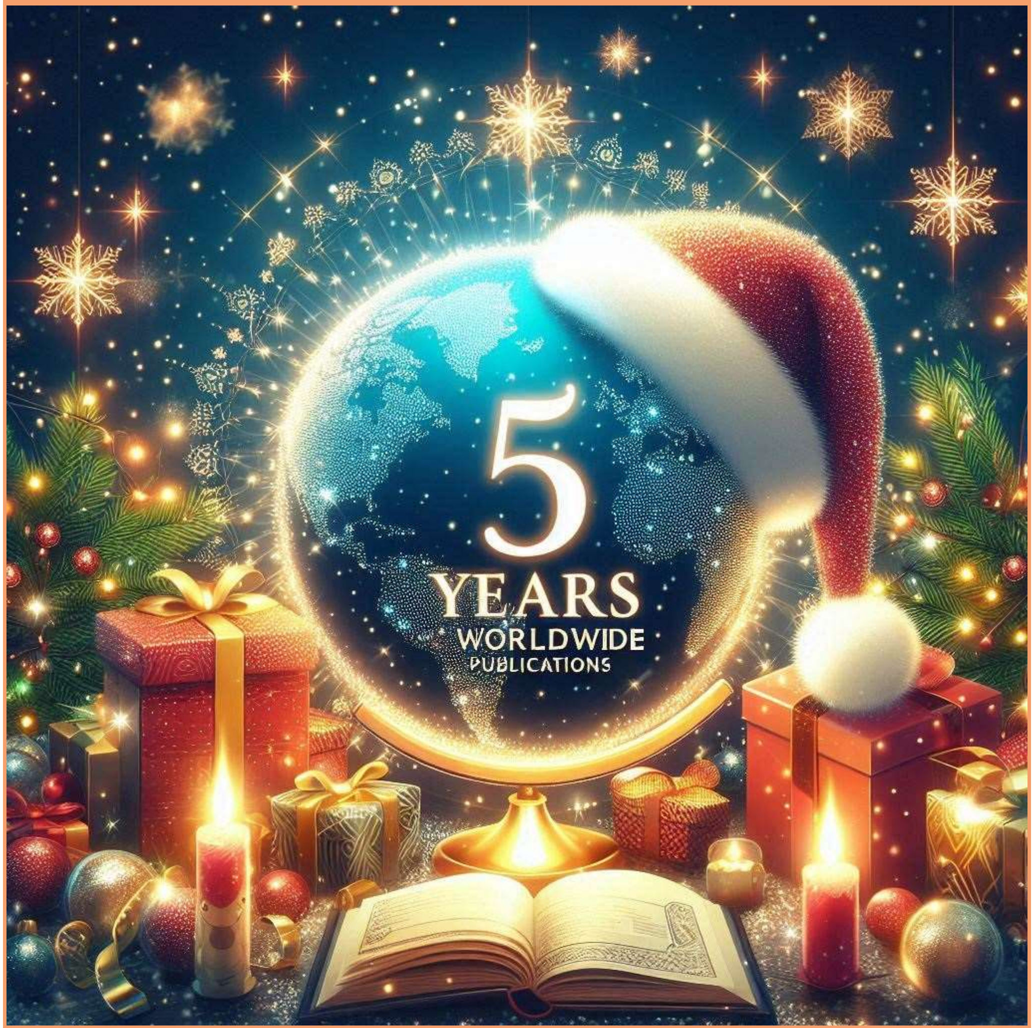
Journal of Digital Science



ISSN 2686-8296

Volume 6 Issue 2 December 2024

CONTENTS

Applying deep learning to automatically detect fly-tips in satellite imagery

Vadim Danelian [1][0009-0006-4774-4340], Andrei Kliuev [1][0000-0002-5168-5252]

[1] Perm National Research Polytechnic University, Perm, Russia, 614990

https://doi.org/10.33847/2686-8296.6.2_3

Received 25.11.2024/Revised 13.12.2024/Accepted 23.12.2024/Published 24.12.2024

Abstract. The research is dedicated to the development of neural networks for the detection of fly-tips on satellite images. The problem is relevant for Russia, where about 70 million tons of solid waste are generated annually, a significant part of which is dumped in fly-tips. Deep learning methods were used to solve two problems: binary classification of images for the presence of dumps and detection of their location. Unique datasets were collected to train the models, including more than 29,000 images for classification and 500 images for detection. The best models for classification were found to be VGG16 and VGG19 with an F1 measure of 0.91. The Faster R-CNN architecture was used for detection, achieving an accuracy of 89% on the AP metric. The results demonstrate the high effectiveness of deep learning in automating fly-tip monitoring, which helps to improve waste management control and environmental conditions in general.

Keywords: deep learning, fly-tip, satellite imagery, waste management, CNN.

1. Introduction

As you know, in Russia and in many other countries, waste management is an important environmental and social problem. This is directly related to the lack of a full-fledged recycling and disposal system, as well as to weak control over this process by state supervisory authorities. As expected, this leads to the emergence of a large number of fly-tips, which causes enormous harm to the environment.

According to the estimates of the Ministry of Natural Resources and Environment of the Russian Federation, about 70 million tons of solid waste are generated annually, which turn into fly-tips, the area of which is increasing by 500 thousand hectares per year [1]. This was of all these wastes, according to Greenpeace [2], less than 2% is burned, and about 4% is recycled. The rest of the garbage is sent either to landfills or to fly-tips, the exact number of which is unknown, which does not allow drawing conclusions about the scale of the environmental consequences. It is also worth noting that police work in these cases are not actually carried out [3].

In connection with the above, the urgency of the task of detecting, recognizing and monitoring fly-tips is obvious. At the moment, there are solutions to involve the public: special Internet maps have been created on which any user can leave a message and designate a fly-tip. Due to various satellite services, it is possible to search for illegal dumps by manually viewing satellite imagery. Obviously, both of these methods are rather laborious and ineffective. That is why the use of artificial intelligence methods, namely deep learning, seems to be the most promising in solving the problem under discussion. This approach has clear advantages. Firstly, it is the efficiency of detecting new fly-tips. Secondly, it is possible to relatively inexpensively monitor previously discovered places, track changes in the area of the object and the activity of the fly-tip (using images in the infrared range), record the moment of reclamation, etc. It should be noted that previously, if such tasks were solved, it was extremely expensive and inefficient.

The task of recognizing objects in satellite images (Earth remote sensing data) is becoming increasingly popular with the development of computer vision technologies and the growth of computing capabilities. Tasks such as recognition and segmentation of vehicles [4], ships [5], objects of agricultural activity [6] and others are already being formulated and solved. All these examples not only emphasize the relevance of using remote sensing images, but also reveal the advantages of modern computer vision methods.

For example, the recent application of deep learning methods for automated detecting marine oil spills at sea has been more successful than using traditional machine learning methods (various quality assessment metrics are given in the work) [7]. The authors presented an image segmentation model based on the Mask R-CNN that achieved an overall accuracy of 96.6% in spill detection.

Also, using convolutional neural networks (CNN), you can automate the process of annotating roads in high-resolution aerial images [8]. The authors were able to improve the results of previous researchers by using unsupervised learning methods to initialize the initial model weights and implement pre- and post-processing of images. Later, scientists in the article [9] managed to improve the quality of recognition on a dataset consisting only of Massachusetts images. It should be noted that they formulate the problem of semantic segmentation of not only roads, but also buildings.

Many researchers have been dealing with the problem of semantic segmentation of buildings. So, scientists in their work [10] give a solution method based on the use of a fully convolutional architecture (FCN). This approach, according to their estimates, surpassed the recognition efficiency of solutions based on convolutional neural networks [11] (Patch-based CNN) and on the support vector machine [12].

Some authors have shown in their works that it is a good practice to use convolutional neural networks pre-trained on everyday objects (ImageNet datasets) for pattern recognition on Earth remote sensing images [13-14].

Thus, in this work, the tasks are formulated for the development, implementation and training of neural network models to solve two closely related problems: binary classification of satellite images (into two classes – containing fly-tips and not containing ones) and detecting the desired objects – fly-tips. And the main goal is to improve the situation in the field of waste management through the use of modern digital methods and technologies.

2. Problem formulation

For the problem of binary classification (remote sensing images) the mathematical formulation is as follows. Let us introduce the notation: X is a set of instance features, Y is a set of class labels. The instance in this task is an image, and each of it's features simply represents one pixel's intensity, from 0 to 255 in the RGB color model. The set Y consists of two elements: Y={0,1}, where 0 and 1 are labels of two classes, fly-tip and not fly-tip. Suppose there is an unknown target functional dependence – some mapping $f: X \rightarrow Y$. There is also a finite set $\Xi = \{\mathbf{x}_1, \mathbf{x}_1, \dots, \mathbf{x}_m\}$, $\mathbf{x}_i \in X$, $i = \overline{1,m}$, which is called training set (dataset), and each instance is labeled with identifier: $(\mathbf{x}_i, y_i)$, $\mathbf{x}_i \in X$, $y_i \in Y$ [15]. It is possible to create a new set $(\Xi, Y) = \{(\mathbf{x}_i, y_i): i = \overline{1,m}\}$, for the elements of which the following is performed:

$$f(\boldsymbol{x}_i) = y_i, \qquad i = \overline{1,m}, \tag{1}$$

where $\mathbf{x}_i \in X, y_i \in Y$.

Thus, using a training set for which class labels are known, it is required to develop a decision rule (algorithm) – a function $g: X \to Y$, with which it is possible to classify any element $\mathbf{x} \in X$ with the least number of errors. In other words, it is necessary to construct a function g that approximates the desired f.

Let's move on to the mathematical formulation of the problem of object detection (in this case, fly-tips) from remote sensing images. For this, we introduce a two-dimensional rectangular coordinate system with the origin in the upper left corner of the image and axes directed parallel to the sides of the image: to the right and down (see Figure 1).

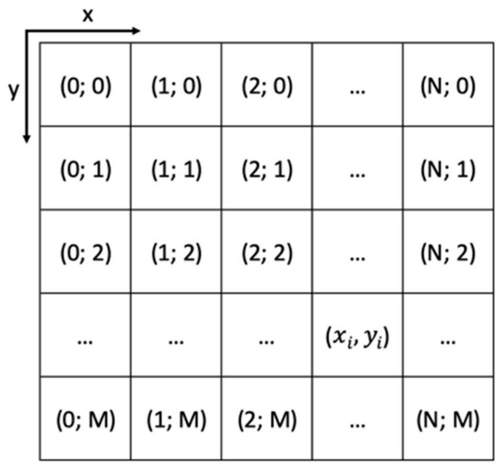


Figure 1. Coordinate system referenced to the image with N×M resolution.

In addition to defining the class, it is necessary to localize the object itself: find a rectangular area (bounding box) specified by coordinates$(x_1, y_1), (x_2, y_2)$ (upper left and lower right corners, respectively), in which the recognized object is located. For each such region, we introduce a set containing the corresponding coordinates of the image pixels: $V_i = \{(x, y): x_1 \le x \le x_2,\ y_1 \le y \le y_2\},\ i \in \mathbb{N}$, where $\mathbb{N}$ – the set of natural numbers. For a dataset of size m we will have a set $V = \{V_{ij}: i = \overline{1,m}, j = \overline{1,n_i}\}$, where n_i – is the number of annotated objects in the *i*-th image. Based on the model predictions using the g' algorithm, we obtain – $P = \left\{P_{ij}^{g'}: i = \overline{1,m}, j = \overline{1,n_i}\right\}$. It is important to note that the number of objects proposed (predictions) by the algorithm in the *i*-th image is equal to the number of annotated objects in this image.

We can calculate the similarity measure for each object using the IoU (Intersection over Union) metric, also known as the Jaccard index, using the formula:

$$IoU\left(V_{ij}, P_{ij}^{g'}\right) = \frac{\left|V_{ij} \cap P_{ij}^{g'}\right|}{\left|V_{ij} \cup P_{ij}^{g'}\right|}. \qquad (2)$$

Thus, using the training data, it is necessary to find a rule – the g^* algorithm, with which you can classify all the desired objects (fly-tips) in the image $\mathbf{x} \in X$ and determine their location so that the maximum is achieved:

$$g^* = \max_{g'} \left[\sum_{i=1}^{m} \sum_{j=1}^{n_i} IoU\left(V_{ij}, P_{ij}^{g'}\right) \right]. \quad (3)$$

3. Dataset

The correct formation of the dataset is essential in machine learning problems. In this article, the training data consists of two sets of images: containing and not containing fly-tips.

The dataset design plan can be described as follows:

1. Search for the coordinates of the location of fly-tips.
2. Translation of geographical coordinates (latitude / longitude) into tile coordinates (for convenient and efficient display of the map, its image is cut into squares of 256×256 pixels – tiles).
3. Downloading images from a satellite service (in this work – DigitalGlobe).
4. Image annotation (the process of labeling images of a dataset).

An additional task arises for image detection. For images with the desired objects, it is necessary to indicate a bounding box around each of them.

When searching for the coordinates of the location of fly-tips, the above-mentioned Internet maps were used, with the help of which it was possible to find the coordinates of more than 26 thousand unauthorized waste disposal sites. However, it should be borne in mind that not all fly-tips are visible on satellite images: trees, bushes, etc. often interfere with the view. And according to some coordinates, the fly-tips had already been reclaimed at the time of compiling the dataset, so all images had to be checked and classified manually.

The formation of negative (non-fly-tips) examples is also an important task. To effectively train the model, the dataset must contain a variety of examples of all kinds of objects found on satellite images, especially if these objects look like fly-tips. These can be urban developments, suburban areas, roads, roofs of various buildings, cars, etc.

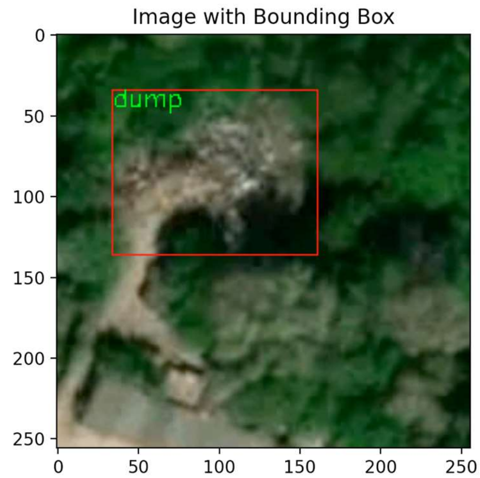


Fig. 2. An example of dump detection is shown in the image. The vertical and horizontal axes are pixel numbers.

Geographic coordinates were translated first into spherical Mercator coordinates (many mapping services use it), and then into tile numbers. After that, the images corresponding to the given tile numbers were downloaded, followed by manual classification and detection. An example of detecting the original image is shown in Figure 2.

Figure 3. Images from the training set for the classification problem. The two columns on the left are fly-tips and on the right are examples of images without fly-tips.

As a result of the work carried out, a dataset of 29913 images was formed for the classification problem: 2254 with fly-tips and 27659 – without. The data was divided into training (60% of the total), validation (20%) and test (20%) sets. It is worth paying attention to the imbalance of the dataset. On the one hand, this is necessary to achieve representativeness of the dataset, since fly-tips on satellite images are extremely rare in the real world. However, in this case, it is required to use weights for classes (in the loss function) for more successful training of models. 500 images were prepared for the detection task: 400 for training and 100 for testing. Several satellite images from the training sample for the classification problem are shown in Figure 3.

4. Experiments and results

A common approach to deep learning with a small dataset is to use a pretrained network [16]. In our case, this can be a good practice due to the inability to extract large amounts of data.

To solve the problem of binary classification, we propose to use several popular architectures of convolutional neural networks: Xception [17], VGG16 [18], VGG19 [18], InceptionV3 [19], MobileNet [20]. Each of them was pretrained on the ImageNet dataset [21]. A block of fully connected layers was added to the convolutional base of the networks using some regularization techniques (see Figure 4, L2 regularization with a value of 0.0005 was added to the dense layer).

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg19 (Functional)          (None, 8, 8, 512)         20024384

 flatten (Flatten)           (None, 32768)             0

 dense (Dense)               (None, 256)               8388864

 dropout (Dropout)           (None, 256)               0

 dense_1 (Dense)             (None, 1)                 257

=================================================================
Total params: 28,413,505
Trainable params: 27,859,969
Non-trainable params: 553,536
_________________________________________________________________
```

Figure 4. The architecture of a neural network on a convolutional base VGG19.

Binary cross-entropy was chosen as the loss function, and to assess the performance of neural networks, the following were used: accuracy, F_1-score and confusion matrix (see Fig. 5). Several layers were frozen in convolutional bases: for VGG16 and VGG19 – from layer 2 to 8, for MobileNet and Xception – from 2 to 32, and InceptionV3 – from 2 to 64. Each network was trained for 50 epochs, and the mini-batch size is 32 images. SGD with a learning rate of 0.001 and a moment of 0.9 was chosen as the optimizer. Due to the imbalance in the dataset, it was decided to use the following weights for the classes: for fly-tips – 3, for the opposite class – 1. Note that these weights do not correspond to the ratio of the number of images in the classes, since the number of images with fly-tips is about 12 times less.

		Predict	
		0	1
Actual	0	TN	FP
	1	FN	TP

Figure 5. Confusion matrix.

The results of model training are shown in Table 1.

Table 1. The results of training various networks on a test set

№	Convolutional base	F_1-score	Confusion matrix
1	VGG19	0.91	[5512 21] [57 394]
2	VGG16	0.90	[5515 18] [66 385]
3	MobileNet	0.46	[4858 675] [114 337]
4	Xception	0.24	[5531 2] [390 61]
5	InceptionV3	0.08	[5533 0] [432 19]

We can note that the networks of the VGG family were able to show the best results with a high value of the F_1-score. Both models do an excellent job of recognizing fly-tips and give few false positive responses (FP). However, the use of MobileNet, Xception and InceptionV3 has not been so successful: it may require more detailed fine-tuning of various hyperparameters to achieve any competitive results. Also note that deeper models performed worse on the task. The plot of the first training is shown in Figure 6.

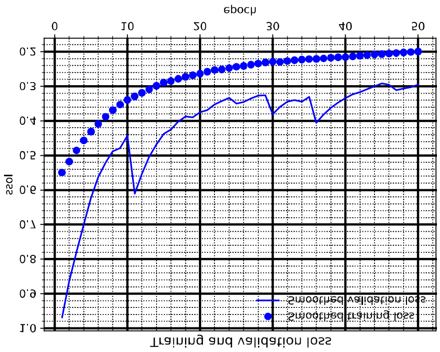


Figure 6. The value of the loss function at the stages of training and validation of the VGG19 model with the settings of the optimizer №1.

A model based on VGG19 was chosen for further research. Various optimizers were selected to train it. The results can be seen in Table 2. It can be noticed that the first two configurations showed the best results on the F_1-score.

Table 2. The results of training a network on a VGG19 convolutional base with different optimizer settings on a test set

№	Optimizer	F_1-score	Confusion matrix
1	SGD(lr=0.001, momentum=0.9), class_weight={0: 1, 1: 3}	0.91	[5512 21] [57 394]
2	SGD(lr=0.01), class_weight={0: 1, 1: 3}	0.91	[5514 19] [62 389]
3	SGD(lr=0.001, momentum=0.9)	0.88	[5471 62] [52 399]
4	RMSprop(lr=2e-5), class_weight={0: 1, 1: 3}	0.86	[5515 18] [97 354]
5	RMSprop(lr=0.001), class_weight={0: 1, 1: 3}	0.83	[5477 56] [89 362]
6	Adam(lr=0.001), class_weight={0: 1, 1: 3}	0.82	[5397 136] [40 411]
7	Adadelta(lr=0.001), class_weight={0: 1, 1: 3}	0.79	[5458 75] [109 342]
8	Nadam(lr=0.001), class_weight={0: 1, 1: 3}	0.76	[5503 30] [154 297]
9	Adam(lr=2e-5), class_weight={0: 1, 1: 3}	0.72	[5187 346] [7 444]

If you look into the confusion matrix (Figure 5), then the largest number of true positive answers in the 9th configuration (Table 2) is 444 pieces. So, if in the task it is a priority to identify as many fly-tips as possible, then this model should cope better than the rest. However, as expected, it is this network that has the largest number of false positive errors. If the goal is in the least number of these same false positives, then it is worth looking at the models numbered 1, 2 and 4. So, depending on the goals of the final product and the priorities set in the task, you can choose the appropriate model configurations.

To solve the problem of detecting fly-tips on remote sensing images, the original Faster R-CNN architecture [22] was used with several parameters changed, which needed to be adjusted to the size of the images used. The neural network responsible for feature extraction (in this case, VGG16) was pretrained on the ImageNet dataset. The training lasted 60 epoches, the results can be seen in Figure 7.

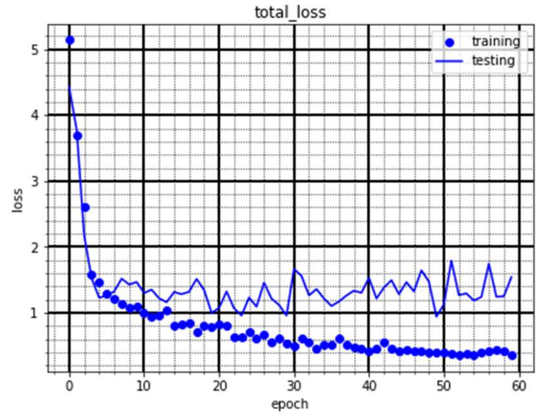


Figure 7. Total value of Faster R-CNN loss functions.

The plot of the loss function on the training data has a downward trend. However, the situation with the test set is a little different – you can notice the effect of overfitting.

Figure 8. Detection Model Predictions.

Some predictions from the test set are shown in Figure 8. It can be noted that the results are quite successful: in many cases, the network was able to predict the correct location of the fly-tip. The final detection accuracy reached 89% according to the AP metric.

5. Conclusion

The primary aim of this study was to develop and assess deep learning models for automatically detecting fly-tips in satellite imagery, using a dataset of over 29,000 images for classification and 500 samples for object detection. We tested multiple pre-trained CNN architectures (including VGG16, VGG19, MobileNet, Xception, and

InceptionV3) and found that VGG19 and VGG16 consistently delivered the strongest performance in differentiating images with fly-tips from those without (F1-score of 0.91). Furthermore, we employed a Faster R-CNN model built on the VGG16 backbone to localize the dumps, achieving 89% accuracy according to the AP metric, confirming the feasibility of using deep learning methods for pinpointing illegal dumping sites in satellite data.

These findings demonstrate that automated detection can significantly benefit agencies and organizations by expediting the identification and monitoring of unauthorized waste disposal sites, thus supporting more efficient waste management. Nonetheless, several limitations exist, such as the relatively small size of the dataset, lack of extra spectral channels, and significant computational requirements. To address these challenges, future research could expand the dataset to include broader geographic regions, integrate multispectral or hyperspectral imagery, and explore advanced architectures like transformers for improved accuracy. Additionally, incorporating temporal analysis would help track the expansion or remediation of dumping areas, making it possible to optimize resource allocation and strengthen environmental oversight.

References

1. D.N. Kobylkin, "Otkhody – v dokhody?" Argumenty i fakty (2019), available at: https://aif.ru/society/ecology/othody_v_dohody_glava_minprirody_o_tom_zachem_nuzhny_musornye_peremeny, last accessed 15 June 2019
2. "Chto delat' s musorom v Rossii?", Moskva: Greenpeace (2019), available at: https://greenpeace.ru/wp-content/uploads/2019/10/report-RUSSIA-GARBAGE.pdf, last accessed 15 November 2019
3. I. Egorov, Pogriazli: analitika, Rossiiskaia Gazeta (2018), available at: https://rg.ru/2018/06/28/chajka-nazval-regiony-s-nezakonnym-oborotom-othodov.html, last accessed 15 November 2019
4. L. Mou, X.X. Zhu, IEEE Trans. Geosci. Remote Sens., 56.11, 6699–6711 (2018)
5. Y. Feng, W. Diao, Y. Zhang, H. Li, Z. Chang, M. Yan, X. Sun, X. Gao, IEEE Int. Geosci. Remote Sens. Symp., 1025-1028 (2019)
6. K. Bhosle, V. Musande, J. Indian Soc. of Remote Sens., 47.11, 1949-1958 (2019)
7. S.T. Yekeen, A.L. Balogun, K.B.W. Yusof, ISPRS J. Photogramm. Remote Sens., 167, 190-200 (2020)
8. V. Mnih, G.E. Hinton, Proc. 11th Eur. Conf. Comput. Vision, 210-223 (2010)
9. S. Saito, T. Yamashita, Y. Aoki, Electronic Imaging, 2016.10, 1-9 (2016)
10. E. Maggiori, Y. Tarabalka, G. Charpiat, P. Alliez, Proc. IEEE Int. Geosci. Remote Sens. Symp., 5071-5074 (2016)
11. V. Mnih, Ph. D. thesis, University of Toronto (2013)
12. Y. Tarabalka, J.A. Benediktsson, J. Chanussot, IEEE Trans. Geosci. Remote Sens., 47.8, 2973-2987 (2009)
13. F. Hu, G.S. Xia, J. Hu, L. Zhang, Remote Sens., 7.11, 14680-14707 (2015)
14. O.A.B. Penatti, K. Nogueira, J.A. Dos Santos, Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. Workshops, 44-51 (2015)
15. A.E. Lepsky, A.G. Bronevich, Taganrog, Taganrogskii tekhnologicheskii institut Iuzhnogo federal'nogo universiteta, 155 (2009)
16. F. Chollet, Manning Publications (2018)
17. F. Chollet, Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 1251-1258 (2017)
18. K. Simonyan, A. Zisserman, arXiv preprint arXiv:1409.1556 (2015)
19. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2818-2826 (2016)
20. A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam arXiv preprint arXiv:1704.04861 (2017)
21. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. FeiFei, Int. J. Comput. Vis., 115.3, 211-252 (2015)
22. S. Ren, K. He, R. Girshick, J. Sun, NIPS, 28, 91-99 (2015)

Aims and Objectives

Published online by Institute of Cited Scientists, Cyprus, two times a year, Journal of Digital Science (JDS) is an international peer-reviewed journal which aims at the latest ideas, innovations, trends, experiences and concerns in the field of digital science covering all areas of the scholarly literature of the sciences, social sciences and arts & humanities.

The main goal of this journal is the effective dissemination of original incites/results generated by the human brain and presented/reflected in articles using modern information/digital technology.

Current Issue is 11th issue of Journal of Digital Science for five years of publishing. The main subjects currently covered include: Telecommunication, Cyber threats, Deep Learning, Neural Network Technologies.

Contact information

Journal URL: https://ics.events/journal-of-digital-science/
Email: conf@ics.evnets
Printed online from the original layout under the imprint at:
1, Vlachou, Nicosia, The Republic of Cyprus