Object Detection in Video Surveillance Based on Multiscale Frame Representation and Block Processing by a Convolutional Neural Network

Rykhard Bohusha,*, Guangdi Ma**, Yang Weichenb,***, and Sergey Ablameykok,d,****

a Polotsk State University, Novopolotsk, 211440 Republic of Belarus
b EarthView Image Inc., Huzhou, 313200 China
c Belarusian State University, Minsk, 220030 Republic of Belarus
d United Institute for Informatics Problems, National Academy of Sciences of Belarus, Minsk, 220012 Republic of Belarus

*e-mail: bogushr@mail.ru
** e-mail: magd@ev-image.com
*** e-mail: yangwch@ev-image.com
**** e-mail: ablameyko@bsu.by

Abstract—A method for detecting objects in high-resolution images is proposed that is based on representing an image as a set of its copies of decreasing scale, splitting it into blocks with overlap at each level of the image pyramid except for the top one, detecting objects in the blocks, and analyzing objects at the boundaries of adjacent blocks to merge them. The number of pyramid layers is determined by the size of the image and the input layer of the convolutional neural network (CNN). At all levels except for the top one, a block splitting is performed, and the use of overlap allows one to improve the correct classification of objects, which are divided into fragments and located in adjacent blocks. The decision to merge such fragments is made based on the analysis of the metric of intersection over union and membership in the same class. The proposed approach is evaluated for 4K and 8K images. To carry out experiments, a database is prepared with objects of two classes, person and vehicle, marked in such images. Networks of the You Only Look Once (YOLO) family of the third and fourth versions are used as CNNs. A quantitative assessment of the detection efficiency of objects is performed using the mAP metric for various combinations of parameters such as the degree of threshold confidence of the CNN and the percentage of intersection of blocks in the hierarchical representation of images. The results of the investigations are presented.

Keywords: high-resolution images, small-size objects, pyramidal representation of images, accuracy of object detection

DOI: 10.1134/S1054661822010035

INTRODUCTION

The development of digital video cameras has led to an ever-increasing use of devices that allow one to form images and videos in 4K and larger formats. The large side for 4K images can range from 3600 to 4500 pixels, and, for 8K, from 7600 to 8500 pixels [5]. Moreover, the quality of displaying the features of objects significantly increases, including small-size ones, or objects located at a considerable distance from the camera during shooting.

To detect objects, one applies a comparison with a reference set of features or uses convolutional neural networks (CNNs). The first approach involves comparing the input image with the features of previously prepared images, such as the brightness level of each pixel and features formed according to certain rules, for example, histograms of oriented gradients, the Haar features, etc. [4]. Since the choice of an optimal set of informative features for accurate description of arbitrary images is not a completely solved problem, the approach to solving applied problems requires the development and testing of adapted algorithms to form object descriptors. Therefore, in recent years, thanks to the rapid development of computing technologies, a different approach has been increasingly used, which is based on the application of pretrained CNNs on large image databases [7]. The presence of convolution operations with different types of filters and a set of layers in their architecture makes it possible to form effective feature sets whose size decreases from the input to the output of the CNN to describe images of objects with a fuzzy structure and a number of variations within a class. However, at the first stage of using a CNN, the input image is scaled to the size of the input layer, which leads to a decrease in the information content of objects, and, in case of small-
size objects, this makes impossible their detection in the image. Therefore, an urgent task is to develop methods and algorithms for detecting small-size objects in high-resolution images.

In this work, we propose a method for detecting objects in high-resolution images on the basis of pyramidal-block analysis and the use of CNNs for detecting objects in each block. This method does not impose restrictions on the CNN model and improves the detection accuracy compared to the baseline model. Examples of 4K and 8K resolution images show that the method provides an increase in the mAP for CNNs of the You Only Look Once (YOLO) family: YOLOv3 and YOLOv4.

1. EXISTING METHODS OF OBJECT DETECTION IN HIGH-RESOLUTION IMAGES

In [18], the authors proposed an approach for localizing objects in high-resolution images that is based on block processing with overlapping and the use of a CNN to distinguish areas of objects in each block. The main disadvantage of this method is the high probability of combining different objects located on the boundaries of the processed blocks into a single area and missing an object due to its fragmentation into blocks.

In [22], the authors presented an algorithm that is fed with the original image scaled to a size of $227 \times 227$. Objects are detected on the reduced copy using the AlexNet CNN. To detect smaller size objects, the original image was divided into nonintersecting blocks of sizes ranging from 600 to 75 pixels and processed by the spatial correlation network (SC-Net). The overlap in this case varied from 25 to 50% relative to the size of the blocks. For this algorithm, the authors obtained a recall of 80% on the SUN2012 database [20]. It is known that the recall metric does not allow evaluating false positives, and the database used by the authors was almost free of images with 4K resolution.

In [15], an approach for human detection was considered, in which the input image was divided into large intersecting blocks, and, when people were detected, the image was divided into smaller blocks to refine the bounding boxes of the detected regions of interest. The regions were combined based on the vertical shape of a human body. This means that the algorithm was not adapted for other classes of objects. The accuracy of the algorithm was estimated using the PEVID-UHD database [8] and a closed database containing a slightly larger number of objects. For the first dataset, the average accuracy was 90.7% and, for the second, 75.4%. However, the use of block overlap in the range from 20 to 50 pixels is not sufficient in some cases; moreover, it is not adaptive to the size of the input image.

In [17], Unel et al. proposed an algorithm for detecting small-size objects of classes “vehicle” and “person” in images with a resolution of 2K and a size of $1920 \times 1080$, obtained from small unmanned aerial vehicles for further implementation on the graphic processors of mobile devices. In this case, a PeleeNet CNN was used for block processing the original images with mutual overlap of 25%. The test results obtained on the VisDrone2018 database show that the mAP metric reaches a value of 36.67% for this algorithm. However, the authors did not assess the effect of block overlapping on the accuracy of the algorithm. The number of missing objects of class person was significantly higher compared with the class vehicle. Hence, the performance of this algorithm on detecting small-size objects is much worse even in images with resolution below 4K.

To solve the problem posed, we should distinguish the following modifications of CNN models among the well-known models differing in architecture, accuracy, and efficiency: region-based convolutional neural networks (R-CNNs) [2], residual networks (ResNet) [6], and YOLO [12]. The first version of the R-CNN model is based on preliminary selection of regions in the image, calculation of features using a CNN (for example, AlexNet), and the use of a classifier to identify objects. This model was developed in the Fast R-CNN [3] and Faster R-CNN [14] versions. Fast R-CNN architecture is aimed at reducing the number of regions in the image by applying a RoI (region of interest) polling layer. On this layer, max-polling is applied to reduce the dimension, and all regions of potential objects have the same fixed dimension. In Faster R-CNN, two steps are executed. At the first of them, one applies a deep fully connected network (region proposal network) to determine regions in the image that may contain objects. At the second step, the Fast R-CNN detector is used, which searches for objects in the proposed regions. The Faster R-CNN model is not resistant to noisy images and requires significant computational costs.

The ResNet CNN architecture uses shortcut connection to minimize the performance degradation with an increase in the number of CNN layers, provided that the accuracy limit is reached in some layer of the network. The error rate is 3.57% in the top-5 metric. In [16], the Inception-ResNet model is presented as a development of the Inception model by introducing shortcut connections, which, if necessary, allow one to skip a layer and, accordingly, nullify its effect on the result of the detector operation. This makes it possible to change the network architecture so that a finite number of layers are determined for a particular problem during training, which, in turn, allows one to reduce the error rate to 3.1% in the top-5 metric. However, the presented models are characterized by high time costs. Therefore, to solve many problems, one uses truncated versions of architectures with a reduced number of layers (for example, ResNet-34, ResNet-50, etc.), which, however, do not allow achieving the accuracy of the baseline model.
The YOLO family of CNNs refers to one-pass networks and makes it possible to localize and identify objects by a single model. Among the existing versions of these CNNs, according to the results of testing in the top-5 metric, YOLO of the third version is distinguished by high accuracy (93.8%) [13]; it uses, for feature extraction, an improved Darknet–53 architecture containing 53 layers and using 23 residual connections. The next version of this CNN is YOLOv4 [1].

The structure of YOLOv4 can be divided into three main blocks: the feature extraction block, which serves to reveal the characteristic features of objects in the input image; a block for collecting feature maps from different layers, which collects and transmits feature maps from different levels of the neural network to the detection and classification unit, which, in turn, generates output feature maps for different scales, thus allowing one to predict the coordinates of the bounding boxes and classify the contents of each cell in the input image. This structure of the YOLOv4 network is aimed at increasing the running speed and optimizing parallel computing. Accordingly, these modifications of the YOLO CNN are promising for processing high-resolution images.

2. METHOD OF OBJECT DETECTION ON THE BASIS OF MULTISCALE REPRESENTATION AND BLOCK PROCESSING BY A CONVOLUTIONAL NEURAL NETWORK

The proposed method requires a pyramidal representation of the original image as a set of its copies of decreasing scale. As one moves up the pyramid, the scale (size and resolution) of the original image decreases. At each level of the pyramid, the images are divided into blocks in which detection is performed using a CNN. To ensure high efficiency of detecting small-sized objects, it is necessary to minimize their splitting into parts before processing, that is, it is important that the object is not fragmented in any of the blocks fed to the CNN input. Therefore, overlapping blocks are used. In this case, the higher the overlap value, the larger object can be detected without being divided into parts, which, however, requires additional computational costs. After processing by the CNN, one needs a procedure for combining the regions obtained at all levels.

Based on this method, the formalization of the object detection algorithm in high-resolution images can be described as follows.

Step 1. Initialization of the original data: the size of the input image, the type of CNN used, and the size of the input layer.

Step 2. Determination of the number of levels of the image pyramid, taking into account that the dimensions of the top level should be close to the dimensions of the input layer of the CNN, according to the formula

\[ P = \left\lfloor \log_2(\max(W, H)/l) \right\rfloor + 1, \]

where \( W \) and \( H \) are the width and height of the input image, \( l \times l \) is the size of the input layer of the CNN, and \( \lfloor \cdot \rfloor \) is the nearest integer.

This approach eliminates the fragmentation of objects at the minimum scale and allows one to increase their correct classification as a whole.

Step 3. Decomposition of the image on layer \( p \) into overlapping blocks. The number of blocks is determined by the equality

\[ B_p = \left( \frac{W/p - l}{k} + 1 \right) \left( \frac{H/p - l}{k} + 1 \right) = [B_{Wp}] \cdot [B_{Hp}], \]

where \( p \) is the number of pyramid layer, \( p = 1, \ldots, P \), and \( k \) is the shift of a block (in pixels). When \( B_{Wp} \) or \( B_{Hp} \) are rounded downward, the last incomplete block is combined with the preceding block and is scaled to the size of the input layer of the CNN, while, when rounding upward, this block is filled with zeros.

The overlap of blocks can be calculated by the formula

\[ \alpha = \frac{k}{l} \times 100\%. \]

Step 4. Detection of candidate regions that may contain an object or a fragment of an object. To do this, each block is processed by the CNN, and the detected candidate region is described by the feature set

\[ F = (x_1, y_1, x_2, y_2, E, Cl), \]

where \( x_1 \) and \( y_1 \) are the coordinates of the upper left corner of the found region in the original image, \( x_2 \) and \( y_2 \) are the coordinates of the lower right corner of the found region in the original image, \( Cl \) is the class of the selected object, and \( E \) is the CNN confidence score.

Step 5. Merging candidate regions.

5.1. To merge candidate regions \( O_i \) and \( O_j \), we analyze the magnitude of their overlap at all levels and for all blocks, as well as their membership in the same class:

\[ IoU(O_i, O_j) > 0.5, \quad Cl_i = Cl_j, \]

where

\[ IoU(O_i, O_j) = \frac{O_i \cap O_j}{O_i \cup O_j}, \quad i, j \in 1, N \mid i \neq j, \]

\( N \) is the number of candidate regions.

The descriptor of the merged domain \( F(O') \) is formed as follows:
5.2. In the block processing of the detection results, it is possible that the same object or its fragments are detected with a shift of coordinates at different scales of the image pyramid or in adjacent cells. In addition, the features of a smaller fragment of the object may differ from the features of the original object, and even be closer to the descriptors of a different class. Therefore, the merging process additionally involves post-processing. To this end, it is proposed to use a rule according to which all regions obtained in the previous step are analyzed and merged:

where $d_x$ and $d_y$ are the differences between the dimensions of the candidate regions along axes $x$ and $y$, respectively, and $t_x$ and $t_y$ are threshold levels.

3. METHOD OF INVESTIGATION

To carry out experiments, we developed a technique that implements the proposed method and implies further analysis of the detected objects preliminarily marked in the input image. The preselected objects $O^s$ are described by their coordinates in the image and the membership in the classes person or vehicle. The class vehicle is a composite class, because it includes such objects as bus, car, and truck. The preparation of annotated databases of 4K and 8K images is performed using the labelImg tool [9]. Figure 1 shows an example of object annotation for an 8K image.

The main stages of the method are demonstrated in Fig. 2 and include the following: formation of an image pyramid until the dimensions of the top level approach the dimensions of the input layer of the CNN; division into overlapping blocks at all levels of the image presentation; the application of a CNN to detect objects in each block; localization of detected objects $O^d$ in frame coordinates; post-processing of the results obtained in the previous step for analyzing objects at the boundaries of blocks and describing each object by coordinates and class type. If, for previously selected and detected objects, the amount of their overlap is $IoU(O^s, O^d) > 0.5$ and the objects belong to the same class, then these objects are considered correctly detected.

The quality of the object detection algorithm is assessed using the mean average precision (mAP) metric, which implies averaging the values of the known average precision (AP) metric over classes.

At the first stage, the number of true positive $TP$ and false positive $FP$ detections is calculated based on the Intersection over Union metric for the rectangles describing the found objects and annotated on the images of the database ($IoU_{ed}$). Then the precision $p$ and recall $r$ are calculated:

$$p = \frac{TP}{TP + FP},$$

$$r = \frac{TP}{TP + FN},$$

where $FN$ is the number of omissions.

Based on the obtained values, the average precision AP is calculated for each class for eleven threshold lev-
els in the range from zero to one with a step of 0.1 by the formula

$$\text{AP} = \frac{1}{11} \sum_{i=0}^{10} \rho_{\text{int}}(r_i),$$

where $\rho_{\text{int}}(r_i) = \max p(\tilde{r}), \tilde{r} \geq r_i$.

Thus, for each of the eleven levels of $r$, we determine the interpolated value of precision, after averaging which the AP is formed. At the last stage, when calculating the mAP, the AP values obtained at the previous step are averaged for two classes of objects.

In order to more optimally use the capabilities of the CNN when detecting and identifying small-size objects, it is necessary to determine the threshold degree of confidence of the CNN and the size of the input layer, which ensure the highest performance. The results of experiments have confirmed that, to ensure high accuracy when detecting small objects in high-resolution images, it is necessary to minimize their scaling. For example, for an input layer size of $608 \times 608$, the mAP value is 0.21, while, for a size of $1024 \times 1024$, the highest value of mAP = 0.47 is achieved on a test set of images with 4K resolution, the threshold degree of confidence being $T = 50\%$.

4. RESULTS OF EXPERIMENTS

The proposed method was implemented in the Python programming language using the PyTorch machine learning framework, with the help of which the main elements of the CNN architecture were synthesized, which made it possible to conduct experiments to assess the detection accuracy. Basic operations on images are implemented using the OpenCV computer vision library. To increase the detection speed, batch processing of image blocks was performed for each pyramid level using the CUDA technology, which allows parallel processing of data by a graphic processor.

4.1. Detection of Objects in 4K Images

In total, we annotated 6049 objects of two classes, person and vehicle, with sizes from $24 \times 14$ to $308 \times 763$ in 4K images. Then, we evaluated the accuracy of the algorithm for different block overlap values and threshold levels of the CNN varying with a step of 5%. Note that, for the minimum scale of representation of the original image, the threshold value $T$ corresponds to the most effective one for the CNN; in this case, $T = 50\%$. The results of experiments are presented in Table 1. The analysis of these results shows that, for any values of $\alpha$ and $T$, the mAP of the algorithm exceeds the mAP of the CNN. The maximum gain is obtained for $\alpha = 60\%$ and $T = 75\%$, when mAP = 0.759.

Figure 3 demonstrates the fragments of processed images with resolution 4K from the prepared database. Using the algorithm, we detected nine objects (Fig. 3a); the minimum size of one of these objects was $33 \times 13$, and the application of the YOLOv3 CNN allowed us to detect an object of size $149 \times 54$ (Fig. 3b).

The performance testing of the algorithm was also performed on other 4K images that were not included in the annotated database. Figure 4a shows an example of object detection in a 4K video frame, using the YOLOv3 CNN, the input layer size $1024 \times 1024$, and $T = 50\%$. In this case, 50 objects were detected, the minimum size of the objects being $41 \times 20$ pixels.

Figure 4b shows the result of object detection by the algorithm with the parameters determined during the experiments: the size of the input layer of the CNN is $1024 \times 1024$, the number of pyramid levels is $P = 3$, $\alpha = 60\%$, and $T = 75\%$. In this case, 80 objects were detected, the minimum size being $33 \times 11$. Figure 4 also shows that the maximum gain in detection is...
achieved precisely for small-size objects, which are located in the upper and right parts of the image.

Analysis of the results of detection at different image scales (Figs. 5a–5c) and a combination of regions (Fig. 5d) for enlarged fragment 2 in Fig. 4b with three objects confirms that the use of pyramidal processing makes it possible to detect objects of different sizes at different levels and increases the accuracy of the algorithm as a whole. The YOLOv3 CNN did not detect a person in the selected fragment (Fig. 4a).

Figure 6 shows an example of merging candidate regions at the fifth step of the algorithm for

\[
\begin{array}{cccccccccccc}
\alpha & 15 & 20 & 25 & 30 & 35 & 40 & 45 & 50 & 55 & 60 & 65 \\
T 65 & 0.712 & 0.713 & 0.703 & 0.654 & 0.736 & 0.723 & 0.717 & 0.711 & 0.728 & 0.739 & 0.728 \\
70 & 0.716 & 0.718 & 0.712 & 0.664 & 0.729 & 0.744 & 0.729 & 0.723 & 0.730 & 0.742 & 0.749 \\
75 & 0.702 & 0.686 & 0.699 & 0.654 & 0.735 & 0.731 & 0.711 & 0.714 & 0.747 & 0.759 & 0.733 \\
80 & 0.678 & 0.682 & 0.674 & 0.645 & 0.732 & 0.711 & 0.709 & 0.705 & 0.740 & 0.733 & 0.727 \\
85 & 0.641 & 0.668 & 0.657 & 0.645 & 0.696 & 0.686 & 0.687 & 0.693 & 0.707 & 0.700 & 0.722 \\
90 & 0.613 & 0.639 & 0.638 & 0.605 & 0.646 & 0.643 & 0.655 & 0.646 & 0.640 & 0.699 & 0.684 \\
95 & 0.545 & 0.567 & 0.572 & 0.54 & 0.568 & 0.567 & 0.582 & 0.581 & 0.595 & 0.597 & 0.612 \\
\end{array}
\]

Fig. 3. Examples of detecting objects in images with 4K resolution (a) by the proposed algorithm and (b) by the YOLOv3 CNN algorithm.

Fig. 4. Examples of detecting objects in a video frame with 4K resolution (a) by the YOLOv3 CNN and (b) by the method proposed.
Fig. 5. Examples of detecting objects at different levels of the pyramid for an image of size 3840 × 216 (a) in the original image, (b) at the second level 1920 × 1080, (c) at the third level 960 × 540, and (d) the final result.

Fig. 6. Examples of merging candidate regions: (a) the result of detecting candidate regions at all levels of the pyramid, (b) merging of overlapping blocks (see step 5.1), and (c) the result of postprocessing (see step 5.2).

Fig. 7. Examples of detecting objects in an 8K image: (a) reduced copy of the original image, (b) on the basis of the YOLOv4 CNN, and (c) on the basis of the method proposed.
fragment 1 in Fig. 4b, which testifies to the efficiency of this stage.

After the fourth step of the algorithm, six candidate regions were found (Fig. 6a); the application of the merging procedure allowed us to correctly localize a person and improve the result for a car (Fig. 6b), and further post-processing made it possible to filter out the false positive localization of the front part of the car. The YOLOv3 CNN model did not detect a person in this region of the video frame (see Fig. 4a).

4.2. Detection of Objects in 8K Images

On the set of 8K images in [10, 11, 19], 4200 objects were marked, the smallest of them being of size 38 × 10. The YOLOv4 CNN was implemented on the PyTorch framework, and files with weighting coefficients from [21] were used. Table 2 presents the mAPs determined for various combinations of the parameters: the threshold confidence degree $T$ of the CNN and the percentage of mutual intersection of blocks in the pyramidal image representation.

Table 2 shows that, under the given conditions, the most optimal combination of parameters is $T = 70\%$ and $\alpha = 30\%$, for which the maximum value $\text{mAP}_{\text{max}} = 0.6087$ is obtained, the size of the smallest detected object being $39 \times 17$. When using YOLOv4 with an input layer of 1024 × 1024 for $T = 70\%$ and $\text{mAP} = 0.295$, the size of the smallest detected object is $44 \times 27$. Thus, the approach using a database of 8K images allows one to detect objects of smaller size and to increase the detection efficiency by more than two times.

Figure 7a demonstrates a reduced copy of an 8K video image; Figs. 7b and 7c show the results of object detection for the upper right part of this image on the basis of the YOLOv4 CNN and the proposed technique.

Using the YOLOv4 CNN, we detected 50 objects in the image (Fig. 7b) with a minimum size of 319 × 124 pixels, and, using the considered technique, we detected 132 objects (Fig. 7c), the size of the minimum of them being 42 × 36 pixels. In addition, the advantage of the proposed technique for 8K images is that it can significantly increase the detection efficiency when many objects are located close together and overlap. In this case, the features of objects will be significantly different from those obtained as a result of training the CNN, and the reduction of the 8K format of the image to the size of the input layer further reduces the information content of such objects. The applied technique, using block and hierarchical processing, allows one to preserve the original features of each object and ensures their correct detection. Figure 8 demonstrates examples of detecting closely spaced people in an 8K image.

The use of the YOLOv4 CNN made it possible to detect one person with a size of $235 \times 110$ (Fig. 8a), and the approach proposed ensured the detection of 210 objects, the minimum size being $41 \times 21$ (Fig. 8b).

Fig. 8. Results of detection (a) on the basis of the YOLOv4 CNN and (b) on the basis of the proposed method.
OBJECT DETECTION IN VIDEO SURVEILLANCE

CONCLUSIONS
We have proposed an algorithm for detecting objects in high-resolution (4K and 8K) images, which includes the following basic steps: forming an image pyramid until the dimensions of the top level approach the dimensions of the input layer of the CNN; division into overlapping blocks at all levels of the image representation; the use of a CNN to detect objects in each block; and post-processing of the results obtained in the previous step for the analysis of objects at the boundaries of blocks.

We used two architectures of the YOLO family of CNNs of the third and fourth versions to detect objects with an input layer size of 1024 × 1024 to ensure high efficiency of detecting small objects in images with resolution 4K and 8K. The proposed method has been implemented in Python using the PyTorch machine learning framework, the OpenCV computer vision library, and the CUDA technology. To carry out experiments to evaluate the effectiveness of the method, we have prepared a set of high-resolution images with annotated small and medium objects of two classes: person and vehicle. Experiments have shown that the mAP performance score can be more than doubled for the data used. Thus, the results obtained indicate that the approach under consideration is promising for solving applied problems of detecting small objects in high-resolution images. In some cases, when objects of the same class are located near each other, our algorithm may combine them by mistake. Therefore, our forthcoming work will be aimed at solving this problem.

COMPLIANCE WITH ETHICAL STANDARDS
This article is a completely original work of its authors; it has not been published before and will not be sent to other publications until the PRIA Editorial Board decides not to accept it for publication.

Conflict of Interests
The process of writing and the content of the article do not give grounds for raising the issue of a conflict of interest.

REFERENCES
1. A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” 2020, arXiv:2004.10934 [cs.CV].
2. R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Region-based convolutional networks for accurate object detection and segmentation,” IEEE Trans. Pattern Anal. Mach. Intell. 38, 142—158 (2016). https://doi.org/10.1109/TPAMI.2015.2437384
3. R. Girshick, “Fast R-CNN,” in IEEE Int. Conf. on Computer Vision (ICCV), Santiago, 2015 (IEEE, 2015), pp. 1440—1448. https://doi.org/10.1109/ICCV.2015.169
4. R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed. (Pearson Education, New York, 2018).
5. K. Goulekas, Visual Effects in a Digital World: A Comprehensive Glossary of over 7000 Visual Effects Terms, (Morgan Kaufmann, San Diego, Calif., 2001).
6. He, K., Zhang, X., Ren, S., and Sun, J., Deep residual learning for image recognition, in IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016 (IEEE, 2016), pp. 770—778. https://doi.org/10.1109/CVPR.2016.90
7. Jiang, X., Hadid, A., Pang, Y., Granger, E., and Feng X., Deep Learning in Object Detection and Recognition, (Springer, Singapore, 2019). https://doi.org/10.1007/978-981-10-5152-4
8. P. Korshunov and T. Ebrahimi, “UHD video dataset for evaluation of privacy,” in Sixth Int. Workshop on Quality of Multimedia Experience (QoMEX), Singapore, 2014 (IEEE, 2014), pp. 232—237. https://doi.org/10.1109/QoMEX.2014.6982324
9. LabelImg is a graphical image annotation tool and label object bounding boxes in images. https://github.com/tzutalin/labelImg. Cited December 20, 2020.
10. London, United Kingdom. Virtual travel. 360 video in 8K. https://www.youtube.com/watch?v=KGerjHMa90s. Cited March 16, 2021.
11. New York City 8K - VR 360 Drive. https://www.youtube.com/watch?v=2Lq86MKesG4. Cited March 12, 2021.
12. J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016 (IEEE, 2016), pp. 779—788. https://doi.org/10.1109/CVPR.2016.91
13. J. Redmon and F. Farhadi, “YOLOv3: An incremental,” 2018. arXiv:1804.02767 [cs.CV].
14. S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” IEEE Trans. Pattern Anal. Mach. Intell. 36, 1137—1149 (2017). https://doi.org/10.1109/TPAMI.2016.2577031
15. V. Ruzicka and F. Franchetti, “Fast and accurate object detection in high resolution 4K and 8K video using GPUs,” in IEEE High Performance Extreme Computing Conf. (HPEC), Waltham, Mass., 2018 (IEEE, 2018), pp. 1—7. https://doi.org/10.1109/HPEC.2018.8547574
16. C. Szegedy, S. Ioffe, V. Vanhoucke, A. A. Alemi, “Inception-v4, inception-ResNet, and the impact of residual connections on learning,” in Proc. of the Thirty-First AAAI Conf. on Artificial Intelligence, San Francisco, 2017 (AAAI Press, 2017), pp. 4278—4284.
17. F. Ö. Ünel, B. O. Özkalayci, and C. Çiğla, “The power of tiling for small object detection,” in IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, Calif., 2019 (IEEE, 2019), pp. 582—591. https://doi.org/10.1109/CVPRW.2019.00084
18. D. Vorobjov, I. Zakharova, R. Bohush, and S. Ablameyko, “An effective object detection algorithm for high resolution video by using convolutional neural network,” in *Advances in Neural Networks—ISNN2018*, Ed. by T. Huang, J. Lv, C. Sun, and A. Tuzikov, Lecture Notes in Computer Science, vol. 10878 (Springer, Cham, 2018), pp. 503–510. https://doi.org/10.1007/978-3-319-92537-0_58

19. Walk in Shinjuku, Tokyo, Japan @8K 360° VR / Sep 2020. https://www.youtube.com/watch?v=YYQufxYr-BiU. Cited March 14, 2021.

20. J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba, “Sun database: Large-scale scene recognition from abbey to zoo,” in *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, San Francisco, 2010* (IEEE, 2010), pp. 3485–3492. https://doi.org/10.1109/CVPR.2010.5539970

21. YOLO4.weights. https://drive.google.com/u/0/open?id=1cewMfuJypYWhruJRUkMwRe_b9PaT. Cited April 5, 2021.

22. L. Yongxi and T. Javidi, “Efficient object detection for high resolution images,” in *53rd Ann. Allerton Conf. on Communication, Control, and Computing (Allerton), Monticello, Ill., 2015* (IEEE, 2015), pp. 1091–1098. https://doi.org/10.1109/ALLERTON.2015.7447130

---

**Guangdi Ma.** Born in 1985. Graduated from the Chinese Academy of Surveying and Mapping in 2011. Chief Engineer of EarthView Image Inc. Scientific interests: image analysis, photogrammetry, point cloud, and oblique photography aided real 3D reconstruction.

---

**Yang Weichen.** Born in 1979. Graduated from the Jilin University, China in 2001. General manager of EarthView Image Inc. Scientific interests: image analysis, photogrammetry, geographical information systems. Pioneered the business service mode of remote sensing target recognition to assist refined social governance in China.

---

**Sergey Ablameyko.** Born in 1956, DipMath in 1978, PhD in 1984, DSc in 1990, Prof. in 1992. Professor of Belarusian State University. Scientific interests: image analysis, pattern recognition, digital geometry, knowledge-based systems, geographical information systems, medical imaging. He is in Editorial Board of *Pattern Recognition and Image Analysis, Nonlinear Phenomena in Complex Systems*, and many other international and national journals. He is Fellow of IAPR, Fellow of AAIA, Academician of National Academy of Sciences of Belarus, Academician of the European Academy, and many other academies. He was a First Vice-President of International Association for Pattern Recognition IAPR (2006–2008), Chairman of Belarusian Association for Image Analysis and Recognition.

---

**Rykhard Bohush.** Graduated from the Polotsk State University in 1997. In 2002 he received his PhD in the field of Information Processing at the Institute of Engineering Cybernetics, the National Academy of Sciences of Belarus. Head of Computer Systems and Networks Department of Polotsk State University. His scientific interests include image and video processing, object representation and recognition, intelligent systems and machine learning.