Research and Development of Deep Learning Algorithms for the Classification of Pneumonia Type and Detection of Ground-Glass Loci on Radiological Images

A. V. Emchinov,* and V. V. Ryazanov*,**

a Moscow Institute of Physics and Technology, Moscow, 115184 Russia

* e-mail: aleksandr.emchinov@phystech.edu

** e-mail: rvvccas@mail.ru

Abstract—Pneumonia is a highly dangerous state that poses serious risks to the health of a patient. In contrast to common pneumonia, lung disease COVID-19 causes a large number of lethal outcomes. The pneumonia caused by the RNA virus SARS-CoV-2 is visually hardly distinguishable from the bacterial pneumonia or inflammation caused by other viral infections. Now, COVID-19 can be diagnosed using PCR tests or X-rays of the thoracic cage. However, the results of a molecular study take a long time to prepare. In contrast, the radiological images of the thoracic cage can be obtained immediately after the radiological study. Although there exist guiding principles which help radiologists to differentiate COVID-19 from other types of infections, their assessments differ. In addition, doctors who are not radiologists can be assisted in better locating the disease, for instance, by a bounding box. Development of precise computer methods based on artificial intelligence can help medical workers in quickly determining the type of pneumonia and detecting the loci of inflammation. In this study a package of methods is developed to determine the type of pneumonia and detect the ground-glass loci using the appropriate architectures of neural networks, loss functions, augmentations at the training data generation stage, test time augmentation, and computer vision model ensembles. This task is successfully solved in the SIIM-FISABIO-RSNA COVID-19 Detection competition [17] and the proposed algorithm is in the top 10% of the best solutions.

Keywords: deep learning, neural networks, analysis of medical images, analysis of radiological images

DOI: 10.1134/S1054661822030105

INTRODUCTION

With the appearance of artificial intelligence the research area of medicine is rapidly developing. Based on the data accumulated from different devices, diagnoses can be made and even novel methods of treatment can be created. In addition, certain diseases can be prevented from spreading.

The Society for Imaging Informatics in Medicine (SIIM), The Radiological Society of North America (RSNA), and The Foundation for the Promotion of Health and Biomedical Research of Valencia Region (FISABIO) joined together to improve the quality of diagnosis of lung pneumonia. Reducing the time required to make the correct diagnosis is an important problem in medicine today. The development of machine learning-based methods can help solve this problem. The above-mentioned societies organized an international competition intended to solve this problem. The main task of the competition is to develop the computer vision algorithms to classify pneumonia and detect inflammation loci of lung tissue using radiological images of the patient.

In the past five years, a large number of solutions were developed intended to computer analysis of radiological studies, and, now, numerous approaches are known for classification, detection, and segmentation of target artefacts in radiological images [2, 5–7, 14]. In this study we merge the best practices of solving the classification and detection problems in computer vision for constructing a time-efficient and highly accurate program solution to analyze the radiological studies of the thoracic cage.

In this study we describe the solution to the problem of identifying the type of inflammation of the lung tissue and detecting the inflammation loci developed in the SIIM-FISABIO-RSNA COVID-19 Detection competition, performed on the Kaggle platform.

Our solution is based on using the modern neural network architectures EfficientNet [19] and EfficientNet-V2 [20] for the classification problem and RetinaNet [3, 10] for the detection problem. In addition, in our solution we use such techniques as augmentations (at both steps: training and inference). The algorithm described in this study solved the problem with a high
value of the mean average precision (MAP) and was awarded with a bronze medal.

1. PROBLEM FORMULATION

The task of the current paper is to develop and study precise and efficient algorithms for classification of the type of lung pneumonia and detection of target artefacts and inflammation loci. That is, we need to develop a classifier that maps the input image to the space of probabilities of classes and a detector \( D_n : \mathcal{X} \rightarrow \{\mathcal{Y}, \mathbb{R}^{C_2}\} \) that maps the input image to the pair \( \mathcal{Y} \) and \( \mathbb{R}^{C_2} \), where \( \mathcal{Y} \) is a subset of \( \mathcal{X} \) and \( \mathbb{R}^{C_2} \) is the probability of \( \mathcal{Y} \). The final algorithm is a combination of the classifier and detector:

\[
F_n = C_n \cup D_n.
\]

2. DESCRIPTION OF DATA

In this section we outline the format of initial data of the problem at hand.

The initial data used in the competition were collected from the following databases: BIMCV-COVID19 Data and MIDRC-RICORD Data [9].

The competition data were divided into 3 cohorts:
1. training set (training data);
2. validation set (public test);
3. test set (private test).

All images are stored in paths of the study/series/image type. The identifier study ID is related with predictions at the level of study. The level of the study corresponds to the problem of classification of pneumonia types. The identifier image ID is used for predictions at the image level. The image level corresponds to the problem of detection of the ground-glass loci.

The train dataset includes 6054 studies and 6334 radiological images of the thoracic cage in the DICOM format. All the images were labeled by radiologists with respect to the presence of inflammation loci. Thus, at the image level, for all radiological images (image ID), there is labeling corresponding to the ground-glass regions, bounding boxes, and, at the study level, for each study ID there is a label stating the class to which the study belongs.

The labeling is presented in the following files:
1. train_study_level.csv, a file with the labeling corresponding to the level of studies, including the correct label of the class for each study;
2. train_image_level.csv, a file with the labeling corresponding to the level of images, including the bounding box for each image and the correct label of the class.

One of the tasks of the competition was to solve the problem of classification into four classes. In the train_study_level.csv file, the following classes were represented:
1. negative for pneumonia: 1 if there is no pneumonia in the study; 0 otherwise;
2. typical appearance: 1 if such a type is detected; 0 otherwise;
3. indeterminate appearance: 1 if such a type is detected; 0 otherwise;
4. atypical appearance: 1 if such a type is detected; 0 otherwise.

A hidden test data sample has approximately the same scale as the training dataset. An example of radiological images of the presented data is shown in Fig. 1.

For each image ID (test image), it was required to predict the bounding boxes for inflammation loci and the corresponding label of the class. If there were no inflammation loci, then it was required to create the prediction ‘none 1 0 0 1 1’ (none is the class identifier corresponding to the absence of results, and this provides the bounding box with a size of one pixel with probability 1.0).

In addition to that, for each study ID we need to predict the result in the labels of the class.

The studies in the test dataset can contain more than one label:
1. negative;
2. typical;
3. indeterminate;
4. atypical.
For each study in the test set, we need to predict at least one of the labels indicated above. The format of predicting the current label is the class ID from the list given above, the probability of belonging to the class, and '0 0 1 1' is the bounding box of one pixel.

The images in the test set may contain more than one bounding box. For each object in the current test image, the class identifier 'opacity' is predicted, together with the certainty indicator and the bounding box in the 'xmin ymin xmax ymax' format. If the absence of inflammation loci is predicted, then the prediction is as follows: 'negative confidence 0 0 1 1'.

3. QUALITY CRITERION

The quality criterion of solving the competition problem was chosen to be the PASCAL VOC 2010 mean average precision (mAP) at IoU > 0.5 [4]. The prediction result is considered correct if the overlap area $a_0$ between the predicted bounding box $B_p$ and the real bounding box $B_{gt}$ is larger than 50%, determined by

$$a_0 = \frac{\text{area}(B_p \cap B_{gt})}{\text{area}(B_p \cup B_{gt})}$$

This technique of quality assessment of the trained models is widespread in the problems of detection of target artefacts in images. In this competition two problems are solved: classification and detection. However, the format of the output data at the study ID level reduces the classification problem to the detection problem, which makes it possible to use the MAP metric. This circumstance determines the chosen format of the output data for the classification problem.

4. PROPOSED METHOD

In this paper we propose a solution to the classification problem based on ensembling two neural networks: EfficientNet and EfficientNet-V2. Within our implementation of the algorithm and training of the neural networks, we applied different augmentations of images for extending the dataset and selected a suitable loss function for training the neural networks. An important stage of solving the problem is the choice of the correct strategy of cross validation. Moreover, to achieve higher values of the metric and improve the stability of the resulting algorithm, we used the test time augmentation (TTA) approach.

We propose to solve the problem of the detection of the inflammation loci of the lung tissue using the modern neural network architecture RetinaNet. In solving this subproblem, we applied different augmentations to the images to expand the initial dataset in order to avoid overfitting. The cross-validation applied to the detection problem is identical to the cross-validation from the classification problem. To achieve higher values of the metric and improve the stability of the solution, we used the approach of ensembling the results trained on different datasets. Examples of radiological studies with the drawn bounding boxes are given in Fig. 2.

4.1. Cross-Validation

We developed a strategy for performing the cross-validation. We used the GroupKFold technique, that is, a variant of the $K$-multiple iterator with nonoverlapping groups. As a group, we used the study ID, because it corresponds to a certain patient, which means that the data about the current patient must appear in the prescribed folder and only in it. Application of this technique is explained by the detection problem which requires that all the bounding boxes related to a single image to be located in each partition.
4.2. Augmentation of Original Images in the Classification Problem

Because the number of available images is limited, it is worth increasing the amount of the training data in order to make our model immune to overfitting.

To achieve this goal, we used a rather large set of augmentations [16] provided by the off-the-shelf software program Albumentations [1]. Augmentation consists in transforming the original image \( T_i : \mathcal{X} \rightarrow \mathcal{X} \). For the final transformation of the image, we apply the sequential application of augmentations:

\[
T = \circ_{i=1}^4 T_i
\]

This technique is efficient in training the neural networks of computer vision [13].

Our sequence of augmentations includes the following operations:
1. horizontal flip;
2. shift, scale, rotate;
3. hue saturation value;
4. random brightness contrast;
5. CLAHE;
6. optical distortion, grid distortion, or elastic transform;
7. Gaussian blur, Gauss noise, motion blur, or median blur;
8. Resize;
9. IAA piecewise affine;
10. IAA sharpen;
11. normalize.

Each classification model uses these augmentations at the stage of model training.

4.3. Loss Function in Classification Problem

As a loss function we decided to use the loss function based on the distribution. Such an approach provided strong convergence at the training stage and allowed achieving high values of the metric.

The loss function is a binary cross-entropy, which is prescribed by the following formula:

\[
L_{BCE}(y, \hat{p}) = -[y \log \hat{p} + (1 - y) \log (1 - \hat{p})],
\]

where \( y \) and \( \hat{p} \) are the true label and the predicted probability, respectively.

4.4. Learning Scheme of Classification Modules

To solve the problem of classifying the types of pneumonia, we train the neural networks EfficientNet and EfficientNetV2. As we have mentioned in the previous subsection, the training process is executed with the use of cross-validation. As a result, we train five sets of weights for the EfficientNet model and five sets for EfficientNet-V2. Model training is terminated in the case when ROC-AUC does not improve during five epochs in a row.

4.5. EfficientNet

The first base algorithm of the proposed ensemble is EfficientNet-B4. This architecture was chosen as one of the base models of the final algorithm, because the neural networks form the EfficientNet family reach high values of metric on the ImageNet data, as they have a low prediction time and occupy less memory. As follows from paper [19], this optimization is achieved by scaling the convolutional neural networks and solving the optimization problem. The neural network can be represented as

\[
NN = \circ_{i=1}^\infty \text{Layer}_i^K(X_{(H_i, W_i, C_i)})
\]

where \( \text{Layer}_i^K \) denotes the layer \( \text{Layer}_i \) repeated \( K \) times at step \( i \) and \( (H_i, W_i, C_i) \) denotes the dimension of the input tensor \( X \) of layer \( i \).

In contrast to the common approach used to design architectures of convolutional neural networks, which consist of finding the best architecture of layers, scaling of models is intended to vary the depth \( K \), width \( (C_i) \), and resolution \( (H_i, W_i) \) of the neural network with fixation of the layer \( \text{Layer}_i \), prescribed by the basic architecture. Thus, the problem of maximizing the quality metric \( \text{Metric} \) of our model is solved under limited computational resources:

\[
\max_{d, w, r} \text{Metric}(NN(d, w, r))
\]
where \( w, d \), and \( r \) are the coefficients for scaling the width, depth, and resolution of the neural network; \( \text{Layer}_i \), \( K_i \), \( H_i \), \( W_i \), and \( C_i \) are the prescribed parameters of the base neural network.

The number of epochs was limited by twenty, as a loss function we used the binary cross-entropy, and as an optimizer we used Adam [8]. The values of the training parameters are given in Table 1. If some parameters are absent, it is assumed that they take default values in PyTorch of version 1.7.0 [12].

The graph of the variation in the loss function during training is shown in Fig. 4. Because we train the neural network five times, we obtain five values corresponding to the loss function for training and five values for validation in the initial epochs of learning. However, from a certain time instance there may be less than five values. This is related to the fact that some of these models interrupt their training before the others when the values of the ROC-AUC metric do not improve during five epochs in a row. The variations in ROC-AUC are demonstrated in Fig. 5. All the curves in the graph are the averages of the available values. It is also fruitful to visualize the difference between the models trained on different parts of the initial data. Computation of the standard deviation with only several values is unreliable. Thus, the filled region around the curve means the interval between the minimum and maximum of the available values.

To inspect the quality of five models, we use the mean average precision metric. Its variations are given in Fig. 6. We can conclude from all the figures that these twenty epochs are sufficient to train all five models and that all models are high-quality and are not overfitted.

![Fig. 4. Evolution of loss function at stage of training EfficientNet-B4.](image)

![Fig. 5. Evolution of ROC-AUC at stage of training EfficientNet-B4.](image)

**Table 1. Training parameters for neural network with EfficientNet-B4 architecture**

| Parameter          | Value                        |
|--------------------|------------------------------|
| Kernel type        | tf_efficientnet_b4_ns       |
| Epochs             | 20                           |
| Image size         | 512 × 512                    |
| Batch size         | 16                           |
| Loss function      | BCELoss                      |
| Optimizer          | Adam                         |
| Initial learning rate | 1e-3                       |
| Learning rate scheduler | ReduceLROnPlateau    |
| Minimum learning rate | 1e-6                      |
| Patience           | 1                            |

4.6. EfficientNet-V2

In this subsection we consider the adjustment of the second base algorithm, EfficientNet-V2. Inclusion of this model in the final solution is explained by its high training rate and high values of the metric. According to paper [20], these results are reached using the concept of progressive learning, which is implemented due to the fact that the sizes of images at the start are moderate, but they increase during the training. This approach was used also earlier, but had a considerable disadvantage, because one and the same regularization was used for different sizes of images, which reduced the performance of networks; therefore, in EfficientNet-V2 the regularization dynamically increases together with the size of the image.

It means that the entire process of training occupies \( N \) steps for an image of size \( S \), with amplitudes of regularization \( \Phi = \{ \Phi_i \} \). After that, the training process is divided into \( M \) steps so that for each step \( 1 \leq i \leq M \) the model is trained on the images of the size

\[
S_i \leftarrow S_0 + (S_e - S_0) \cdot \frac{i}{M - 1}
\]
In addition, in the process of the passage of each step \(1 \leq i \leq M\), the dynamic variation in the amplitude of regularization \(\Phi_{i} = \{\phi_{i}\}\) is performed:

\[
R_{i} \leftarrow \left\{ \phi_{i} = \phi_{0} + \left( \phi_{e} - \phi_{0} \right) \cdot \frac{i}{M - 1} \right\}
\]

At each next step, the starting weights for training are the weights trained at the previous step. In addition, the scaling in the EfficientNet-V2 was implemented in a nonuniform way so that the number of layers is added at the later stages.

The training parameters for this network are presented in Table 2. The graphs of variation in the loss function are depicted in Fig. 7. The graphs of the evolution of ROC-AUC and mean average precision are plotted in Figs. 8 and 9, respectively. From the graphs we see that this architecture is trained in approximately the same manner as the previous one, concerning the values of the quality metric. However, the inclusion of it in the ensemble stabilizes and improves the quality of the test data.

**4.7. Test Time Augmentations in Classification Problem**

The data augmentations can be applied not only at the training stage but also at the prediction stage [15].

Table 2. Training parameters for neural network with EfficientNet-V2 architecture

| Parameter               | Value                        |
|------------------------|------------------------------|
| Kernel type            | \texttt{vit\_base\_r50\_s16\_384} |
| Epochs                 | 20                           |
| Image size             | \(384 \times 384\)           |
| Batch size             | 10                           |
| Loss function          | BCELoss                      |
| Optimizer              | Adam                         |
| Initial learning rate  | 1e-4                         |
| Learning rate scheduler| ReduceLROnPlateau            |
| Minimum learning rate  | 1e-6                         |
| Patience               | 1                            |
This means that the input of this technique is the predicting model of machine learning \( f : \mathcal{X} \rightarrow \mathbb{R}^C \), which matches the input tensor and the output distribution of probabilities, where \( \mathcal{X} \) is the space of input images, \( \mathbb{R}^C \) is the number of classes, and the set \( \{t_m\}_{m=1}^M \) consists of \( M \) augmentations.

Afterwards, we construct an aggregated function \( \text{agg} : \mathbb{R}^{M \times C} \rightarrow \mathbb{R}^C \). It transforms the matrix consisting of \( M \) vectors with predictions of dimension \( C \) into the vector of dimensions \( C \). In our case it is the average of predictions on transformed images with equal coefficients:

\[
\text{agg} \left( C \left( t(x_i) \right) \right) = \frac{1}{M} \sum_{m=1}^{M} C \left( t_m(x_i) \right)
\]

That is, the main idea is to predict not only the original figure but also a slightly varied one. This technique was successfully applied within the problem solved and improved the stability of the model. As an augmentation at the stage of prediction, we chose the mirror-horizontal mapping. Thus, for each image the final prediction was constructed by two images, the original and horizontally mirrored one.

4.8. Ensembling of Classification Models

To improve the stability and accuracy of the final classifying algorithm, we decided to merge our models in an ensemble, which is the average of 10 trained models (five models for EfficientNet and five models for EfficientNet-V2). In other words, the final solution for image \( x_i \) is \( S(x_i) \):

\[
S(x_i) = \frac{1}{10} \sum_{k=1}^{10} \text{agg} \left( C_k \left( t(x_i) \right) \right)
\]

The scheme of ensembling the results is presented in Fig. 10. In Table 3 we present the values of the quality MAP metric without including the results of detection on the validation and test data for base algorithms of neural networks and their composition. From the results we clearly see that using the ensemble improves the value of the metric both on the test and on the validation data. Therefore, in the final algorithm, we decided to use the scheme with averaging the results of the neural network models.

4.9. Augmentations of Original Data in Detection Problem

As we have mentioned above, in order to avoid the overfitting problem, due to the boundedness of the size of original dataset, we applied the augmentations

| Test     | Ensemble | EfficientNet | EfficientNet-V2 |
|----------|----------|--------------|-----------------|
| Public   | 0.448    | 0.443        | 0.440           |
| Private  | 0.423    | 0.423        | 0.417           |
to the original figures. The efficient transformations that we used are listed below:

1. horizontal flip;
2. shift scale rotate;
3. hue saturation value;
4. random brightness contrast;
5. CLAHE;
6. Gaussian blur;

### 4.10. Training of Detection Model

To train the detection model, we chose the program framework Detectron2 and the RetinaNet architecture. This model is included in the final algorithm as it shows good results in analyzing the radiological images [5]. This architecture is a time-efficient one-step detector, at the same time showing a high quality of prediction in the case of densely placed target artefacts. According to work [10], the accuracy of this architecture is caused by the elimination of the problem of disbalance of the background classes and the foreground by varying the loss function from the binary cross-entropy to the focal loss, which is a modification of it:

\[
FL = -(1 - p_t)^\gamma \log(p_t).
\]

The training parameters of the model are presented in Table 4.

The graphs of variation in the loss function and mean average precision are presented in Figs. 11 and 12, respectively. Because the model training is performed five times, we obtain five values, corresponding to the loss function for training and five values for validation. A similar situation is observed for the number of values of the MAP measure. All curves in the graph are the average of the available five values. In addition, in the graphs we visualized the difference between the models trained on different parts of the original data. The filled region around the curve denotes the interval between the minimum and maximum of the five values.

Using the graphs of variation in the loss function and the values of the metric, we can conclude that our models are not overfitted and are high-quality.

Thus, we trained five weights of the neural networks of the RetinaNet-101 architecture for detecting the inflammation loci of lung tissues.

### 4.11. Ensembling Detection Models

As we have mentioned in the previous subsection, formally, we obtained five trained weights of the same network architecture. To increase the values of the metric and improve the stability of the solution, we decided to use the technique of the geometric non-maximum weighted ensembling of the bounding boxes [11, 18] which fed the input of this algorithm the outputs of all five trained detection models RetinaNet. According to the work [11], we generate the final set of bounding boxes \( B \) for each image based on the prediction of all available models. We denote box = \( B_{\arg \max C_i} \), where \( C_i \) corresponds to the indicator of the reliability

### Table 4. Training parameters for RetinaNet

| Parameter                  | Value                   |
|----------------------------|-------------------------|
| Architecture               | retinanet_R_101_FPN_3x  |
| Iterations                 | 4000                    |
| Image size                 | 1024 × 1024             |
| Roi batch size per image   | 256                     |
| Images per batch           | 8                       |
| Evaluation period          | 1000                    |
| Base learning rate         | 0.0025                  |
| Learning rate scheduler    | WarmupCosineLR          |

---

**Fig. 11.** Evolution of loss function at stage of training RetinaNet.

**Fig. 12.** Evolution of MAP at stage of training on local validation of RetinaNet.
of the $i$th bounding box. The final bounding box corresponds to

$$\text{box} = \frac{\sum_{i} (w_i \cdot B_i)}{\sum_{i} w_i},$$

where $w_i = C_i \cdot \text{iou}(B_i, B_{\text{arg max}_C})$.

5. RESULTS

The algorithm we proposed won the bronze medal in the SIIM-FISABIO-RSNA COVID-19 Detection competition, being in the top-10% of all solutions.

Table 5 shows the results of the proposed solution on test data. Moreover, the values of the metric computed based on our validation scheme are close to both values from the table. This means that we have chosen the right strategies for cross-validation on the training dataset.

Our code is publicly available on the Kaggle platform (https://www.kaggle.com/greylord1996/efficientnetb4-efficientnetv2-retinanet-nmw) and GitHub (https://github.com/greylord1996/covid19_x-ray_analysis).

CONCLUSIONS

In this study we demonstrated an efficient and stable algorithm for classifying the types of lung pneumonia and detecting the inflammation loci.

The foundation of our solution for the problem of classification of the pneumonia type is an ensemble consisting of networks of architectures as EfficientNet and EfficientNet-V2. To achieve better convergence during training and to improve the performance of the algorithm, we chose the binary cross-entropy as the loss function. To avoid overfitting, we applied multiple augmentations to the data. To improve the stability of the solution and increase the value of the quality metric, we applied the test time augmentation technique and the technique of ensembling the outputs of all trained classifiers. In turn, the kernel of the solution to the detection problem was the neural network RetinaNet-101 architecture. In solving the detection problem, we also increased the size of the dataset by augmentation, and, to improve the stability, we used the non-maximum weighted strategy of geometric assembly.

ACKNOWLEDGMENTS

The authors thank the Kaggle society and organizers of the competition for the data given and the possibility of participating in the development of artificial intelligence algorithms intended to improve the quality of world health care.

COMPLIANCE WITH ETHICAL STANDARDS

This article is an entirely original work of its authors; it has not been published before and will not be sent to other publications unless the PRIA Editorial Board decides not to accept it for publication.

Conflict of Interest

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

REFERENCES

1. A. V. Buslaev, A. Parinov, E. Khvedchenya, V. I. Iglovikov, and A. A. Kalinin, “Albumentations: Fast and flexible image augmentations,” Information 11, 125 (2020). https://doi.org/10.3390/info11020125
2. E. Çallı, E. Sogancioglu, B. Ginneken, K. van Leeuwen, and K. Murphy, “Deep learning for chest X-ray analysis: A survey,” Med. Image Anal. 72, 102125 (2021). https://doi.org/10.1016/j.media.2021.102125
3. R. Del Prete, M. D. Graziano, and A. Renga, “RetinaNet: A deep learning architecture to achieve a robust wake detector in SAR images,” in IEEE 6th International Forum on Research and Technology for Society and Industry (RTSI), Naples, Italy, 2021 (IEEE, 2021), pp. 171–176. https://doi.org/10.1109/RTSI50628.2021.9597297
4. M. Everingham, S. Eslami, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, “The PASCAL Visual Object Classes challenge: A retrospective,” Int. J. Comput. Vision 111, 98–136 (2015). https://doi.org/10.1007/s11263-014-0733-5
5. T. Gabruseva, D. Poplavskiy, and A. Kalinin, “Deep learning for automatic pneumonia detection,” in IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, Wash., 2020 (IEEE, 2020), pp. 1436–1443. https://doi.org/10.1109/CVPRW50498.2020.00183
6. H. Huynh and V. Anh, “A deep learning method for lung segmentation on large size chest X-Ray image,” in IEEE-RIVF Int. Conf. on Computing and Communication Technologies (RIVF), Danang, Vietnam, 2019 (IEEE, 2019), pp. 1–5. https://doi.org/10.1109/RIVF.2019.8713648
7. Z. Karhan and F. Akal, “Covid-19 classification using deep learning in chest X-ray images,” in Medical Technologies Congress (TIPTEKNO), Antalya, Turkey, 2020 (IEEE, 2020), pp. 1–4. https://doi.org/10.1109/TIPTEKNO50054.2020.9299315
8. D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” (2017). arXiv:1412.6980 [cs.LG]
9. P. Lakhani, J. Mongan, C. Singhal, Q. Zhou, K. P. Andriele, W. F. Auffermann, P. Prasanna, T. Pham, M. Peterson, P. J. Bergquist, T. S. Cook, S. F. Ferracioli, G. C. de Antonio Corradi, M. Tákahashi, S. S. Workman, M. Parekh, S. Kamel, J. H. Galant, A. Mas-Sanchez, E. C. Benítez, M. Sánchez-Valverde, L. Jaques, M. Panadero, M. Vidal, M. Culiánuez-Casas, D. M. Angulo-Gonzalez, S. G. Langer, M. de la Iglesia Vaya, and G. Shih, “The 2021 SIIM-FISABIO-RSNA Machine Learning COVID-19 Challenge: Annotation and standard exam classification of COVID-19 chest radiographs,” OSF Preprints (2021). osf.io/532ek.

10. T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in IEEE Int. Conf. on Computer Vision (ICCV), Venice, 2017 (IEEE, 2017), pp. 2999–3007. https://doi.org/10.1109/ICCV.2017.324

11. C. Ning, H. Zhou, Y. Song, and J. Tang, “Inception single shot MultiBox detector for object detection,” in IEEE Int. Conf. on Multimedia & Expo Workshops (ICMEW), Hong Kong, China, 2017 (IEEE, 2017), pp. 549–554. https://doi.org/10.1109/ICMEW.2017.8026312

12. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “Pytorch: An imperative style, high-performance deep learning library,” in Advances in Neural Information Processing Systems; Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. Alché-Buc, E. Fox, and R. Garnett (Curran Associates, 2019), Vol. 32, pp. 8024–8035.

13. L. Perez and J. Wang, “The effectiveness of data augmentation in image classification using deep learning,” (2017). arXiv:1712.04621 [cs.CV]

14. M. Şahin, H. Ulutaş, and E. Yuçe, “A deep learning approach for detecting pneumonia in chest X-rays,” Eur. J. Sci. Technol. 28, 562–567 (2021). https://doi.org/10.31590/ejosat.1009434

15. D. Shanmugam, D. Blalock, G. Balakrishnan, and J. Guttag, “Better aggregation in test-time augmentation,” IEEE/CVF Int. Conf. on Computer Vision (ICCV), Montreal, Canada (IEEE, 2021), pp. 1194–1203. https://doi.org/10.1109/ICCV48922.2021.00125

16. C. Shorten and T. M. Khoshgoftaar, “A survey on image data augmentation for deep learning,” J. Big Data 6, 60 (2019). https://doi.org/10.1186/s40537-019-0197-0

17. SIIM-FISABIO-RSNA COVID-19 Detection Competition, 2021. https://www.kaggle.com/c/siim-covid19-detection

18. R. Solovyev, W. Wang, and T. Gabruseva, “Weighted boxes fusion: Ensembling boxes from different object detection models,” Image Vision Comput. 107, 104117 (2021). https://doi.org/10.1016/j.imavis.2021.104117

19. M. Tan and Q. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” Proc. Mach. Learn. Res. 97, 6105–6114 (2019).

20. M. Tan and Q. Le, “EfficientNetV2: Smaller models and faster training,” Proc. Mach. Learn. Res. 139, 10096–10106 (2021).

Translated by E. Oborin

Aleksandr Vladimirovich Emchinov. Born 1996. He graduated with a Bachelor’s degree from the Faculty of Control and Applied Mathematics of Moscow Institute of Physics and Technology in 2018 and a Master’s degree from the School of Applied Mathematics and Computer Sciences of Moscow Institute of Physics and Technology. Now, he is a postgraduate student of Moscow Institute of Physics and Technology. He has won awards in international competitions in machine learning on the Kaggle platform. In 2021, he received the title Kaggle Competitions Expert. His scientific interests: machine learning, neural networks, computer vision, analysis of medical images, development of computer systems intended to improve the quality of health care, international competitions on machine learning, and machine learning in problems of technological production.

Vladimir Vasil’evich Ryazanov. Born 1950. Graduated from the Moscow Institute of Physics and Technology in 1973. Received Candidate’s degree in 1977 and Doctoral degree in 1994. Full Member of the Russian Academy of Natural Sciences since 1998 and Professor since 2008. He has been working at the Dorodnicyn Computing Center of the Russian Academy of Sciences since 1976. Currently, he is the head of the Department of Classification Methods and Data Analysis of the Dorodnicyn Computing Center of the Federal Research Center Computer Science and Control of the Russian Academy of Sciences. Author of 208 papers. Scientific interests: methods for optimization of recognition models, algorithms for searching and processing logical regularities of classes according to precedents, mathematical recognition models based on voting on sets of logical regularities of classes, committee synthesis of collective clustering and construction of stable solutions in clustering problems, restoring data gaps, restoring regressions from sets of recognizing algorithms, creating software classification systems, and solving practical problems in medicine, technology, chemistry, and other areas.