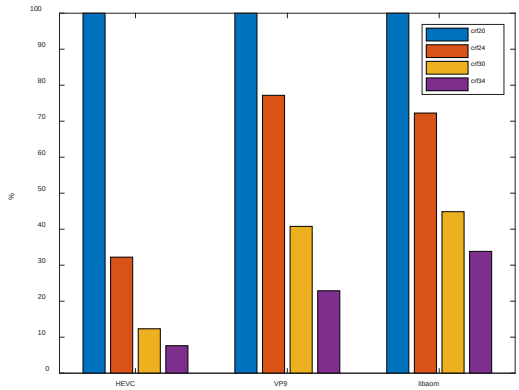


Fig. 9. Relative 2-norm values for tos sequences.

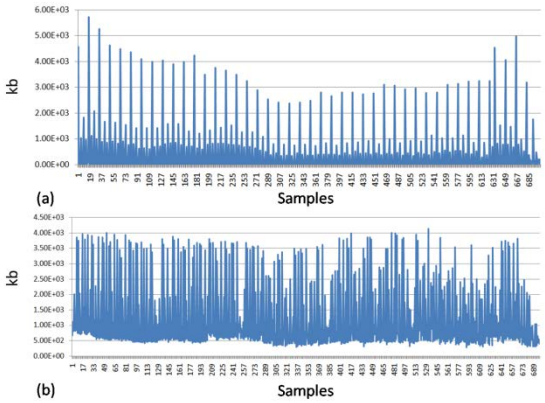


Fig. 10. (a) AV1-libaom and (b) VP9 predicted frames for tos video and crf24.

In this relative estimation it can be seen that the magnitude is similar between VP9 and its successor. As for av1-libaom, smaller difference is obtained for lower quality, i.e. between crf30 and crf34, compared to VP9. This can particularly be observed for tos video in Fig. 9.

In Fig.10(a) it is shown for libaom that frames can be easily differentiated from the size standpoint. In ther words, there are relatively small and relatively high sample values presented. The naturaleness is more visible for VP9 in Fig.10(b), where such differentiation is not obvious. This shows higher control of the traffic in the case of AV1-libaom. The traffic behaviour for different video quality is changed in this way.

V. Conclusion

In this paper low frame rate experiment with constant quality AV1-libaom codec is performed. The obtained results show slow coding performance of the reference software, as well as similar relative magnitude to VP9. Nevertheless, the traffic using new codec shows different behavior in comparison to VP9.

In future work further experiments should be performed in order to evaluate the performance of AV1 standard and its specific implementations, as well as to analyze the behavior in high frame rate and high dynamic range cases. Moreover, other formats are expected to have a great impact on the market, and are expected to be a part of future experiments.

Acknowledgment

This research is supported and funded by Serbian Ministry of Education, Science and Technological Development.

References

[1] CISCO, “CISCO visual networking index: forecast and methodology, 2017–2022,” November 2018.

[2] J. Shulman, “Bitmovin’s video top Video Technologz Trends 2020”, Bitmovin, March 2020. https://bitmovin.com/video-technology-trends/ (last accessed 15.07.2020.)

[3] F. Zhang, A.V. Katsenou, M. Afonso, G. Dimitrov, and D.R. Bull, “Comparing VVC, HEVC and AV1 using Objective and Subjective Assessments,” *arXiv preprint arXiv:2003.10282*, 2020. https://arxiv.org/pdf/2003.10282.pdf (last accessed 15.07.2020.)

[4] I. Trow, “AV1: Implementation, Performance, and Application,” *SMPTE Motion Imaging Journal*, *129*(1), pp.51-56. 2020.

[5] Y. Chen, D. Mukherjee, J. Han, A. Grange, Y. Xu, S. Parker, C. Chen, H. Su, U. Joshi, C.H. Chiang, and Y. Wang, “An Overview of Coding Tools in AV1: the First Video Codec from the Alliance for Open Media,” *APSIPA Transactions on Signal and Information Processing*, Vol. 9, 2020.

[6] Alliance for Open Media – An Alliance of Global Media Innovators, http://aomedia.org/ (last accessed 15.07.2020.)

[7] D. R. Marković, A. M. Gavrovska, I. S. Reljin, “4K Video Traffic Prediction using Seasonal Autoregressive Modeling“, *Telfor Journal*, Telecommunications Society, Belgrade, Vol. 9, No. 1, pp. 8-13, 2017.

[8] A. M. Gavrovska, M. S. Milivojevic, G. Zajic and I. S. Reljin, “Video traffic variability in H.265/HEVC video encoded sequences,“ *13th Symposium on Neural Network Applications in Electrical Engineering* (NEUREL), 22-24. November, Belgrade, Serbia, pp. 109-112, 2016.

[9] Ffmpeg, https://ffmpeg.org/ (last accessed 12.08.2020.)

Real-Time Moving Object of Interest Detection in Multi-Sensor Imaging System

Miloš Pavlović, *Member, IEEE,* Nikola Stojiljković, Ivan Gluvačević,
Miljan Vučetić, *Member, IEEE,* Miroslav Perić, *Member, IEEE*

***Abstract*— The video surveillance systems usually require a constant user's observation of the scene in order to respond on time to situations that could be risky. In order to allow the system to understand and react to the environment and in that way to reduce user's need for manual intervention, objects of interest detection algorithms need to be integrated into the video systems. This paper proposes a real-time algorithm for moving object of interest detection, applicable to the multi-sensor imaging system. Algorithm introduces a combination of the method for motion detection in images - Optical flow and deep learning method for detecting objects of interest - YOLO algorithm. The objects of interest for detection in this paper are pedestrians and cars.**

***Index Terms*—Motion detection, Object detection, Real-Time, Multi-Sensor imaging**

I. Introduction

The video surveillance systems usually require a constant presence of user and observation of the scene in order to respond on time to situations that could be risky. In other words, a common problem is the lack of "intelligence" to function independently without the need of user's manual intervention.

Object detection and tracking algorithms are also an important part of the video surveillance systems. Most often, tracking algorithms implemented in video systems are algorithms from the class of semi-automated algorithms. This means that the user needs to manually select the target to be tracked, and then the algorithm takes over the tracking process. In the case when objects that are in motion need to be selected (especially when they are moving fast), the problem of manual selection becomes especially pronounced. For this reason, there is a need for algorithms for fully automated detection of moving objects.

The detection of moving objects in video as a problem of computer vision from early beginnings has been approached in various ways, which often depended on the computer processing power, but also very often of the problem understanding. The rapid improvements in video sensor quality and resolution and the dramatic increase in processing power of platforms running the algorithms in the previous period have favored the development of new algorithms and applications in this segment of computer vision.

The primary goal of a moving object detection algorithm is to achieve high detection accuracy in different conditions in the scene, as well as sufficient execution speed for real-time operation. In addition to work on visible light (color) images, in order to provide the ability to work in different weather and low light conditions, and also to work in complete darkness, using sensors that work in infrared (IR) range, the aim is development of algorithm for detection in Short-Wave IR (SWIR - 1.4-3μm), Medium-Wave IR (MWIR - 3-8μm) and Long-Wave IR (LWIR - 8-14μm) [1] imaging as well.

IR imaging sensors have the grayscale output and objects on these type of images have very specific features compared to the features of same type of objects in color images. Most of publicly available datasets for training neural networks for object classification and detection contain only color images, so neural networks trained only on color images are not able to detect objects in infrared images with high accuracy. On the other hand, color images are very sensitive to many types of visual obstacles (low light conditions, night conditions, haze problems, various atmospheric disturbances, etc.), especially in detection various objects at very large distances, where the influence of these various disturbances is very high, and all these problems prevent algorithm from reaching its maximum detection accuracy. As the task is to develop an algorithm for moving objects of interest detection for real-time application, the presented algorithm in this paper is a combination of conventional method for motion detection in images - Optical flow [2] and deep learning method for objects of interest detection - YOLO algorithm [3]. The objects of interest for detection in this paper are pedestrians and cars.

The paper is organized as follows. Section II describes the methodology of work, method for the motion detection - Optical flow, as well as the objects of interest detection method based on deep learning. Section III gives a description of the implemented system for the moving object of interest detection. Section IV describes the experimental work including the experimental setup, results and discussion. Section V lists conclusions and indicates directions for future work in this research area.

Miloš Pavlović is with Vlatacom Institute of High Technologies, Blvd. Milutina Milankovića 5, 11070 Belgrade, Serbia (e-mail: milos.pavlovic@vlatacom.com).

Nikola Stojiljković is with Vlatacom Institute of High Technologies, Blvd. Milutina Milankovića 5, 11070 Belgrade, Serbia (e-mail: nikola.stojiljkovic@vlatacom.com).

Ivan Gluvačević is with Vlatacom Institute of High Technologies, Blvd. Milutina Milankovića 5, 11070 Belgrade, Serbia (e-mail: ivan.gluvacevic@vlatacom.com).

Dr Miljan Vučetić is with Vlatacom Institute of High Technologies, Blvd. Milutina Milankovića 5, 11070 Belgrade, Serbia (e-mail: miljan.vucetic@vlatacom.com).

Dr Miroslav Perić is with Vlatacom Institute of High Technologies, Blvd. Milutina Milankovića 5, 11070 Belgrade, Serbia (e-mail: miroslav.peric@vlatacom.com).

II. METHODOLOGY

A. Optical flow

Optical flow [2] represents the movement of objects between consecutive frames in a video sequence, caused by the relative movement of the object and the camera. The problem can be expressed as illustrated in Fig. 1.

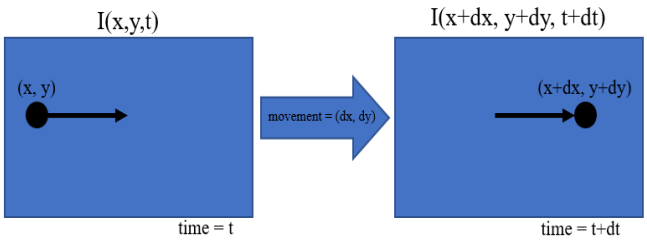


Fig. 1 Optical flow illustration

As shown in Fig. 1, between consecutive frames it is possible to express the intensity of image I as a function of the spatial coordinates (*x*, *y*) and time (t). In other words, if the pixels of image I(x, y, t) from the video sequence at time *t* are moved by (*dx*, *dy*) over time *dt*, result is the new image I(x + dx, y + dy, t + dt).

Several assumptions are made for the computation of Optical flow.

1. An object's pixel intensities are constant between consecutive frames.

$$I(x, y, t) = I(x + dx, y + dy, t + dt) \tag{1}$$

2. With Taylor Series Approximation, we have:

$$I(x + dx, y + dy, t + dt) = \\ = I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t + \cdots \\ \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 \tag{2}$$

3. By dividing the previous equation by δt, the equation of Optical flow is:

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0 \tag{3}$$

where: $u = \frac{\partial x}{\partial t}$ and $v = \frac{\partial y}{\partial t}$

$\frac{\partial I}{\partial x}$, $\frac{\partial I}{\partial y}$, and $\frac{\partial I}{\partial t}$ present the image gradients along the horizontal and vertical axes and time. To solve the Optical flow problem, a solution of u ($\frac{\partial x}{\partial t}$) and v ($\frac{\partial y}{\partial t}$) are required to determine motion over time. It can be noted that the Optical flow equation for u and v cannot be solved directly, since there is only one equation for two unknown variables.

There are two different approaches to solving Optical flow problems: Sparse [4] and Dense [5]. Sparse gives the flow vectors of characteristic features in the image, while Dense Optical flow gives the flow vectors of all pixels in the image up to one flow vector per pixel. Dense Optical flow has better accuracy in motion detection but is more computationally complex [6].

B. Object Detection in Image

In recent years, much has been done in the development of object detection algorithms using a standard camera without additional sensors. State-of-the-art object detection algorithms use deep neural networks.

YOLO (YOU ONLY LOOK ONCE) is an object detection algorithm published in 2016 [3]. The algorithm was developed to perform the process involving detection and classification in one step. The idea behind the YOLO algorithm is different from other detection methods (RCNN, Fast RCNN, and Faster RCNN) in that the bounding boxes and the classes of the detected objects are determined after only one evaluation of the input image.

First, the input image is divided into a grid of S × S cells. Further, for each grid cell, B bounding boxes are defined together with a confidence score. The Confidence score is defined as:

$$\mathrm{C} = P_r(Object) * \mathrm{IOU}_{\mathrm{pred}}^{\mathrm{truth}} \tag{4}$$

In this case, C is the probability that an object exists in each bounding box, while the IOU represents the Intersection Over Union ratio and takes the value from the range [0, 1]. The intersection is the area of overlap between the predicted bounding box and the ground truth bounding box, while union represents the total area of the predicted bounding box and the ground truth bounding box. The IOU value closer to 1 indicating that the predicted bounding box is closer to the ground truth bounding box. The illustration of the IOU metrics is as shown in Fig. 2.

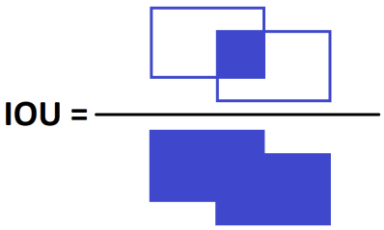


Fig. 2 Illustration of the IOU metrics

Simultaneously with the construction of bounding boxes, each grid cell also predicts a conditional probability of class. The class-specific probability [3] for each grid cell is defined as:

$$P_r(Class_i|Object) * P_r(Object) * \mathrm{IOU}_{\mathrm{pred}}^{\mathrm{truth}} \\ = P_r(Class_i) * \mathrm{IOU}_{\mathrm{pred}}^{\mathrm{truth}} \tag{5}$$

The following equation is used for loss calculation and ultimately confidence optimization:

$$Loss = \sum_{i=0}^{s^3}\sum_{j=0}^{A} 1_{ij}^{obj}\left[\left(b_{x_i} - b_{\hat{x}_i}\right)^2 + \left(b_{y_i} - b_{\hat{y}_i}\right)^2\right]$$

$$+\alpha_{coord}\sum_{i=0}^{s^3}\sum_{j=0}^{A} 1_{ij}^{obj}\left[\left(\sqrt{b_{w_i}} - \sqrt{b_{\hat{w}_i}}\right)^2 + \left(\sqrt{b_{h_i}} - \sqrt{b_{\hat{h}_i}}\right)^2\right]$$

$$+\sum_{i=0}^{s^3}\sum_{j=0}^{A} 1_{ij}^{obj}(C_i - \hat{C}_i)^2 \qquad (6)$$

$$+\alpha_{noobj}\sum_{i=0}^{s^3}\sum_{j=0}^{A} 1_{ij}^{obj}(C_i - \hat{C}_i)^2$$

$$+\sum_{i=0}^{s^3} 1_{i}^{obj}\sum_{c\in classes}(p_i(c) - \hat{p}_i(c))^2$$

For each prediction, the position of the center and the size of the bounding box are corrected using the given loss function. In loss function, parameter A is the number of bounding boxes for each grid cell of S × S image grid. The variables b_x and b_y present the center of each bounding box prediction, while b_w and b_h refer to the bounding box width and height. The importance of boxes with and without objects is controlled with the variables α_{coord} and α_{noobj}. C presents confidence, while *p(c)* refers to classification prediction. 1_{ij}^{obj} equals to 1 presents the responsibility for predicting the object of the j^{th} bounding box in the i^{th} grid cell and is equal to 0 otherwise. If the object is in the cell *i*, 1_{i}^{obj} is 1, 0 otherwise.

While the loss is used to evaluate model performance, the accuracy of model predictions in detecting objects is calculated by the equation of average precision, where P(k) refers to precision at threshold *k*, while $\Delta r(k)$ refers to recall change.

$$argPrecision = \sum_{k=1}^{n} P(k)\Delta r(k) \qquad (7)$$

The basic architecture of the YOLO model contains 24 convolution layers followed by two fully connected layers. YOLO is later improved with different versions such as YOLOv2 [7], YOLOv3 [8] or YOLO-LITE [9], YOLOv4 [10] in order to improve detection performance and reduce computational time.

III. Algorithm Description

Fig. 3 shows a block diagram of the moving objects of interest detection algorithm.

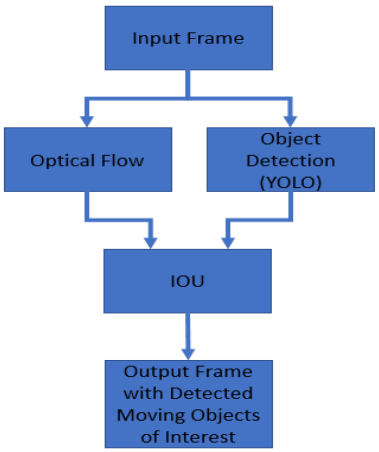


Fig. 3 Block diagram of the moving objects of interest detection algorithm

As can be seen from the block diagram shown in Fig. 3, the moving object of interest detection algorithm in this paper is implemented from two steps.

The input image is forwarded to two blocks running in parallel. The first block is responsible for motion detection in the image independently of which object from the scene caused that motion. This block uses the conventional method for motion detection in an image - Optical flow, primarily because of the high execution speed of this method in real-time. As the advantage of Dense over Sparse Optical flow has been presented in the literature [6], Dense optical flow has been applied in this work for motion detection. The result of calculating Optical flow is a flow matrix that is represented by its *hsv* (hue, saturation, value) model, and then converted into the RGB model. Converting the obtained RGB model to a grayscale image and then binarizing with a defined threshold *Tbin* produces a binary image in which the white pixels present the movements in the image. In such a binary image, it is then necessary to find the closed contours of the white pixels that actually represent the moving objects. For the obtained contours, the next step is comparing their surfaces with a defined threshold *Tcont*. In the set of moving objects, only those whose surface exceeds the defined threshold *Tcont* should be retained in order to eliminate very small objects or detected movements that present noise in the image. As a result of this procedure and the output of this block are bounding boxes around the regions detected as moving objects.

The second block to forward the input image is the block for detection objects of interest. Bounding boxes around the objects of interest are the output of this block. The detection method used in this work is the YOLO object detection method of version 3 [8], which has been shown to have very good performance in objects detection from the YOLO method class, and can be executed fast enough on a computer with a GPU for possible real-time implementation.

Detections of moving objects and detections of objects of interest are then forwarded to the block to calculate the IOU metric between them. By the IOU metric is actually determined whether the detected movements in the image

were caused by the objects of interest. If the overlap between detected movements and detections of objects of interest is sufficient and exceeds the defined threshold *Tiou*, it can be said that the detected objects of interest are in movement.

The output of the algorithm is an image with bounding boxes around moving objects of interest.

IV. Experimental work

The system setup used in this paper is based on vMSIS3 (Vlatacom Multi-Sensor Imaging System) [11]. Integrating visible and infrared imaging sensors and providing ultra-long-range target detection, recognition, and identification, vMSIS is a state-of-the-art monitoring and surveillance system. It contains of three video channels: visible light (FULL HD resolution: 1920x1080 pixels), thermal (resolution: 640x480) and short-wave infrared (resolution: 720x576 pixels).

The Algorithm is implemented in Python programming language using the OpenCV library to perform most of the functions related to image processing. The algorithm is applied to the compressed video signal. The stream from cameras was achieved via RTSP (Real-Time Streaming Protocol).

The algorithm is initialized by the first frame of the video sequence. For each subsequent frame that arrives from the camera, it is observed in relation to the previous one whether and what kind of movement occurred in the video sequence. After the input frame arrives, the Optical flow calculation for two successive frames is started. Dense Optical flow [12], *Farneback* implementation [13] from OpenCV library was applied. In parallel with the Optical flow calculation, the input frame is forwarded to the object of interest detector, YOLO algorithm, version 3. The YOLO method is implemented in the C programming language in the Darknet framework [14] using the CUDA acceleration library on the GPU. Object detection was performed on a reduced resolution image (448x448 pixels) primarily due to the execution speed to enable real-time operation. Using the NVIDIA RTX 2080 GPU for 448x448 pixel images, the processing time per frame is about 16ms. Reducing the resolution affects the detection performance (lower resolution - poorer detection), but the empirical assessment has shown that the resolution used is quite satisfactory for the quality of the detector in relation to the execution speed. Processing time per frame required for the Optical flow calculation on the same resolution (with parameters: *Tbin* = 50 and *Tcont* = 0.35% of the image area) is about 8ms, while the time consumption for the whole algorithm is about 27ms per frame ~ 37 frames per second (fps). The achieved speed is quite satisfactory for cameras that operate with 30 fps. The YOLO model pretrained on the COCO data set [15], which includes classes of pedestrians and cars, was used for object detection on visible light and SWIR images. Although the COCO dataset contains only the color images, the model was able to generalize well enough that the same model can be applied to the SWIR images. For detection on thermal images that are significantly different from color and SWIR images, pretrained model on COCO dataset was additionally trained with 9229 thermal images (8314 from FLIR dataset [16] and 915 taken from VMSIS thermal channel) to further improve detection performance on thermal images, especially for small objects at greater distances from the camera. Images from FLIR dataset were labeled for car and pedestrian classes, while for the purposes of this work we manually labeled images from VMSIS thermal channel for car and pedestrian classes using *Yolo mark* - application for marking bounded boxes of objects in images for training YOLO detector [17].

For the obtained detections of moving objects and detections of objects of interest, their overlap is then determined by applying the IOU metric, and if the overlap is sufficient and exceeds the threshold of 0.5, a decision is made that detected objects of interest are in motion and only bounding boxes around these objects are retained represents the end result of the algorithm.

In Fig. 4 are given three image sequences (Visible, SWIR and Thermal) that show the results of the algorithm, while Table I presents objective performance of the algorithm on the same sequences. In Table I Accuracy is the percentage of video sequence frames in which all moving objects of interest are detected, while false alarm is the percentage of frames in which some of the objects of interest that are not in movement are detected as moving.

In visible and thermal image sequences, cars were selected as objects of interest. The observed sequences present scene where pedestrians are beside the cars, and in the background is parking with static cars. The scene illustrates a situation with moving objects of interest, but also objects of interest that are not in movement. The algorithm distinguishes between objects of interest in movement (moving cars at the crossroads) and other objects in the scene (moving pedestrians and static cars in parking). In the sequence of SWIR images pedestrians were selected as objects of interest. The observed scene contains pedestrians standing, as well as those in motion. The algorithm detects when pedestrians who in the previous scenes were not in the frame appear, and if they are in motion, they are detected with a red bounding box.

Detection accuracy is mostly affected by the partial occlusions that are very often - a static or moving object covers an object of interest, image contrast changing, and always present noise especially in infrared images. The lowest accuracy is obtained for SWIR sequence, and the main reason for that is used detector of objects of interest, trained with visible images. The false alarm error occurred due to the appearance of any moving object in front of the object of interest that is not in movement, which is the main weakness of the simple IOU comparison between the objects of interest detections and detections of movement in the image.

TABLE I

	Number of frames	Accuracy [%]	False alarm [%]
Visible	622	68.3	2.03
SWIR	724	53.8	3.18
Thermal	600	80.5	3.32

V. Conclusion

The paper shows that the combination of the Optical Flow and the YOLO algorithm can be very successfully used to detect objects of interest in motion in a video sequence. As Optical flow is an exact method, regardless of the type of sensor and the moving object, using Optical flow, motion in a video sequence can be detected very successfully. However, the YOLO detector, like any other deep learning-based model, largely depends on the training set used for training. Object detection problem in color images is largely solved and the detection performances are very good. However, the problem of detection of objects of interest on infrared images remains open for further improvements. The most common problem is the lack of data of infrared images for training, primarily due to the cost of the infrared sensors themselves. Although insufficient, publicly available sets of LWIR and MWIR images can be found, but they are almost non-existent for SWIR sensors. As future work in this area and the improvement of the presented algorithm is seen in the creation of bigger infrared image datasets for detectors training to achieve better performance on LWIR and SWIR images.

References

[1] J. Lloyd, "Thermal imaging systems", Springer Science & Business Media, 2013.

[2] Beauchemin, S. S., & Barron, J. L., "The Computation of Optical Flow", *ACM computing surveys (CSUR)*, *27*(3), 433-466.

[3] Redmon J., Divvala S., Girshick R., & Farhadi A. (2016), "You Only Look Once: Unified, Real-Time Object Detection", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 779-788).

[4] T. Lim, B. Han, J. Han, "Modeling and Segmentation of Floating Foreground and Background in Videos", Pattern Recognition 45 (4) (2012) 1696-1706.

[5] Farnebäck, G. (June, 2003), "Two-frame Motion Estimation Based on Polynomial Expansion", Scandinavian Conference on Image analysis, (pp. 363-370). Springer, Berlin, Heidelberg.

[6] Chapel M. N., Bouwmans T. (2020), "Moving Objects Detection with a Moving Camera: A Comprehensive Review", *arXiv preprint arXiv:2001.05238*.

[7] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," arXiv preprint, 2017.

[8] Redmon J., Farhadi A. (2018), "Yolov3: An Incremental Improvement", *arXiv preprint arXiv:1804.02767*.

[9] Huang, R., Pedoeem, J., & Chen, C. (2018, December), "YOLO-LITE: a Real-Time Object Detection Algorithm Optimized for Non-GPU Computers", 2018 IEEE International Conference on Big Data (Big Data) (pp. 2503-2510).

[10] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020), "YOLOv4: Optimal Speed and Accuracy of Object Detection", *arXiv preprint arXiv:2004.10934*.

[11] Vlatacom Institute: Electro-optical systems VMSIS3, product brochures, available on-line: https://www.vlatacominstitute.com/electro-optical-systems, accessed on 09-July-2020.

[12] OpenCV Optical flow https://docs.opencv.org/3.4/d4/dee/tutorial_optical_flow.html, accessed on 09-July-2020.

[13] OpenCV Farnerback optical flow https://docs.opencv.org/3.4/dc/d6b/group__video__track.html#ga5d10ebbd59fe09c5f650289ec0ece5af, accessed on 09-July-2020.

[14] https://pjreddie.com/darknet/yolo/, accessed on 09- July -2020.

[15] COCO dataset, http://cocodataset.org/#home, accessed on 09-July-2020.

[16] https://www.flir.com/oem/adas/adas-dataset-form/, accessed on 09-July-2020.

[17] https://github.com/AlexeyAB/Yolo_mark, accessed on 04-September-2020.

Fig. 4 Examples of the moving objects of interest detection algorithm

Infrared Focal Plane Arrays Performance Achievements and Future Trends

Dragana Perić, *Member, IEEE,* Branko Livada

***Abstract*—This paper gives a review of current infrared focal plane arrays from the point of view of the practical application capabilities in the long range surveillance systems. We have tried to point out the vital components of the IR detector and gave the current trend of development that will be useful for a system architect whose task is to select components from the most competitive technologies available on the market now and to have in mind what is expected to be available in the near future.**

***Index Terms*—Infrared technology, infrared detectors, Infrared focal plane array, System architecture, Long range surveillance.**

I. Introduction

Infrared Focal Plane Array (IR FPA) is a sensing device, consisting of an array of light-sensing pixels at the focal plane of a lens that is sensitive to infrared radiation. The device operates by converting infrared photons to an electrical signal, and using it to construct an image of the sample. Strictly speaking detector array should be placed in the image plane that is close to focal plane but not exactly the same.

Five decades ago, a new semiconductor concept introduced Charge Coupled Devices (CCD) which paved the way to solid state imaging systems [1-2]. Successes from implementation of silicon CCDs in visible spectrum were followed by employing similar techniques to obtain IR FPAs that were realized in integrated two-dimensional arrays of detectors on the focal plane with multiplexed readouts [3].

Nevertheless, physical differences between visible CCD and IR FPA required additional procedures to be undertaken. Most obvious difference is resulting from the material used for detector and multiplexer readouts. While visible CCDs have both realized in silicon, IR FPAs have to cope with different type of materials (narrow bandgap semiconductor photon detectors, silicon multiplexers) and technology challenge of their interconnection. Also, narrow bandgap materials used in IR FPAs imposed the necessity of cryogenic cooling in order to decrease electronic noise to approach the photon noise limit. For this reason such IR FPAs are constructed in integrated dewars that complicate the detector design and impose strict requirements for electrical and mechanical interfaces [4].

IR technology was military used and controlled technology and its extraordinary advances in capabilities within a short time period during the last century were boosted by Cold War arms race [5]. These advances reflect in researches considering photon IR detection technology semiconductor material science and sophisticated manufacturing technologies.

IR FPA manufacturing technology is expensive on the one hand but should be high volume production on the other hand. Since IR FPA found their place also in civilian applications, even in our homes and mobile phones, the quantity of IR FPA is increasing and high volume production is becoming reality.

As commercial application need less expensive detectors, also uncooled IR FPA are being developed, offering good performances while less complicated IR FPA manufacturing, operating at room temperature therefore not needing cryogenic cooler. In some medium range surveillance application this type of detectors is also interesting. Such detectors are used in Shortwave Infrared (SWIR) and Longwave Infrared (LWIR) spectral range.

In the literature there are a lot of various reviews of the IR detector technology examples mainly from scientific development results. These reviews are good for judging scientific advancement, strategic technology development planning, manufacturing facility development and deployment. Goal of this paper is to give some guidelines to a systems architect of a long range surveillance system for selecting the proper IR FPA in order to achieve desired performances. We will point out some parameters that are important and specific for IR FPA and discuss the current trends of IR FPA development.

II. TRENDS IN INFRARED FOCAL PLANE ARRAYS

Historically, IR FPAs are appearing in the second generation (staring systems–electronically scanned). On the detector roadmap, third generation is considered staring systems with large number of pixels and two-color functionality, and fourth generation (staring systems with very large number of pixels, multi-colour functionality and other on-chip performance improvement functionalities [5].

The historical time line of the IR detector road map is illustrated in Fig. 1. It took more than half of century to pass the way from single detector to high density FPA. The spectral sensitivity of the various detector types is illustrated in Fig.2, The lot of different narrow band semiconductor

Dragana Perić is with the Vlatacom Institute of High Technologies, 5 Milutina Milankovica, 11070 Belgrade, Serbia (e-mail: dragana.peric@ vlatacom.com).

Branko Livada is with the Vlatacom Institute of High Technologies, 5 Milutina Milankovica, 11070 Belgrade, Serbia (e-mail: branko.livada@ vlatacom.com).

materials were considered as good candidates, but only sone of them survived the race.

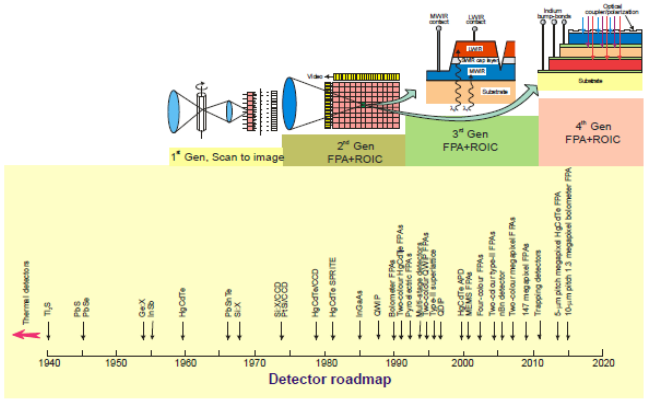


Fig. 1. History of the development of infrared detectors and systems. [5]

Different detector materials

Among materials used in third and fourth generation IR detectors which are selected due to their advantages [6-9] are:

- HgCdTe detectors are used to image rapidly moving objects for the very short integration time.
- QWIP devices had excellent homogeneity and better Noise Equivalent Temperature Distance (NETD), but their integration time had to be much longer.
- Sb-based III–V material systems offered mechanical robustness and have quantitatively weak dependence of band gap on composition,
- Type II InAs/GaInSb superlattices offer the capacity to tune the cutoff wavelength between 3 and 30 μm by varying the individual layer thickness [9]. Detectors have very uniform image (lower NEDT) and lower unit cost.

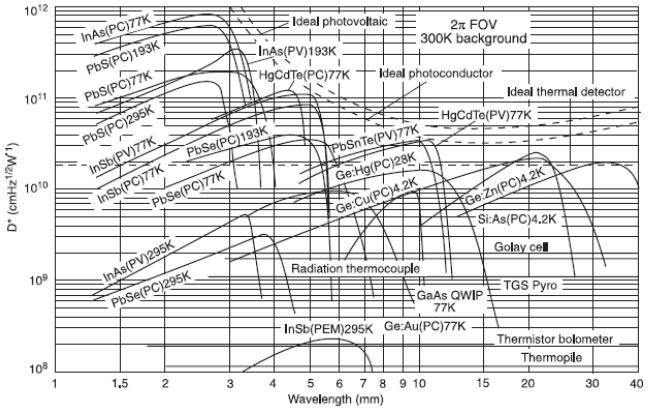


Fig. 2. Comparison of detectivity *D** of various available detectors when operated at the indicated temperature. [6]

As we see, most of the IR detector materials, in order to provide good detectivity (sensitivity to IR radiation), operate at cryogenic temperatures and require cooling.

Uncooled IR detectors materials

Uncooled IR FPA technology failed to attract much attention in the beginning, but around 1992 with promising results in LWIR detection, they became popular [10, 11]. Materials used for uncooled microbolometers are principally:

- Vanadium oxide – VOx - high performance demonstrated with this technology, lower NETD.
- Amorphous silicon – α-Si - technology much more oriented on low price commercial applications.

Another spectral region, Shortwave infrared (SWIR), also offers uncooled IR detection and has recently gaining a lot of application both in commercial apart from its use in military. Material that have the predominant use is InGaAs, which can be used with Termo Electrical (TE) cooling in order to minimize noise effects and obtain high performance.

Therefore, selection of the technology which is best to use is driven by the specific requirements of the system and the project application.

A. Trends in detector materials development

Actual strategies of leading institutions for planning IR detector development, that are driving the progress in this field, are in setting up an alternative business model for production in which there is a tendency to expand cooperation from dedicated military to commercial foundries in order to reduce production and maintenance costs [12]. For the last 50 years, HgCdTe was the most used material for IR FPA used in military. Since it is costly, low yield, difficult to process and not used for commercial products, strategy is turning to other materials from III-V based structures, that can offer affordable production, high yield, high performances even on higher operating temperature, which can be manufactured using commercial foundries. Superlattice structures based on antimonide, offer similar features as HgCdTe because their spectral response can be tuned to cover from SWIR region to Very Longwave Infrared (VLWIR) and their quantum efficiency is high. Their advantage over older technology is in being more robust and flexible for design.

Major goals in front of the new III-V based design and production are:

- Provide alternative to HgCdTe for LWIR spectral range detection and dual band detectors.
- Provide alternative to cryogenically cooled InSb detectors in Midwave Infrared (MWIR) spectral detection by Type II Strained Layer Superlattice (SLS) and High-operating Temperature (HOT) technology. This could increase overall system Mean Time Between Failure (MTBF) because cooler lifetime is increased.
- Large format FPA high yield production using high

diameter (up to 8”) substrate material, that is compatible with well developed Silicon Very Large Scale Integration (VLSI) technology.

- Decrease cost of the detector.

B. Detector cooling

Cooled IR FPAs need special type of architecture that is called Integrated Dewar Detector Cooler electronics Assembly (IDDCA) in order to provide optimal working environment to the IR FPA and to deliver digital data to user system. In Fig.3. block scheme with main components of the cooled IR FPA detector architecture is presented.

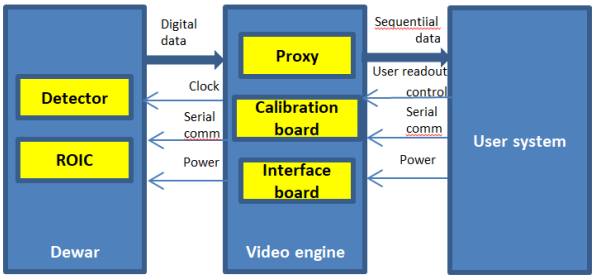


Fig. 3. Block scheme of the cooled IR FPA detector architecture

Detector is set in cooled environment and in order to eliminate stray light internal cold shield is provided that matches required optics F number. Feedthrough unit with adequate number of pins is used for connection of FPA with electronics of read out circuitry. Dewar is integrated with Stirling type cooler of required power.

As a new trend in cooled IR development brings HOT detectors with higher operating temperature 150K, required power of the Stirling cooler is decreased and that results in increase in lifetime of the cooler which is often critical component for MTBF of a cooled electro optical system. An increase in the FPA temperature up to 150K and above improves cooler thermodynamic efficiency reduces the detector assembly thermal losses. These are potential benefits allowing a cryocooler's size, weight and power consumption reduction and also improved performance and low price [13]. From the point of view of systems architect it is important to be aware of all advantages and drawbacks of a cooled system, that although offers the highest optical performance, need to be dimensioned well for overall cost of maintenance and in this evaluation cryogenic cooler plays an important role.

C. Read Out Circuitry ROIC

Read Out Circutry (ROIC) is responsible for transfering data from IR FPA to the user systems. IR FPA sensors pixel size is impacted by the challenges associated with hybridization of the detectors material to the silicon ROIC’s (bonding of indium bumps), but efforts that have been recently made and technology advancements made possible realization of smaller pixel IR FPAs. For alignment optimization, coefficient of thermal expansion between the materials of the ROIC substrate, detector epi layers, detector substrate etc., have to be taken into account and compensated using compensatory materials in order to minimize thermal-induced deflection.

Small pixel performance in IR systems requires the optimum trade balance between the optics design, the spectral bands, integration time and the applied signal processing techniques [14-18]. In new trend of IR detector developments pixel is smaller while FPA dimension is increased with more pixel elements.

ROIC can comprise more advanced image processing functionalities as shown in Fig.4, or they can be implemented in further processing chain, so called video engine or video core unit.

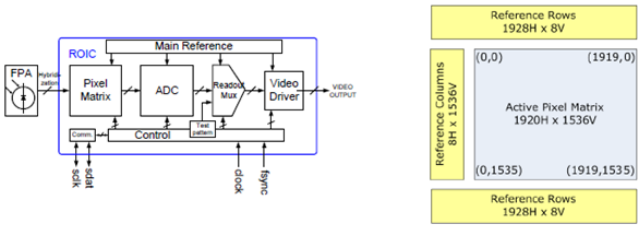


Fig 4 . High definition ROIC structure [19]

D. Video Engine Processing

Video engine is the electronics that performs image processing. It is usually provided by the camera manufacturer and offers several levels of integrations by modular design. Usually it is composed of several electronic boards, which split different functionalities for different level of image processing. Terminology used is most frequently referring to proxy board, calibration board and interface board of video engine. Typical functionalities of proxy board include low noise power distribution, cooler control, shutter control, Termo Electric Cooler (TEC) control and video analog to digital conversion. After this basic video signal acquisition it is necessary to do some corrections and calibrations like Non-Uniformity Correction (NUC), Bad Pixel Replacement (BPR), Dynamic Range Compression (DRC) and probably some more advanced processing, then to pass the video signal further to interface board that will put it in a delivery format (analog or digital like Camera Link, HD-SDI, Ethernet etc.).

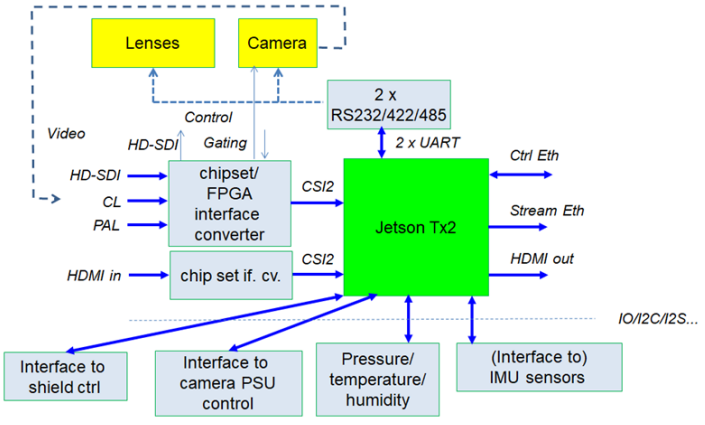


Fig.5. Block diagram of single vVSP channel processing unit

After this basic processing, system integrator can add more processing blocks in which other controls and algorithms are performed. In the Fig.5 block diagram of part of the Vlatacom

Institute video single vVSP channel processing unit is presented.

III. FPA APPLICATION CRITICAL PARAMETERS

When selecting IR detector for long range surveillance application, first decision have to be made about selecting the working spectral sensitivity band. Usually for coastal scenarios, MWIR spectral range (3-5 μm) is preferred, and for land application, medium distances LWIR (8-14 μm) uncooled detectors can answer the project need. As we discussed in the previous work, cooled IR FPA require use of cryogenic coolers that increase the overall system price and require costly system maintenance. However, current trends towards using HOT detectors are making possible the use of high performance IR detectors with lower prices and longer operational time with coolers' MTTF declared above 10.000 hours.

Technological improvements in manufacturing processes have enabled pixel minimization and also high diameter substrate material so it possible to manufacture IR FPA that have 1920 x 1024 pixels, or 1920 x 1536 with 10 – 15 micron pixel technology.

In order to optimize the overall dimensions and weight of the system it may be practical for a system integrator to implement its own electronics for video processing and, if possible, to use input signal as close as possible to the ROIC circuitry of the IR FPA. In that case, knowledge has to be acquired about manufacturing processes, and methodologies for compensation of detector non-uniformity and bad pixels. From the other side, this type of overall signal processing that is under control of the integration engineer offers more flexibility in customizing the solution for the specific project and implementing the most advanced algorithms for video processing like many types of video enhancement, video stabilisation using external Inertial Measurement Unit (IMU) sensors etc.

IV. CONCLUSION

IR FPA market is rapidly changing because of the increase of high volume production for non-military and commercial applications. The major trend in IR detectors design implies use of commercial foundries in order to decrease detector price and delivery time. Also reliability of the new products is designed to be higher because of the optimal use of coolers at higher operating temperatures. Large format small pixel IR FPA are becoming reality for use in surveillance projects such as border protection, coastal surveillance etc. Image fusion with other channels benefits from this development since IR FPA formats and resolutions are approaching those in visible spectrum range. For the multiple camera system integrators it becomes possible to approach close to detectors output and to control image processing pipeline in order to be able to optimize overall system performances.

System architects while selecting the proper equipment should be familiar with most critical steps of IR FPA manufacturing chain and strategical planning of future IR FPA generations development.

REFERENCES

[1] W. S. Boyle, G. E. Smith, „Charge Coupled Semiconductor Devices“, *The Bell System Technical Journal* , pp.587-593, 1970

[2] G. E. Smith, „The Invention and Early History of the CCD“, Nobel Lecture, December 8, 2009

[3] D. A. Scribner , M. R. Kruer , J. M. Killiany, „Infrared Focal Plane Array Technology“, *Proceedings of the IEEE*, Vol. **79**, No 1 , pp. 66-85, 1991

[4] Latika Becker, „Current and Future Trends in Infrared Focal Plane Array Technology“, *Proc. of SPIE* Vol. **5881**, 588105, (2005)

[5] P. Martyniuk, J. Antoszewski, M. Martyniuk, L. Faraone, and A. Rogalski, „New concepts in infrared photodetector designs“, *Applied Physics Reviews* **1**, p. 041102, 2014

[6] A.Rogalski, "Optical Detectors for Focal Plane Arrays“, *OPTO-ELECTRONICS REVIEW* **12**(2), 221–245, 2004

[7] Antoni Rogalski, "Progress in focal plane array technologies", *Progress in Quantum Electronics*, **36**, pp. 342–473, 2012

[8] A Rogalski, "History of infrared detectors", *OPTO−ELECTRONICS REVIEW* **20**(3), pp. 279–308, 2012

[9] A Rogalski, "New material systems for third generation infrared photodetectors", *OPTO-ELECTRONICS REVIEW* **16**(4), 458–482, 2008

[10] Masafumi Kimata, „Uncooled Infrared Focal Plane Arrays“, *IEEJ Trans*, **13**, pp. 4–12, 2018

[11] J.L. Tissot, „IR detection with uncooled focal plane arrays: Stae-of-art and trends“, *OPTO-ELECTRONICS REVIEW* **12**(1), pp.105–109, 2004

[12] Meimei Z. Tidrow and Donald A. Reago Jr. "VISTA video and overview (Conference Presentation)", Proc. SPIE 10177, Infrared Technology and Applications XLIII, 101770M (21 June 2017); https://doi.org/10.1117/12.2266259

[13] Eli Levin, Amiram Katz, Zvi Bar Haim, Ilan Nachman, Sergey Riabzev, Dan Gover, Victor Segal, and Avishai Filis "RICOR Cryocoolers for HOT IR detectors from development to optimization for industrialized production", Proc. SPIE 10180, Tri-Technology Device Refrigeration (TTDR) II, 1018005 (5 May 2017); https://doi.org/10.1117/12.2262334

[14] Richard F. Lyon, „A Brief History of 'Pixel'“, *IS&T/SPIE Symposium on Electronic Imaging*, San Jose, California, USA, 15–19 January 2006

[15] John. T. Caulfield, Jerry A. Wilson, Nibir K. Dhar, „Performance benefits of sub-diffraction sized pixels in imaging sensors*", Proc. of SPIE* Vol. **9100**, p.91000J, 2014

[16] Pratik CHATURVEDI, Nicholas X. FANG, " Sub-diffraction-limited far-field imaging in infrared“, *Front. Phys. China*, 5(3): p.324–329, 2010

[17] Kenneth I. Schultz, Michael W. Kelly, Justin J. Baker, Megan H. Blackwell, Matthew G. Brown, Curtis B. Colonero, Christopher L. David, Brian M. Tyrrell,and James R. Wey „Digital-Pixel Focal Plane Array Technology“, *LINCOLN LABORATORY JOURNAL* Vol**. 20**, No 2,pp.36-71, 2014

[18] Bruno Fièque, Lilian Martineau, Eric Sanson, Philippe Chorier, Olivier Boulade, Vincent Moreau, and Hervé Geoffray "Infrared ROIC for very low flux and very low noise applications", *Proc. SPIE* **8176,** p.81761I, 2011

[19] G. Gershon, E. Avnon, M. Brumer, W. Freiman, Y. Karni, T. Niderman, O. Ofer, T. Rosenstock, D. Seref, N. Shiloah, L. Shkedy, R. Tessler, and I. Shtrichman "10μm pitch family of InSb and XBn detectors for MWIR imaging", Proc. SPIE 10177, Infrared Technology and Applications XLIII, 101771I (16 May 2017); https://doi.org/10.1117/12.2261703

Humidity Sensor Circuits Based on the Current Processing

Predrag B. Petrović, Marija-Vesna Nikolić , Mihajlo Tatović

***Abstract*—A novel electronic conditioning circuits based on the current-processing technique for accurate and reliable humidity measurement, without post-processing requirements, are presented. The pseudobrookite nanocrystalline (Fe_2TiO_5) thick film was used as capacitive humidity transducer in proposed design. The interface circuitry was realized in TSMC 0.18µm CMOS technology. The sensing principle of the sensor was obtained by converting the information on environment humidity into a frequency variable square-wave current signal. The proposed solution features high linearity, insensitivity to temperature, as well as low power consumption. The sensor has a linear function with relative humidity in the range of Relative Humidity (RH) 30-90%, error below 1.5% and sensitivity 8.3 x1014Hz/F evaluated over the full range of change. A fast recovery without the need of any refreshing methods was observed with a change in RH. The total power dissipation of readout circuitry was 1mW.**

***Index Terms*—Current-processing, humidity sensor, CMOS integrated circuit.**

I. Introduction

HUMIDITY sensors have a wide application in everyday practice, including agriculture, monitoring climate change, food storage processes and in the operation of various home appliances [1, 2]. In order to meet the demands of such applications, humidity sensors should provide high sensitivity and linearity in response, in a wide range of possible changes in processed humidity under different temperature conditions. In addition to the aforementioned demands, the sensor circuits must provide long-term stability, short response times with low energy consumption. There are different types of humidity sensors in accordance with the physical principle used to make the conversion: resistive, mechanical, gravimetric, capacitive and thermal humidity sensors [3]. Most electronic circuits, representing the interface between the sensor itself and the processor unit, are based on the use of operational amplifiers [4]. The growing desire for miniaturization of such systems and reduction of their consumption places increasing challenges in the design process before the circuit designers. CMOS technology is a logical response to such challenges, but due to the different ratio of the width and length (W/L) of the transistors used, there is a trade-off between the speed, gain, power and other parameters [5]. The sensor circuit has often been based on the principle of a resistor sensor [6]. Such sensors can detect changes in temperature, humidity, pressure, etc. The capacitive sensor on the other hand can process moisture, speed, and acoustic shift and so on. Sensing circuits that detect the change of resistance enable a relatively simple realization of the accompanying electronic interface.

Everything listed was the reason for the development of the sensor circuit suggested in this paper. Metal oxide semiconductor materials have been intensively investigated for application as humidity sensors [7, 8]. The humidity sensing mechanism of metal oxides is simple. It is based on water adsorption on the material surface that is composed of grains, grain boundaries and pores. Nanostructures and nanomaterials have led to many new applications of metal oxides due to changing and enhancing their microstructural properties [9]. Pseudobrookite (Fe_2TiO_5) is an iron titanium oxide, with a band gap similar to hematite with potential for application as a gas sensor. We have investigated possible application of pseudobrookite for NO gas sensing [10], but our recent work has focused on humidity sensing properties of this material [11].

Analogue interface circuits used in capacitive sensors are based on the application of one of the following methods: measurement based on the application of AC sources to detect the voltage and current at unknown capacitance; using a capacitance divider; resonance and bridge circuits containing the measured capacity; methods based on charge transfer; differential methods [2, 12].

This paper proposes a cheap, accurate, and reliable humidity sensor by integrating the sensing element and the conditioning circuits using standard CMOS technology for fabrication. We propose a new electronic interface circuit based on the concept of current mode processing, which is capable of converting information on environment humidity to variable frequency dependency current (or voltage) signals. To do this we used only one active element, DXCCTA- dual-X current conveyor transconductance amplifiers. Application of this active element is dictated by characteristics of the sensor element-pseudobrookite, and its equivalent impedance circuits.

Practically, design of interface electronic circuits is the central and completely new part of the paper, because all other parts are based on known configurations, not used up till now for practical realization of a humidity sensor. DXCCTA not used until now as the base component for realization of square-wave signals-converter of moisture to time depended

This work was supported in part by the projects OI 172057, 42009 and III45007, funded by the Ministry of Education, Science and Technological Development of the Republic of Serbia.

P. B. Petrović is with the Facuty of Technical Sciences, University of Kragujevac, 32000 Čačak, Serbia (e-mail: predrag.petrovic@ftn.kg.ac.rs).

M. V. Nikolić, was with Institute for Multidisciplinary Research, University of Belgrade, 11000 Belgrade, Serbia (e-mail: mariav@rcub.bg.ac.rs).

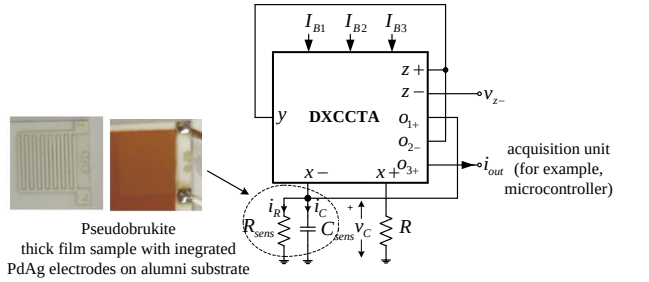
M. Tatović is with the Facuty of Technical Sciences, University of Kragujevac, 32000 Čačak, Serbia (e-mail: mihajlo.tatovic@ftn.kg.ac.rs).

current signal. We created a system for checking humidity information in real time. Moreover, the use of grounded passive components in circuit implementation is also beneficial from the integration point of view.

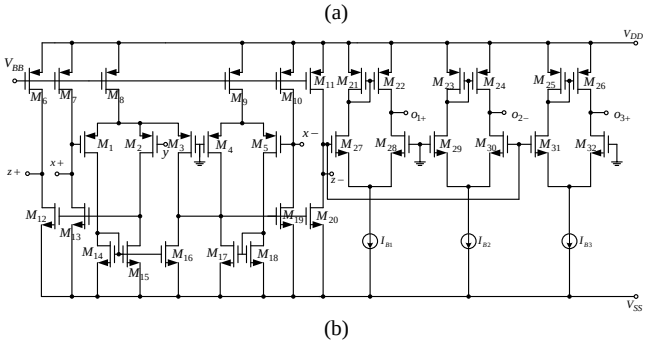
The proposed conditioning circuit was verified through the HSPICE simulation results carried using 0.18 μm CMOS technology, and can operate very well with nonlinearity less than 1%. This technology is a strong candidate for the easy-to-scale implementation of next generation electronics, such as the Internet of things (IoT) [13], LoRa-based sensor technology (for example the RN2483 LoRa transceiver module), built around a Semtech SX1276 transceiver [14], and printed passive/active electronics.

II. Proposed Electronic Interface Circuits

We used a newly proposed active element, DXCCTA as the base for the realization of the interface between the transducer and acquisition unit (for example, microcontroller PIC 18F45K80, a high performance 8-bit MCU [16]), Fig 1a. Practically, DXCCTA is a combination of two versatile active elements: DXCCII and an operational transconductance amplifier (OTA), with electronic tunability capability [16]. Thus, DXCCTA enjoys all the benefits of DXCCII and OTA, but they have never been used in realization of a sensor circuit. Some applications require an extra buffer in the active element to meet the requirement of an appropriate impedance level for the output signal [12]. This is not the case with our proposed sensor circuits, because we select port connections in a new and appropriate way. Fig. 1b shows CMOS implementation of the used DXCCTA.



(a)



(b)

Fig. 1. (a) Circuit of the proposed conditioning-interface circuits, (b) CMOS implementation of DXCCTA.

As we can see, DXCCTA has eight terminals of which *x+* and *x-* terminals are low impedance terminals whereas the terminals *y*, *z+*, *z-*, *o1+*, *o2-* and *o3+* are high impedance terminals. In CMOS implementation of DXCCTA, Fig. 1b, MOS transistors, M1-M20 form sub-block DXCCII, and MOS transistors, M21-M32 form OTA stages.

Detailed operation of DXCCTA in the saturation region is given in [16], and this operation mode is further utilized in operation of the proposed sensor circuit. The proposed humidity to frequency converter (Fig. 1a) comprises of a single DXCCTA, one grounded resistor and grounded humidity transducer, which we can equivalent represent as parallel connection of resistor and capacitor, R_{sens} and C_{sens} (transducer equivalent impedances). The output current mode square wave, i_{out} is explicitly available from the high impedance terminal, o_{3+}. Fig. 1a shows that terminal *z-* is floated, therefore the voltage, v_{z-} saturates either to V_{DD} or V_{SS} depending on the current i_{z-}. The voltage, v_{z-} in saturation mode is expressed as follows (1).

$$v_{z-} = \begin{cases} V_{DD} \ \ for\ i_{z-} \geq 0 \\ V_{SS} \ \ for\ i_{z-} < 0 \end{cases} \tag{1}$$

These levels of v_{z-} are high enough to saturate all OTAs and currents, i_{o1+}, i_{o2-} and i_{o3+} become totally dependent on bias currents of their respective OTA stage as per (2). The two saturation levels of i_{out} are I_{B3} and $-I_{B3}$. The duration of intervals in which this square wave current signal follows this saturation levels depend on the voltage dynamics across the sensor equivalent circuit. The threshold levels are V_{HT} (higher-upper threshold) and V_{LT} (lower threshold), and they are defined with a bias current I_{B2} and external resistance R. The operation of the proposed circuit can be described as follows: Suppose initially i_{out} is at its positive saturation level, I_{B3}. At the same time i_{o1+} and i_{o2-} will be I_{B1} and $-I_{B2}$, respectively. The current, i_{o1+} causes the capacitor to charge with the following dynamics:

$$i_{Csens}(t) = \frac{v_{Csens}(t)}{R_{sens}} + C_{sens}\frac{dv_{Csens}(t)}{dt}, 0 \leq t \leq T_{ON} \tag{2}$$

$$\Rightarrow R_{sens}C_{sens}\frac{dv_{Csens}(t)}{dt} + v_{Csens}(t) = I_{B1}R_{sens}$$

The limiting values of voltage across the sensor were defined with the threshold level V_{HT} of the v_{x+}. When v_{x+} becomes just higher than the level V_{HT}, the sum of currents i_{z+} and i_{o2-} becomes negative causing the v_{z-} to saturate to V_{SS}. Therefore, i_{out} is now changed to $-I_{B3}$ from positive saturation level, I_{B3}. Thus, i_{o1+} and i_{o2-} are now changed to $-I_{B1}$ and I_{B2}, respectively. A negative current, $-I_{B1}$ at terminal, o_{1+} causes the sensor to discharge with the above defined dynamics until v_{x+} reaches the threshold level V_{LT}. When v_{csens} becomes just less than V_{LT}, the sum of currents i_{z+} and i_{o2-} becomes positive causing v_{z-} to saturate to V_{DD} and i_{out} is again changed to I_{B3}. The amplitude of i_{out} is expressed as

$$i_{out} = \begin{cases} I_{B3} \ \ for\ v_{z-} = V_{DD} \\ -I_{B3} \ \ for\ v_{z-} = V_{SS} \end{cases} \tag{3}$$

The threshold levels of v_{x+} and the peak to peak amplitude, $v_{x+(p\text{-}p)}$ are given as

$$V_{HT} = I_{B2}R \ and \ V_{LT} = -I_{B2}R$$
$$v_{x+(p-p)} = V_{HT} - V_{LT} = 2I_{B2}R \tag{4}$$

The on and off time periods (T_{ON} and T_{OFF}, respectively) are obtained from voltage across the sensor (their waveforms) by comparing the slope during these two time periods

$$v_{Csens}(t) = I_{B1}R_{sens}\left(1-e^{-t/\tau}\right)+V_{TL}e^{-t/\tau}, \tau = R_{sens}C_{sens}$$
$$\Rightarrow V_{TH} = I_{B1}R_{sens}\left(1-e^{-T_{ON}/\tau}\right)+V_{TL}e^{-T_{ON}/\tau}$$
$$\Rightarrow T_{ON} = \tau \ln\frac{V_{TL}-I_{B1}R_{sens}}{V_{TH}-I_{B1}R_{sens}} = \tau \ln\frac{I_{B2}R+I_{B1}R_{sens}}{I_{B1}R_{sens}-I_{B2}R} \quad (5)$$

During the off time period, the voltage across the sensor will be change with the following dynamics:

$$R_{sens}C_{sens}\frac{dv_{Csens}(t)}{dt}+v_{Csens}(t) = -I_{B1}R_{sens}, T_{ON} \le t \le T_{ON}+T_{OFF} = T$$
$$\Rightarrow v_{Csens}(t) = I_{B1}R_{sens}\left(e^{-(t-T_{ON})/\tau}-1\right)+V_{TH}e^{-(t-T_{ON})/\tau}, v_{Csens}(T_{ON}+T_{OFF}) = V_{TL} \quad (6)$$
$$\Rightarrow T_{OFF} = \tau \ln\frac{V_{TH}+I_{B1}R_{sens}}{V_{TL}+I_{B1}R_{sens}} = \tau \ln\frac{I_{B2}R+I_{B1}R_{sens}}{I_{B1}R_{sens}-I_{B2}R}$$

From (6), it is noted that both cycle periods are equal thus the duty cycle of the generated square current signal on port o_{3+} is fixed to 50%. The oscillation frequency, f_0 is obtained from T_{ON} and T_{OFF} as follows

$$f_o = \frac{1}{T_{ON}+T_{OFF}} = \frac{1}{2R_{sens}\,C_{sens}\ln\frac{I_{B2}R+I_{B1}R_{sens}}{I_{B1}R_{sens}-I_{B2}R}} \quad (7)$$

It is observed from (3) and (7) that the amplitude of i_{out} and oscillation frequency, f_0 are electronically and independently tunable via bias currents, I_{B3} and I_{B1}, respectively. Also, the period of the generated square wave current output signal directly depends on parameters of sensor circuits. This way we come into the position to indirectly recalculate the humidity of the environment in which we place our sensor from information on the generated frequency of the current output signal. The generated output signal is completely autonomous and its frequency has no effect on the capacitance of the sensor, which is very often seen in the so far known interface circuit realizations [17], in which the detection of moisture is based on the principle of moisture adsorption and desorption.

The performance of the proposed circuits can be further evaluated based on the sensitivity of its response relative to the sensors parameters C_{sens} and R_{sens}. Sensitivity (S) is defined as an incremental change in the output signal value relative to the incremental change in the sensor parameter [18]. According to such a criterion, the sensitivity of the analysed circuit for the interface is obtained as:

$$S_{C_{sens}} = \frac{\partial f_0}{\partial C_{sens}} = -\frac{1}{2R_{sens}\,C_{sens}^2\ln\frac{I_{B2}R+I_{B1}R_{sens}}{I_{B1}R_{sens}-I_{B2}R}}$$
$$S_{R_{sens}} = \frac{\partial f_0}{\partial R_{sens}} = \frac{\frac{I_{B1}I_{B2}RR_{sens}}{(I_{B2}R+I_{B1}R_{sens})(I_{B1}R_{sens}-I_{B2}R)}-\ln\frac{I_{B2}R+I_{B1}R_{sens}}{I_{B1}R_{sens}-I_{B2}R}}{2R_{sens}^2\,C_{sens}\ln^2\frac{I_{B2}R+I_{B1}R_{sens}}{I_{B1}R_{sens}-I_{B2}R}} \quad (8)$$

Based on the obtained relations (8), the sensitivity can be adjusted by properly selecting the values of the parameters, and for measured values (section IV) we can conclude that the proposed circuits offer satisfactory low sensitivity.

A. Humidity Transducer

The pseudobrookite humidity transducer was developed by screen printing thick film paste on alumina substrate with test interdigitated PdAg electrodes. This design is simple, and is commonly applied for sensing [19]. Pseudobrookite powder and thick film paste was synthesized and characterized in detail and this is described [11]. Interdigitated PdAg electrodes were first screen printed on alumina substrate and fired in a conveyer furnace at 850ºC for 10 minutes in air [11]. The electrode dimensions were: width 8mm, length 8mm, electrode spacing 0.25 mm (Fig. 1a). Five layers of pseudobrookite thick film paste were then screen printed on the substrate with electrodes, with the procedure described in detail in [11] achieving a porous nanocrystalline thick film layer about 60 μm thick (as each layer was~12 μm).

The influence of the change in relative humidity (RH) 30-90% of several thick film pseudobrookite sensors on complex impedance were monitored in a humidity chamber and analysed in detail in [11]. The response and recovery times were relatively rapid (16 s) and relatively low hysteresis (difference between absorption and desorption) were obtained showing that pseudobrookite thick film sensors are good candidates for application in humidity sensing.

III. Estimation of the Influences of Non-idealites and Effects on Measurement Accuracy

In a non-ideal case the terminal characteristics of DXCCTA can be described in the following manner

$$v_{x+} = \beta_1 v_y, v_{x-} = -\beta_2 v_y, i_{z+} = \alpha_1 i_{x+}, i_{z-} = \alpha_2 i_{x-},$$
$$i_{o1+} = \gamma_1 g_{m1} v_{z-}, i_{o2-} = -\gamma_2 g_{m2} v_{z-}, i_{o3+} = \gamma_3 g_{m3} v_{z-} \quad (9)$$

In a non-ideal case, the currents, i_{o1+}, i_{o2-} and i_{o3+} are:

$$i_{o1+} = \alpha_3 I_{B1}; i_{o2-} = -\alpha_4 I_{B2}; i_{o3+} = \alpha_5 I_{B3} \ \text{for}\ v_{z-} = V_{DD}$$
$$i_{o1+} = -\alpha_3 I_{B1}; i_{o2-} = \alpha_4 I_{B2}; i_{o3+} = -\alpha_5 I_{B3} \ \text{for}\ v_{z-} = V_{SS} \quad (10)$$

where, α_1, α_2, α_3, α_4 and α_5 are the non-ideal current transfer gains from i_{x+} to i_{z+}, i_{x-} to i_{z-}, I_{B1} to i_{o1+}, I_{B2} to i_{o2-} and I_{B3} to i_{o3+}, respectively; β_1 and β_2 are non-ideal voltage transfer gains from v_y to v_{x+}, and v_y to v_{x-}, respectively; γ_1, γ_2 and γ_3 are transconductance inaccuracies from v_{z-} to i_{o1+}, v_{z-} to i_{o2-} and v_{z-} to i_{o3+}, respectively.

TABLE I
DESCRIPTION SIMULATED VALUES OF PARASITIC COMPONENTS, I.E. THE PARASITIC IMPEDANCES OF DXCCTA

Parasitic	Simulated Values
R_y, C_y	1932x10^{12}Ω, 2.45 fF
R_{x+}	132Ω
R_{x-}	298Ω
R_{z+}, C_{z+}	30.2kΩ, 4.38 fF
R_{z-}, C_{z-}	30.3KΩ, 13 fF
R_{o1+}, C_{o1+}	58.2 KΩ, 3.27fF
R_{o2-}, C_{o2-}	58.2 KΩ, 3.31fF
R_{o3+}, C_{o3+}	58.2 KΩ, 3.35 fF

Furthermore, the various parasitic components, i.e. the parasitic impedances involved in DXCCTA are as follows: three small resistances R_{x+} and R_{x-} appear at $x+$ and $x-$ terminals whereas the parallel combinations of (R_y//(1/sC_y)), (R_{z+}//(1/sC_{z+})), (R_{z-}//(1/sC_{z-})), (R_{o1+}//(1/sC_{o1+})), (R_{o2-}//(1/sC_{o2-})) and (R_{o3+}//(1/C_{o3+})) appear at y, $z+$, $z-$, o_{1+}, o_{2-} and o_{3+} terminals-Fig. 1a, respectively. For the CMOS implementation

of DXCCTA in Fig. 1b, we measured the parasitic impedances of the terminals (using simulation in HSPICE programme), in order to evaluate their effects on processing capabilities of the proposed interface circuits. The values of these parasitic elements are given in Table 1.

Taking the non-idealities and the parasitic impedances into consideration, the proposed conditioning circuit is reanalysed. The amplitude of i_{out} is now modified and given as follows.

$$i_{out} = \begin{cases} \alpha_5 I_{B3} \ for \ v_{z-} = V_{DD} \\ -\alpha_5 I_{B3} \ for \ v_{z-} = V_{SS} \end{cases} \quad (11)$$

The modified threshold levels and peak to peak amplitude are given as

$$V_{HT} = \frac{\alpha_4 I_{B2} R'}{\alpha_1}; V_{LT} = -\frac{\alpha_4 I_{B2} R'}{\alpha_1} \quad (12)$$

$$V_{x+(p-p)} = V_{HT} - V_{LT} = \frac{2\alpha_4 I_{B2} R'}{\alpha_1}$$

where, $R'=R+R_{X+}$. The oscillation frequency is also modified as follows

$$f_o = \frac{1}{T_{ON} + T_{OFF}} = \frac{1}{2R_{sens}\, C' \ln \frac{\alpha_4 I_{B2} R' + \alpha_1 \alpha_3 I_{B1} R_{sens}}{\alpha_1 \alpha_3 I_{B1} R_{sens} - \alpha_4 I_{B2} R'}} \quad (13)$$

where, $C'=C_{sens}+C_{o1+}$. It is observed from (11)-(13) that non-idealities and the parasitic impedances of DXCCTA affect the amplitude of i_{out}, threshold levels, peak to peak amplitude and oscillation frequency, respectively. It is well known that, non-ideal gains deviate from unity only at higher frequencies. Therefore, the effects of these non-idealities could be neglected depending upon the operating frequency range. On the other hand, parasitic effects can be minimized by a proper choice of R and sensor parameters (C_{sens} and R_{sens}). The resistance, R must be selected such that $R>>R_{x+}$ and the capacitor, C_{sens} must be chosen such that $C_{sens}>>C_{o1+}$. Correct selection of the observed parameters leads to improvement in the dynamic range of the sensor circuit, with reduction in the effect of the parasitic components.

In order to further check the performance of the proposed interface circuit, in a situation where there is variation in the fabrication process-the production of semiconductor elements and voltage variation, Monte Carlo simulation (provided by the HSPICE software package itself) was performed in 1000 runs. During this analysis, the voltage supply of ±1.25 V, bias voltage of 0.42V, and bias currents of 50 μA amplitude was used, which resulted in the histogram in Fig. 2. It is assumed that due to possible variation in the manufacturing process, the threshold voltage of all MOS transistors deviates by 5% (Gaussian deviation) and that the variation in the supply voltage (V_{DD} and V_{SS}) is the order of 5% (Gaussian deviation). We assumed that the extreme PVT (Process Voltage Temperature) variations were in the range of +/- 5% (this tolerance is applied over the 0 °C to 100 °C temperature range).

On the basis of such conducted analysis, we are in a position to investigate the effect of the process parameters and the mismatch between transistors on the precision of processing. We define the lower and upper limits of the interval, which contains 95 % of error-absolute value of difference between the predicted and observed output value [20]. The standard deviation in the generated output current input signal was approximately 0.64 μA. It was noted that such changes do not lead to a larger deviation in the frequency of the generated current signal (order of 2%) and that the amplitudes of the output waveforms are not disturbed.

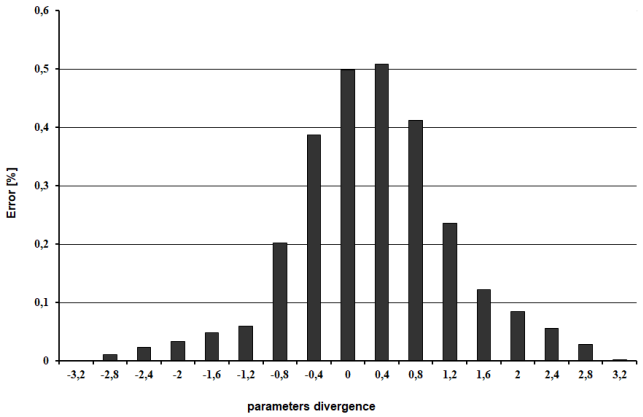


Fig. 2. Distribution of errors in the behaviour of the proposed interface circuit, for divergence in the value of parameters, from their nominal values.

IV. SIMULATION RESULTS

Simulations were performed using HSPICE with 0.18 μm TSMC CMOS process parameters. The supply voltages of ±1.25 V, and bias voltage, V_{BB}=0.42V were used in the simulation. The aspect ratios (*W/L* ratios) of MOS transistors used in the CMOS implementation of DXCCTA are given in Table 2.

TABLE II
MOS TRANSISTOR ASPECT RATIOS (W/L)

M_1-M_2	0.72/0.36
M_3-M_5	1.44/0.36
M_{14}-M_{15}	1.34/0.36
M_{16}-M_{18}	2.4/0.36
M_6-M_{13}, M_{19}-M_{20}	4.8/0.36
M_{21}-M_{26}	1.44/0.36
M_{27}-M_{32}	3.6/0.36

The value of resistance is chosen R=1 kΩ ($R>>R_{X+}$-specified in Table 2). For the specified values of magnitude of i_{out}=50 μA, $v_{C(p-p)}$=100 mV, the values of I_{B1}, I_{B2}, I_{B3} are found according to the design process as follows: $I_{B1}=I_{B2}=I_{B3}$=50 μA and C_{sens}= 500 pF, R_{sens}=4 MΩ. The transient responses of i_{out} and v_C are shown in Fig. 3a. The simulated oscillation frequency of 0.5 MHz is obtained in Fig. 3a, which is similar to the designed oscillation frequency. The transient responses of i_{out} and v_C when I_{B1} is changed to 80 μA are shown in Fig. 3b. The simulated frequency is now changed to 0.789 MHz (1% error) at I_{B1} =80 μA. It can be seen that the threshold levels of v_C and amplitude levels of i_{out} are not affected by the change in bias current, I_{B1}. Fig. 3c shows the electronic tuning of amplitude of i_{out} for different values of I_{B3} (40 μA and 60 μA, I_{B1}=40 μA, I_{B2}=60 μA). It can be seen that the amplitude of i_{out} is independently tunable without affecting the oscillation

frequency.

From the Figs. 3a and 3b we can see that voltage across the sensor possesses a virtually linear characteristic that results from the value of its time constant, which leads to very fast voltage fluctuations in the observed boundaries. Among other things, the proposed conditioning and conversion circuits shows the sensitivity of 8.3×10^{14}Hz/F, which is much better than other circuits used for comparison in Table 1.

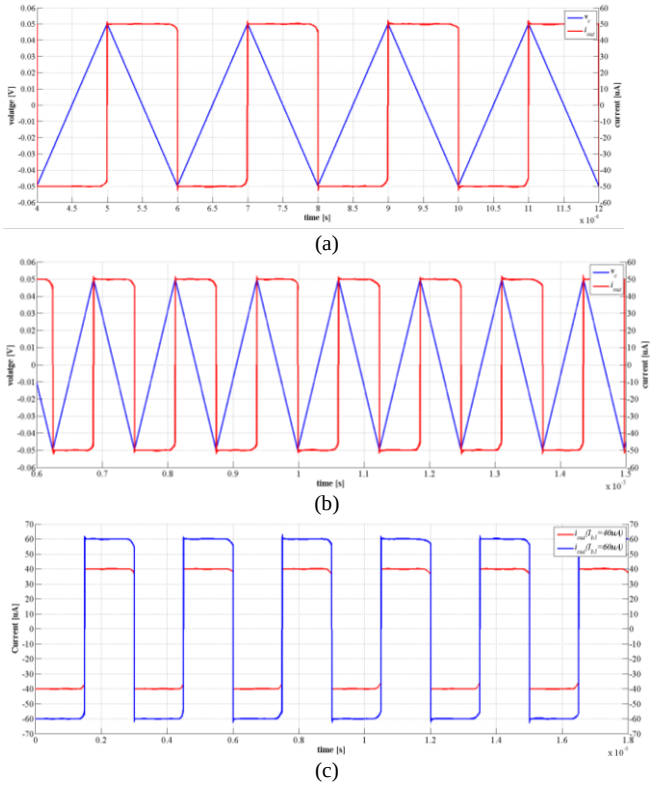


Fig.3. Transient response of proposed interface circuits (a) Simulated waveforms of i_{out} and v_C, $I_{B1}=I_{B2}=I_{B3}=50$ μA, $f_o=0.5$ MHz, (b) simulated waveforms of i_{out} and v_C, $I_{B1}=80$ μA, $I_{B2}=I_{B3}=50$ μA, $f_o=0.789$ MHz, (c) Electronic tuning of amplitude of i_{out} for different values of I_{B3} (40 μA and 60 μA).

For further laboratory and simulation verification of the proposed design we used a JEIO TECH TH-KE-025 temperature and humidity climatic chamber in the relative humidity range 30-90% [11] in which we placed humidity transducer. Prior to each measurement the samples were dried/heated for 20 minutes at 50°C to remove any moisture. The test sample was placed into the chamber and using wires soldered to the electrodes we established connection with acquisition card on our PC and then transferred to the MATLAB environment and after that to HSPICE, without altering to plot the curves. The humidity was varied between 30 and 90% at 25°C, by setting the desired humidity value. The simulation observed in this way (with input data obtained on real humidity transducer), waveforms of i_{out} and v_{x+} are shown in Fig. 4 - the response in the time domain of the proposed sensor system in a situation where the environmental humidity is changed. The simulation oscillation frequency in situation when moisture is 60%RH is 0.1542 MHz (0.58% error in comparison with calculated frequency). The electronic tuning of amplitude of i_{out} via bias current $I_{B3}=250$ μA is shown in Fig. 4c, which shows that the amplitude of i_{out} can be independently controlled without disturbing the oscillation frequency.

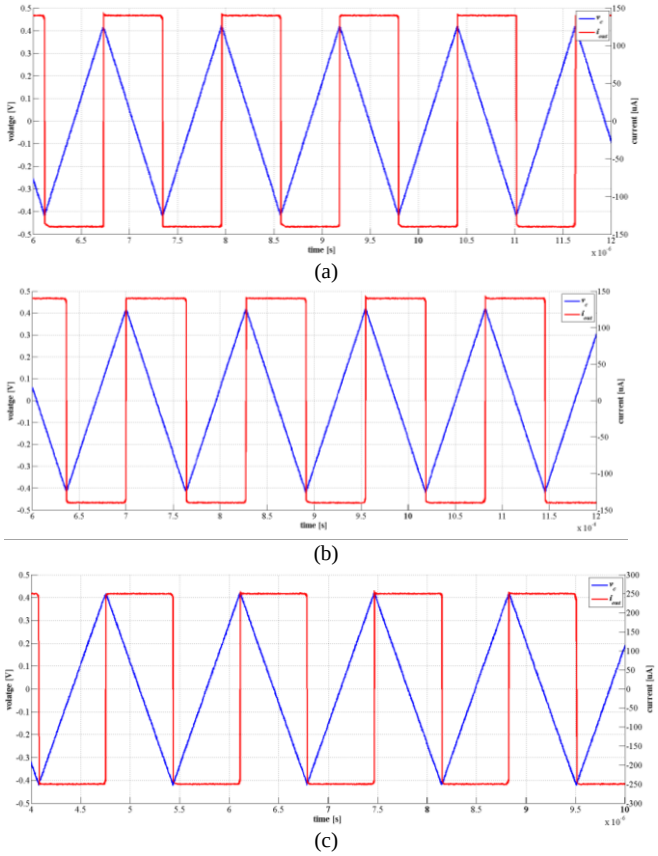


Fig.4. The time domain behaviour of the proposed interface circuits (a) 30%RH, (b) 60%RH, (c) 90%RH and bias current $I_{B3}=250$ μA.

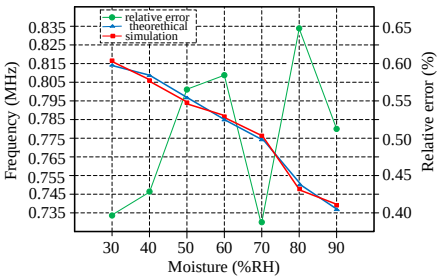


Fig.5. Features of the proposed interface circuits-change in the frequency of the output signal with moisture concentration and difference (error) between theorethical data and expectation-simulation results (theory versus simulation measurement).

Using the MATLAB environment we detected the frequency of the output current signal and subsequent signal conditioning (on external resistance connected to the input pin of the acquisition card to obtain a voltage equivalent). To count input edges of the square wave output signal, the timer as a counter has been used. The dependence of the frequency f of the output current signal on the humidity of the environment in which the sensor circuit is located (the chamber used during the laboratory performance test of the proposed sensor system) is shown in Fig. 5. It shows that the frequency is proportional to

the capacitance change with moisture content. On the same Fig. 5, we compared the simulation and theoretical measurement results. We can see that the obtained results are well correlated because the error (relative) in the operating range of 30% to 90% RH is below 0.65%. The proposed sensor system possesses satisfactory properties in terms of error and gives a linear frequency change relationship.

The response time of a sensor when exposed to moisture is defined as the time in which a sensor reaches 90% of the total response, while recovery is the time required for a sensor to return to 90% of the original baseline signal, when moisture is removed [11]. The average response time was about 16s, while the recovery time was very fast and the sensor recovered in 1s. The difference in response and recovery times was attributed to the microstructure of pseudobrookite thick films that represented a porous network of aggregated nanoparticles [11].

V. CONCLUSION

In this paper, a humidity sensor read out circuitry using DXCCTA has been designed. The proposed interface provides a simple interconnection with the associated processing unit without post-processing, with very low power consumption of 1mW. It is important to note that the proposed design can be fully realized in the form of an integrated circuit. The proposed solution is based on generating a fully autonomous current signal, the period of which is linearly dependent on the capacity of the humidity sensor. The possibility of precise humidity measurement in the range of 30% to 90% RH with error less than 1.5% was experimentally confirmed (sensitivity 8.3 x1014Hz/F over the full range of changes). The design can be used for a humidity sensor and can be adopted by industries due to its flexibility in design that could be beneficial from the point of view of industrial production costs.

REFERENCES

[1] F. Reverter, O. Casas, "Direct interface circuit for capacitive humidity sensors", Sensors and Actuators A, 2008, 143, 315–322.

[2] B. George, et al. (eds.), *Advanced Interfacing Techniques for Sensors*, Smart Sensors', Measurement and Instrumentation, 2017, 25. DOI: 10.1007/978-3-319-55369-6_2.

[3] N. Kuriyal, R. Kumar, V. Ramola, "Optimization and Simulation of humidity sensor readout circuitry using two stage op amp", IOSR Journal of Electrical and Electronics Engineering, 2014, 9 (5), 66-72.

[4] T. Jalkanen, A. Määttänen, E. Mäkilä, J. Tuura, M. Kaasalainen, V.P. Lehto, P. Ihalainen, J. Peltonen, J. Salonen, "Fabrication of Porous Silicon Based Humidity Sensing Elements on Paper", Journal of Sensors, 2015, Article ID 927396, 10 pages. DOI:10.1155/2015/927396.

[5] O. Nizhnik, K. Higuchi, K. Maenaka: "A Standard CMOS Humidity Sensor without Post-Processing", Sensors, 2011, 11, 6197-6202. DOI:10.3390/s110606197.

[6] P. Nath, I. Hussain, S. Dutta, A. Choudhury, "Solvent treated paper resistor for filter circuit operation and relative humidity sensing", Indian Journal of Physics, 2014, 88 (10), 1093-1097. DOI: 10.1007/s12648-014-0547-x.

[7] T.A. Blank, L.P. Eksperiandorova, K.N. Belikov, "Recent trends of ceramic humidity sensors development", Sensors and Actuators B, 2016, 228, 416-442.

[8] A. Urrutia, J. Goicoechea, A.L. Ricchiuti, V.D. Barrera, M.S. Sales, F.J. Arregui, "Simultaneous measurement of humidity and temperature based on a partially coated optical fiber long period grating", Sensors and Actuators B: Chemical, 2016, 227, 135-141.

[9] A. Mirzaei, B. Hashemi, K. Janghorban, "-Fe2O3 based nanomaterials as gas sensors", J. Mater. Sci.: Mater. Electron., 2016, 27, 3109-3144.

[10] G. Miskovic, M.D. Lukovic, M,V. Nikolic, Z.Z. Vasiljevic, J. Nicolics, O.S. Aleksic, "Analysis of electronic properties of pseudobrookite thick films with a possible application for NO gas sensing", Proceedings of the 39th International Spring Seminar on Electronics Technology, 2016, 386-391.

[11] M.V. Nikolic, Z.Z. Vasiljevic, M.D. Lukovic, V.P. Pavlovic, J. Vujancevic, M. Radovanovic, J.B. Krstic, B. Vlahovic, V.B. Pavlovic, "Humidity sensing properties of nanocrystalline pseudobrookite (Fe2TiO5) based thick films", Sensors and Actuators B: Chemical, 2018, 277, 654-664. DOI: 10.1016/j.snb.2018.09.063.

[12] L. Polak, R. Sotner, J. Petrzela, J. Jerabek, "CMOS Current Feedback Operational Amplifier-Based Relaxation Generator for Capacity to Voltage Sensor Interface, Sensors", 2018, 18 (4488), 15 pages. DOI:10.3390/s18124488.

[13] T. Islam, S.C. Mukhopadhyay, N.K Suryadevara, "Smart Sensors and Internet of Things", A Postgraduate Paper, IEEE Sensors Journal, 2017, 17 (3), 577 – 584.

[14] T. Ameloot, P.V. Torre, H. Rogier, "A Compact Low-Power LoRa IoT Sensor Node with Extended Dynamic Range for Channel Measurements", Sensors, 2018, 18 (2173), 1-16. DOI:10.3390/s18072137.

[15] Microchip PIC 16(L)F19155, Microchip Technology Inc., USA, 2017.

[16] A. Kumar, B. Chaturvedi, "Novel CMOS dual-X current conveyor transconductance amplifier realization with current-mode multifunction filter and quadrature oscillator", Circuits, Systems and Signal Processing, 2018, 37, 2250-2277.

[17] A.U. Khan, T. Islam, J. Akhtar, "An Oscillator-Based Active Bridge Circuit for Interfacing Capacitive Sensors With Microcontroller Compatibility", IEEE Trans. Instrum. Meas., 2016, 65 (11), 2560 – 2568.

[18] A.D. Amico, C.D. Natale, "A contribution on some basic definitions of sensors properties", IEEE Sensors J., 2001, 1 (3), 183–190.

[19] G.F. Fine, L.M. Cavanagh, A. Afonja, R. Binions, "Metal oxide semi-conductor gas sensors in environmental monitoring", Sensors, 2010, 10, 5469-5502. DOI: 10.3390/s10060546

[20] Evaluation of measurement data. Supplement 1 to the "*Guide to the expression of uncertainty in measurement*" - Propagation of distributions using a Monte Carlo method, BIPM, 2008.

[21] Shaw Automatic Dewpoint Meter Data Manual, SHAW Moisture Meters Ltd., Bradford, U.K., 2004.