sciendo RIGA TECHNICAL UNIVERSITY *Electrical, Control and Communication Engineering*
ISSN 2255-9159 (online)
ISSN 2255-9140 (print)
2018, vol. 14, no. 2, pp. 157–163
doi: 10.2478/ecce-2018-0019
https://content.sciendo.com

Appropriateness of Numbers of Receptive Fields in Convolutional Neural Networks Based on Classifying CIFAR-10 and EEACL26 Datasets

Vadim Romanuke* (*Professor, Polish Naval Academy, Gdynia, Poland*)

***Abstract* – The topical question studied in this paper is how many receptive fields (filters) a convolutional layer of a convolutional neural network should have. The goal is to find a rule for choosing the most appropriate numbers of filters. The benchmark datasets are principally diverse CIFAR-10 and EEACL26 to use a common network architecture with three convolutional layers whose numbers of filters are changeable. Heterogeneity and sensitiveness of CIFAR-10 with infiniteness and scalability of EEACL26 are believed to be relevant enough for generalization and spreading of the appropriateness of filter numbers. The appropriateness rule is drawn from top accuracies obtained on $10 \times 20 \times 21$ parallelepipeds for three image sizes. They show, knowing that the number of filters of the first convolutional layer should be set greater for the more complex dataset, the rest of appropriate numbers of filters are set at integers, which are multiples of that number. The multipliers make a sequence similar to a progression, e.g., it may be 1, 3, 9, 15 or 1, 2, 8, 16, etc. With only those multipliers, such a rule-of-progression does not give the number of filters for the first convolutional layer.**

***Keywords* – Convolutional neural networks, Convolutional layers, Filters, Performance, Receptive fields.**

I. Introduction to Choosing the Number of Filters

The convolutional layer (ConvL) of a convolutional neural network (CNN) is a set of learnable filters, which is assigned to a stride and a zero-padding [1]. A column or row of biases is appended to the set. The first two dimensions of the filter define the size of the receptive field. Generally accepted receptive fields refer to the image classification problem (ICP) considering mainly the image size and the number of image categories [2]. The number of ConvLs is determined in the similar manner. An open question is how many filters a ConvL should have, if parameters of the corresponding ICP are given.

II. Background and Motivation

Unlike the filter size, the number of filters is not restricted. Selection of the number of filters is a very specific task. It requires much experience. Until recently, it relied on setting these numbers at $h \cdot 2^n$ by $h \in \mathbb{N}$, $n \in \mathbb{N}$ [1], [3], [4]. This can be called a power-of-2 rule (P2R). Although a CNN for a large ICP is trained slowly, the training process can be sped up when the numbers of filters are chosen appropriately. Moreover, performance is expected to be significantly improved [5], [6].

If a formula (rule) for the most appropriate filter numbers is known, we can optimize more complex objects, e.g. architecture of CNN. Such a rule may be an aggregate of simple conditions under which the numbers of filters are chosen. Nevertheless, filter numbers are adjusted in a few steps for definite ICPs [2], [3], [7], [8]. Each step ends with a separate trained CNN, whose performance is gradually improved. It takes years and numerous scientific suggestions before performance in a definite ICP achieves its limit. A prominent example is the MNIST dataset, whose best error rate 0.21 % was achieved over a decade from 0.39 % [9], [10].

III. Goal and Tasks

Finding a formalized rule for choosing the most appropriate numbers of filters is an objective point. This is not a trivial goal, because it requires statistical performance data covering various versions of those numbers. To shorten the time for gathering statistics, a common CNN architecture will be selected, which is supposed to be nearly the best for a series of benchmark ICPs. This is the first task providing a constant number of ConvLs.

The second task is to define filter sizes. It is believed that the best performance is ensured when the first ConvL's filters are of size 3×3, 5×5, or 7×7 at most. Then, thirdly, the benchmark ICP is to be justified. Finally, the fourth task is to statistically determine the performance as a function of the number of filters. The function is a finite approximation of a mesh defined on an integer hyperparallelepiped (IHP). The combinations of those numbers maximizing performance are going to be extracted from this function.

IV. CNN Architecture

In machine learning and image recognition, the known datasets MNIST [10], [11], CIFAR-10 [8], [12], NORB [8], [13], and EEACL26 [8], [14], [15] are simultaneously simple and content-rich. The common CNN architecture for them has four ConvLs [8], [13], [16], [17]. To prevent overfitting and improve performance, three rectified linear units (ReLUs) and a dropout layer (DOL) are inserted:

$$C_1 \to D_1 \to R_1 \to P_1 \to C_2 \to R_2 \to P_2 \to \\ \to C_3 \to R_3 \to C_4 \to S\,, \quad (1)$$

where $\{C_i\}_{i=1}^{4}$ are four ConvLs, D_1 is the DOL, $\{R_j\}_{j=1}^{3}$ are three ReLUs, P_1 and P_2 are two maximum pooling layers

* Corresponding author.
E-mail: romanukevadimv@gmail.com

(MPLs), and S is a softmax layer. Architecture (1) is supposed to be nearly the best for those datasets [8], [16], and for other related benchmark ICPs as well [18]. CNNs (1) are trained fast enough owing to MPLs. Performance accuracy may be not so high, but the purpose is to identify the tendency at varying numbers of filters [7].

V. Sizes of Filters

Under a fixed CNN architecture, the size of a filter (receptive field) depends also on the input volume, stride, zero-padding. The ultimate simplicity is preferable, so let the stride be 1 and the zero-padding be 0. Experience demonstrates that the size of receptive fields, when moving through the CNN's layers, should not be changed much. Besides, no unit filter's size will be taken. That is the main strategy in defining the filter sizes. An example is filter sizes

$$\{5\times5,\ 5\times5,\ 3\times3,\ 3\times3\}$$

or just

$$\{5,\ 5,\ 3,\ 3\} \tag{2}$$

for 32×32 images under the CNN architecture (1). Another set

$$\{5,\ 3,\ 3,\ 4\}$$

is suitable here, but the receptive field size should not increase due to the general CNN conception of volume reduction (e.g., pooling) [1], [3], [16], [19], [20].

VI. The Benchmark ICP

There are three important requirements to benchmark ICPs, which are represented with their datasets divided into training, validation, and testing subsets:

1) sensitiveness;
2) diversity of entries;
3) medium size of images.

Sensitiveness of an ICP is understood as noticeable differentiation of performance while CNN hyperparameters vary. It is considered to see consistency and performance trends. That is why datasets like the MNIST dataset cannot be exploited – CNNs recognize the MNIST dataset handwritten digits at 99.73 % accuracy [10], so any changes of performance for MNIST are hardly noticeable.

Diversity of entries will ensure good generalization of the rule for the most appropriate numbers of filters. For this purpose, the CIFAR-10 dataset will be used, whose images are heterogeneous and miscellaneous (Fig. 1). The original CIFAR-10 dataset consists of 60 000 colour images of size $32\times32\times3$ (the third dimension appears regarding three colour channels). This dataset has 10 image categories (labelled as "airplane", "automobile", "bird", "cat", "deer", "dog", "frog", "horse", "ship", "truck"), with 6 000 images per category. There are 50 000 images intended for training and 10 000 images intended for testing (validating).

However, the ICP of 10 classes may be insufficient for satisfactory generalization and spreading of the appropriateness to other ICPs, image sizes, and the number of image categories. In addition to CIFAR-10, a dataset of enlarged English alphabet capital letters (EEACL26), making up 26 classes, will be used. EEACL26 is an infinite dataset of artificial monochrome images, and it is fully scalable — as many EEACL26 images can be generated as needed (Fig. 2). Sensitiveness of EEACL26 differs from that of CIFAR-10: CNNs are trained on EEACL26 faster and easier requiring fewer training samples. Therefore, CIFAR-10 and EEACL26 are principally different ICPs that are relevant for generalization and spreading of the appropriateness of filter numbers (Fig. 3).

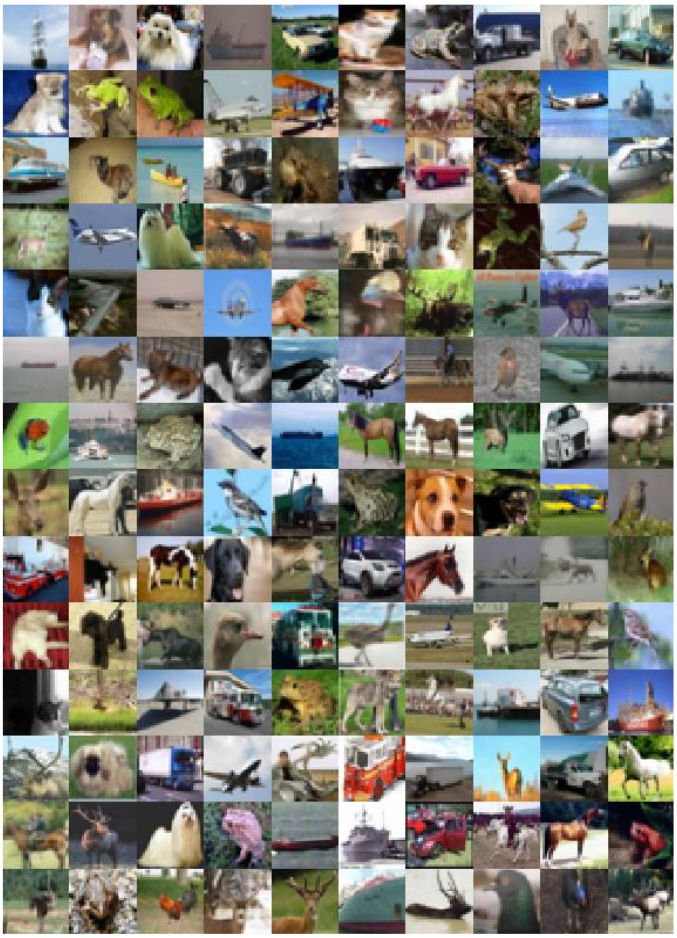

Fig. 1. Heterogeneity of colour images in a subset of the CIFAR-10 dataset [8].

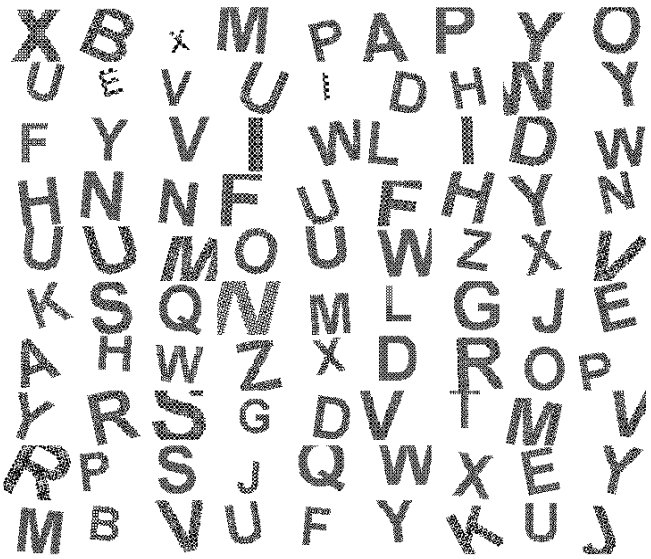

Fig. 2. A finite subset of monochrome images from the EEACL26 dataset [8].

Fig. 3. Generalization and spreading of the appropriateness of a property by an example of CIFAR-10 and EEACL26 datasets. Images of the datasets and their unlike properties are progressively intertwined, appearing more "noisy" (descending from above). This can be thought of as a more complex dataset.

The medium size of images ensures faster training owing to faster operations of convolution. This is why much bigger datasets like the ImageNet dataset cannot be effectively used [4]. The original $32\times32\times3$ images of CIFAR-10 correspond exactly to medium-sized entries. Let the size be denoted by just $N=32$. For ascertaining a relation between the number of filters and image size, another two sizes will be exploited: $N=28$ (usual for the MNIST dataset) with filter sizes

$$\{5, 3, 3, 3\} \tag{3}$$

and $N=36$ with filter sizes

$$\{5, 5, 4, 3\}. \tag{4}$$

Eventually, statistical data will be gathered by $N\in\{28, 32, 36\}$ for both CIFAR-10 and EEACL26. The ratio of the training sample to the testing and validation sample for EEACL26 is 7:3 at 36 400 images on average intended for training. There are 7 800 images intended for validation, and 7 800 images intended for testing. Thus, the whole EEACL26 sub-dataset contains 52 000 images, with 2 000 images per category. Along with sensitiveness and diversity, these parameters and factors are believed to prevent meta-overfitting of the rule for appropriateness of filter numbers. Here, the meta-overfitting is understood as adaptation of the rule to an ICP, whereas the rule loses its appropriateness for other ICPs.

VII. Performance as a Function of the Numbers of Filters

The number of filters in the last ConvL of CNN (1) is equal to the number of image categories. Therefore, the IHP is three-dimensional. Denoting the number of filters in the i-th ConvL by m_i, the performance function c_N can be stated as a function of three variables:

$$c_N = c_N(m_1, m_2, m_3). \tag{5}$$

Values of function (5) are accuracies for the respective dataset.

Ranges of variables in (5) shall enclose any admissible values of triple $\{m_i\}_{i=1}^{3}$. It is experienced that values

$$m_1=20,\quad m_2=30,\quad m_3=100$$

suit well for 28×28 images from MNIST and EEACL26. Besides, the number of filters should not decrease [1], [3], [4], [13], [20]. Hence, the ranges

$$m_1=\overline{10, 100},\quad m_2=\overline{10, 200},\quad m_3=\overline{10, 1000} \tag{6}$$

are valid with a good reserve. Of course, statistical determination of function (5) on IHP whose dimensions are derived straightforwardly from (6) is unreasonable. Function (5) can be evaluated on a subset of those

$$(100-10+1)\cdot(200-10+1)\cdot(1000-10+1)=17224571$$

points, using sampling through ranges (6). Let a step for m_1 and m_2 be 10. A step for m_3 may be broader. Let it be 50 starting off the point $m_3=50$ up to the point $m_3=1000$. Then, function (5) is to be evaluated on IHP

$$\left\{\{10k\}_{k=1}^{10}\right\}\times\left\{\{10j\}_{j=1}^{20}\right\}\times\left\{10, \{50l\}_{l=1}^{20}\right\} \tag{7}$$

which is now of 4200 points (more than 4100 times reduced).

Function (5) on IHP (7) is actually $10\times20\times21$ matrix, whose entries are determined by training and testing 4200 CNNs. The training runs through 8 epochs, which are sufficient for obtaining consistent performance and its trend (rather than perfect accuracy) [8], [18]. Visualizations of six such matrices are shown in Figs 4–9, where darker and thicker dots correspond to the higher accuracy of the CNN performance.

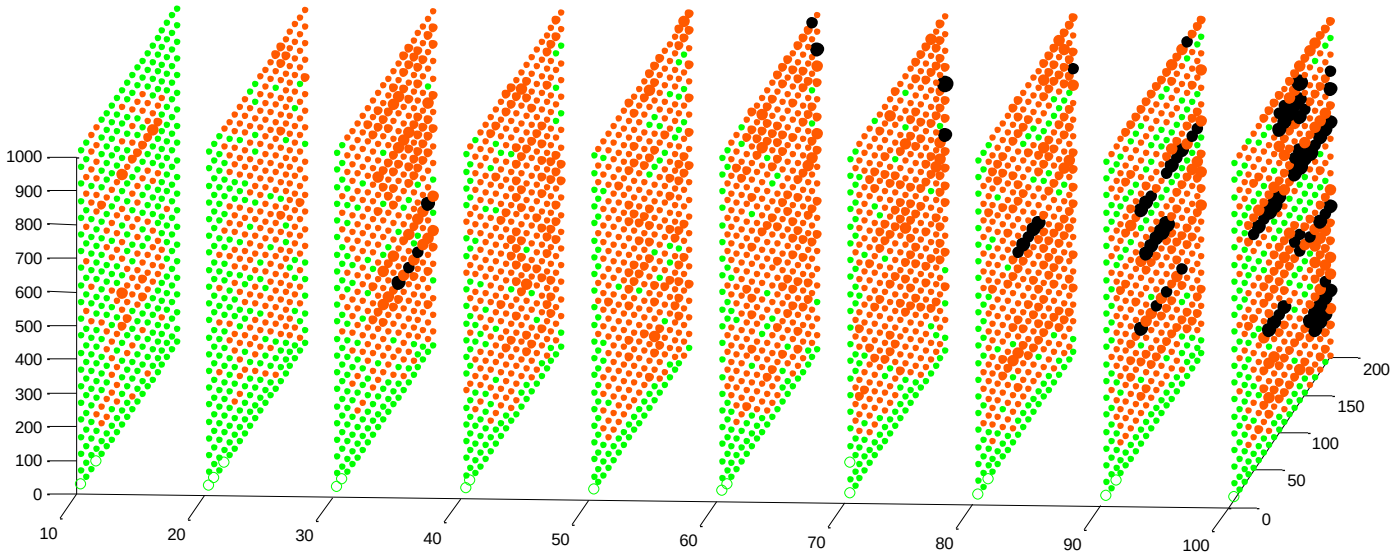

Fig. 4. Performance c_{28} on CIFAR-10 images. The circled dots stand for the (unacceptable) accuracies less than 60 %. Overall accuracy is poorer by $m_1 < 60$.

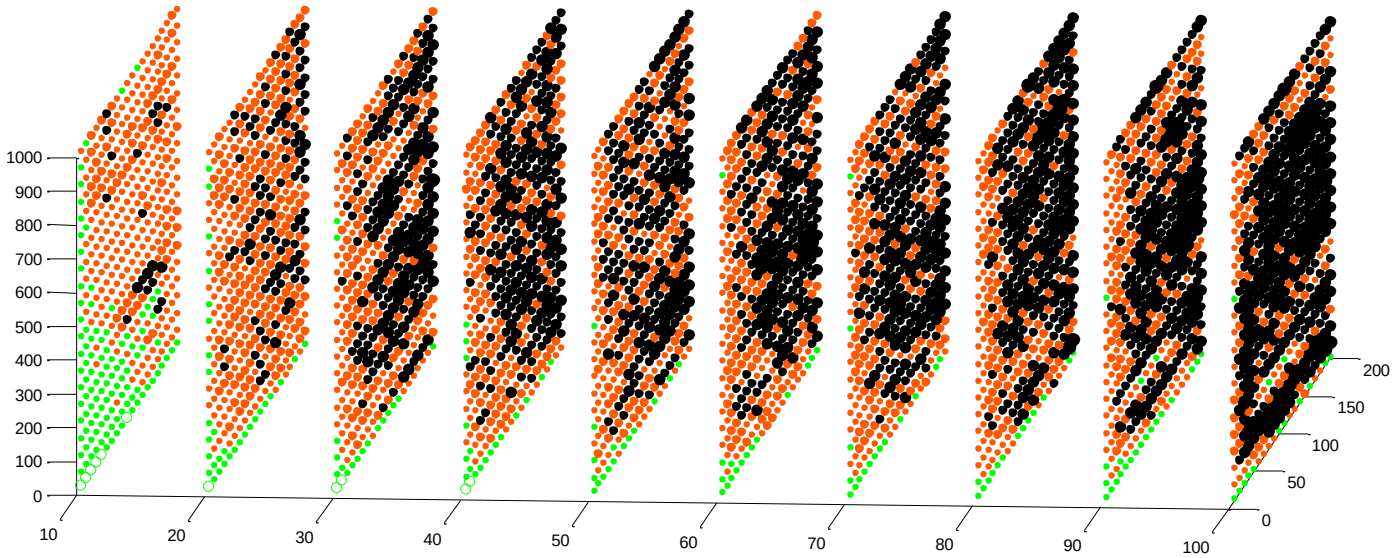

Fig. 5. Performance c_{32} on CIFAR-10 images. This is the best one. Overall accuracy is acceptable, although a few unacceptable (circled) points exist by $m_3 = 10$.

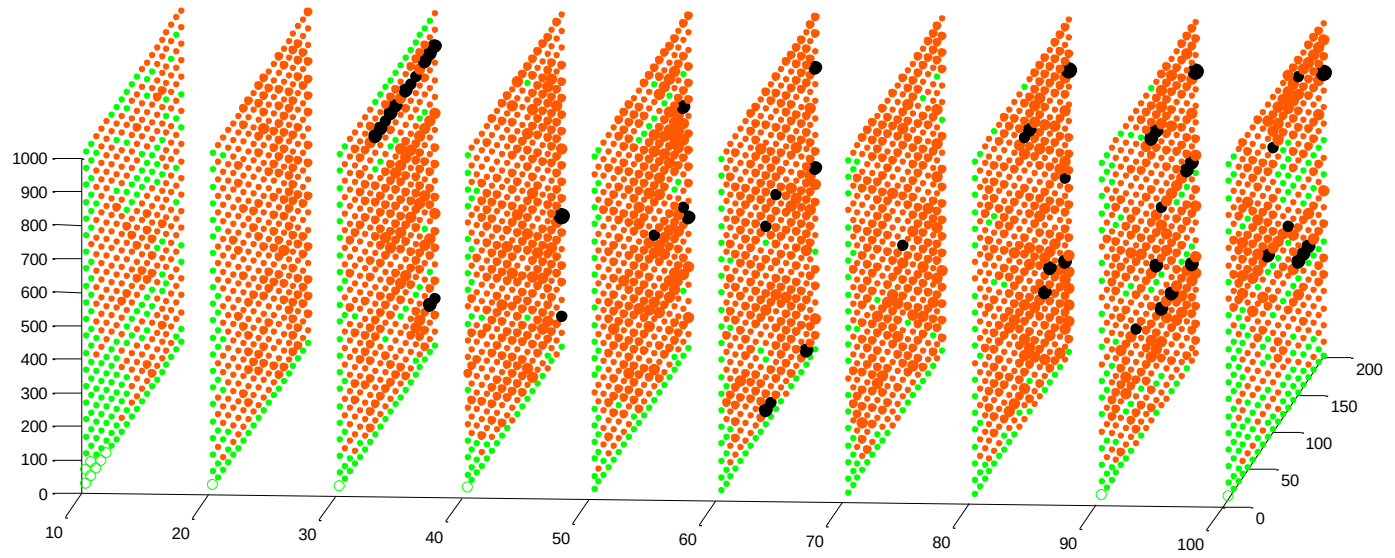

Fig. 6. Performance c_{36} on CIFAR-10 images. It is slightly better than c_{28} (CIFAR-10). Locations of the top accuracies do not seem regular but similitude exists.

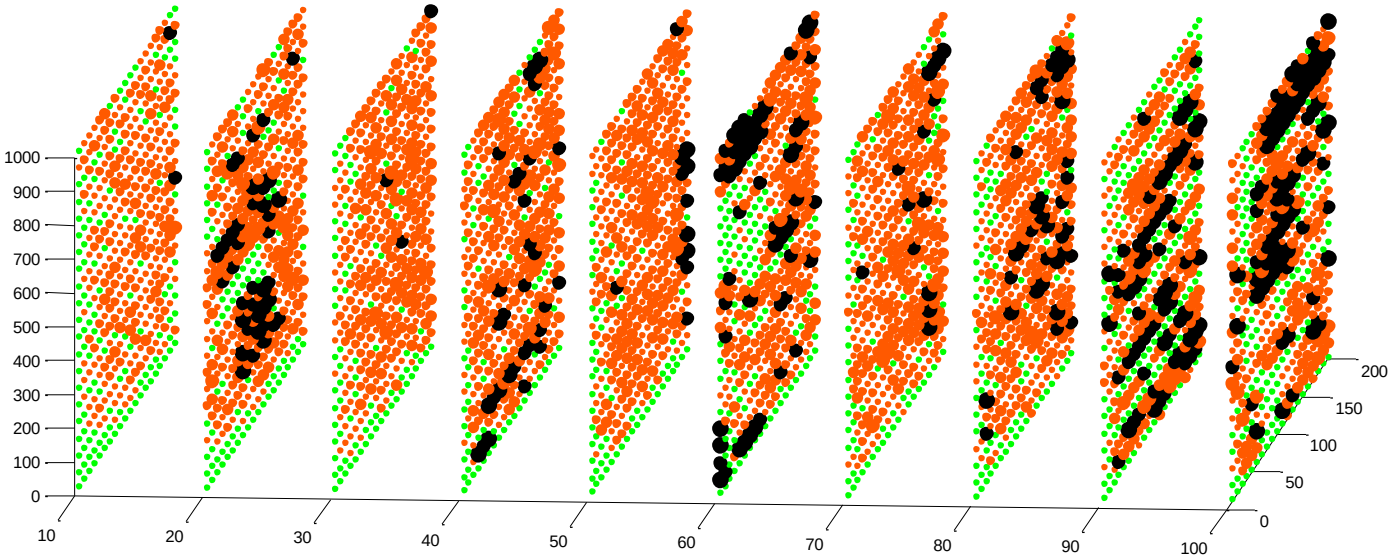

Fig. 7. Performance c_{28} on EEACL26 images. Along with the bottom points, irregular poor accuracies unexpectedly exist at the middle and upper points as well.

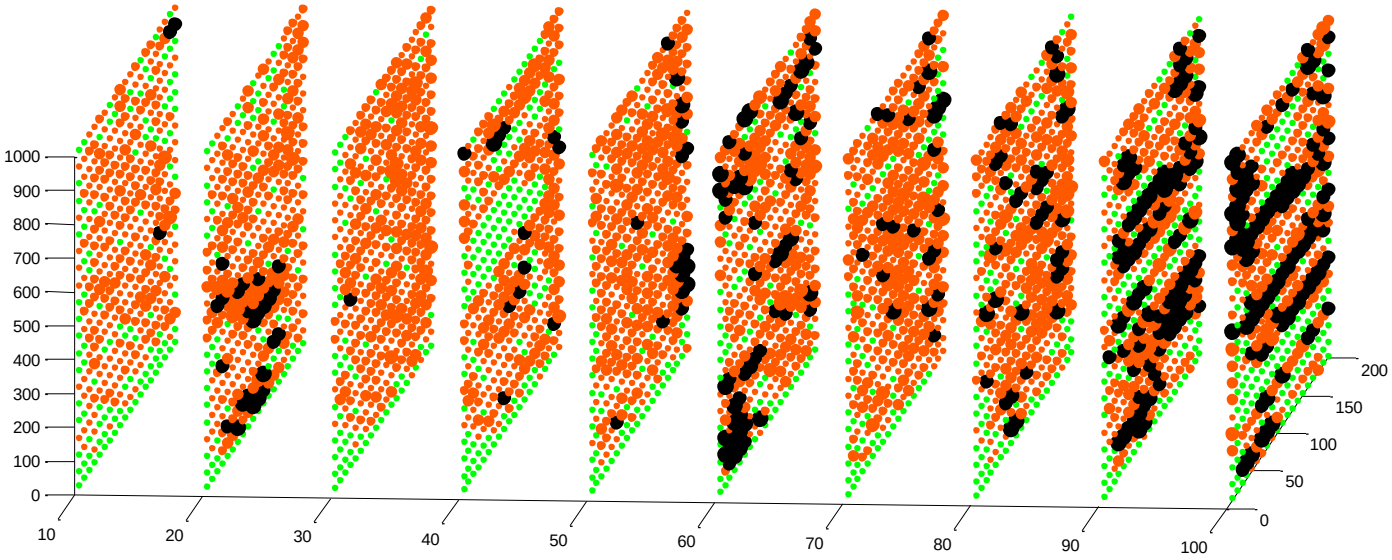

Fig. 8. Performance c_{32} on EEACL26 images roughly similar to c_{28} (EEACL26). The face at $m_1 = 60$ has the same "corners" as the respective face in Fig. 7.

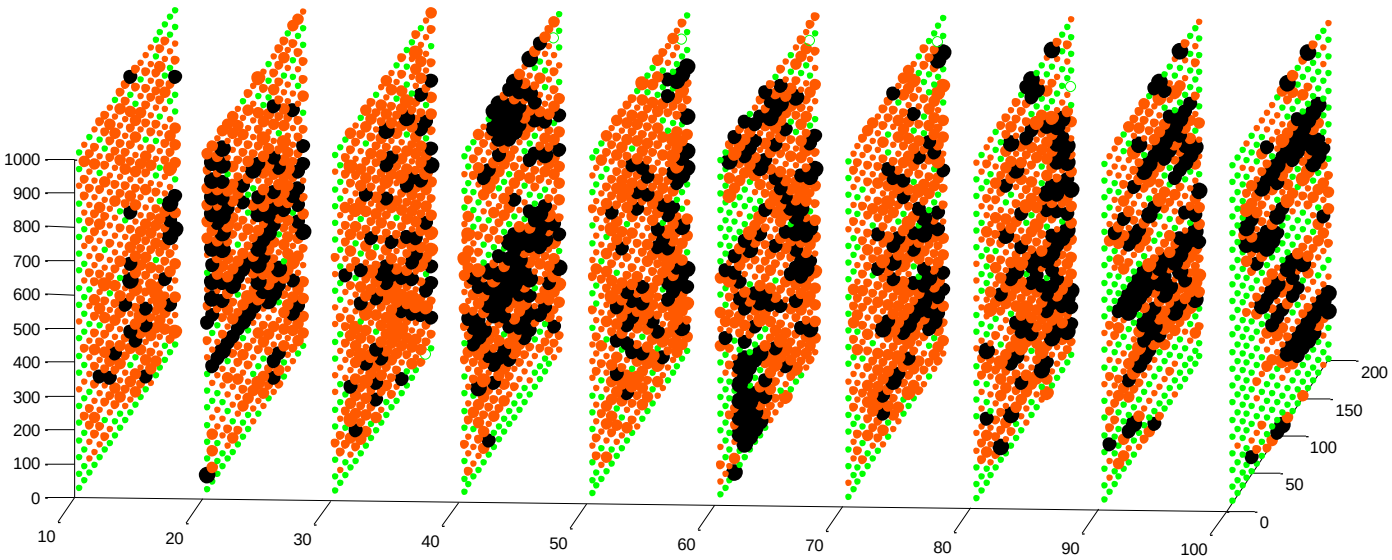

Fig. 9. Performance c_{36} on EEACL26 images, which is roughly "darker" than c_{28} and c_{32}. The quite unacceptable accuracies less than 50 % are circled dots.

Visualizations of those six three-dimensional matrices in Figs 4–9 would have been far denser if smaller steps for sampling through ranges (6) had been chosen (or just equal to 1). On the other hand, performance function (6) could have been linearly interpolated by the data in Figs 4–9. Nonetheless, this does not make sense for two main reasons. Firstly, it takes almost intractably much time and computational resources for evaluating function (5) on (the whole) IHP

$$\left\{\{k\}_{k=10}^{100}\right\}\times\left\{\{j\}_{j=10}^{200}\right\}\times\left\{\{l\}_{l=10}^{1000}\right\}$$

or on IHPs "denser" than IHP (7). Otherwise, interpolating function (5) from sparse IHP (7) is truly possible but not effective because of bad stochastic nature of the CNN performance, for both CIFAR-10 and EEACL26 datasets. Secondly, too dense face-to-face visualization of those six three-dimensional matrices would have interfered, and thus it would be impossible to discern important properties from insignificant deviations. This is why the way in which the performance results are visualized in Figs 4–9 are nearly the best to deal with and interpret.

VIII. Interpretation of the Results

It is visually quite comprehensible that performances on CIFAR-10 images differ badly from those on EEACL26 images. Faces for the EEACL26 dataset are less intelligible. The most intelligible group of faces is seen in Fig. 5, where bunches of darker dots grow and spread from the left to right, like splinters after a shot at $m_1 = 10$. However, the dark dots are not concentrated regularly. Except for $m_1 = 100$, they are rather displaced to the side of greater values of m_2. The bunch of dark dots by $m_1 = 100$ is rectangular-like and the biggest. Here the prime pretty rough inference is that the numbers of filters should be chosen appropriately as

$$20 < m_1 < m_2 < m_3 . \tag{8}$$

Although faces in Fig. 4 bear much poorer accuracies than in Fig. 5, they give precious information. Because of downsizing to $28\times28\times3$ and thus losing further the small resolution of CIFAR-10 images, CNNs perform on those downsized images with accuracies that are factually poor in a wider range. So, the first and second faces (by $m_1 = 10$ and $m_1 = 20$) are much lighter than the following faces. Moreover, an "angle bar" of poor filter numbers can be clearly seen on all faces by $m_1 \geqslant 20$ (Fig. 10). Similar "angle bars" but much fuzzier are seen in Fig. 6, wherein the slightly higher accuracies for $36\times36\times3$ CIFAR-10 images are explained with upscaling from the original images. The most prominent "angle bar" is seen on the face by $m_1 = 10$ for $32\times32\times3$ CIFAR-10 images in Fig. 5 (it is marked out in Fig. 11 with arrows). The form of this "angle bar" supports rough inference (8). This form is believed to be hidden on other faces in Figs 4–6.

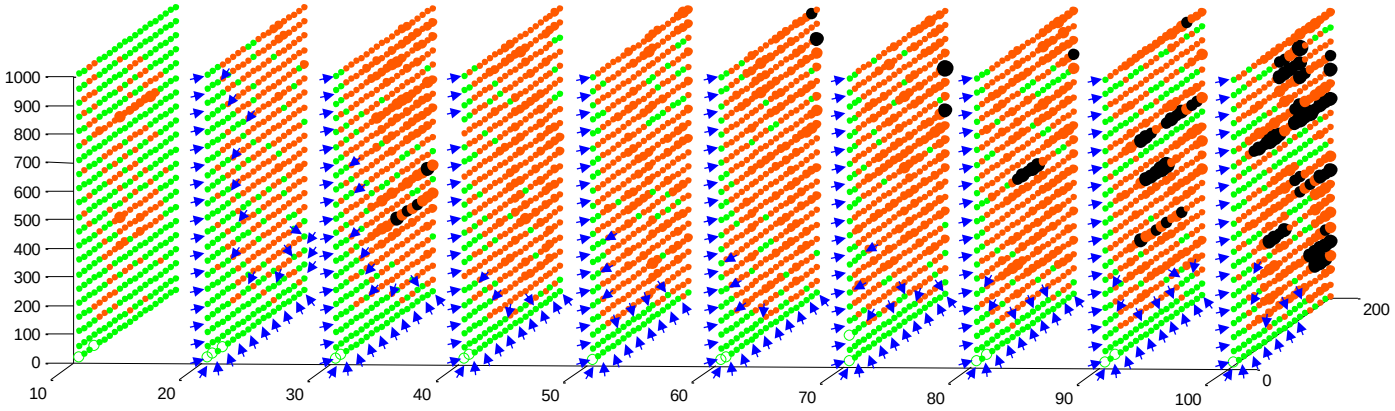

Fig. 10. "Angle bars" of poor filter numbers by performance c_{28} on CIFAR-10 images (Fig. 4) indicated with arrows. The "angle bars" by $m_1 \geqslant 90$ are fuzzy.

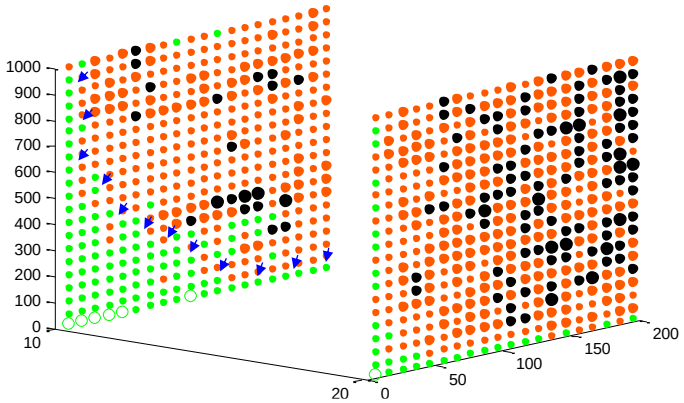

Fig. 11. The almost key "angle bar" from Fig. 5, which supports inference (8).

Faces in Figs 7–9 also contain those "angle bars", although they are not so noticeable. Generally, those faces are confusing: top accuracies are very scattered. Nevertheless, rough canon (8) is definitely confirmed.

Top accuracies for EEACL26 images are achieved when

$$35 < m_1 < 65 , \quad 105 < m_2 < 185 , \quad 425 < m_3 < 525 ,$$

without considering the image size. Apparently, top accuracies for CIFAR-10, without the image size, require more filters:

$$75 < m_1 < 100 , \quad 135 < m_2 < 200 , \quad 575 < m_3 < 925 .$$

According to these inequalities, rough canon (8) can be refined as

$$35 < m_1, \quad 3m_1 \leqslant m_2, \quad 9m_1 \leqslant m_3 \tag{9}$$

and

$$75 < m_1, \quad 2m_1 \leqslant m_2, \quad 8m_1 \leqslant m_3 \tag{10}$$

for EEACL26 and CIFAR-10, respectively. Hence, it turns out that the P2R harmonizes with inequalities (9) and (10). They are generally interpreted in a way that deeper ConvLs must be reinforced with greater numbers of filters.

IX. Conclusion and Outline of Further Research

An exact number of filters in the first ConvL is believed to correlate with parameters of the corresponding ICP, i.e. the image size and number of image categories must influence it. However, Figs 7–9 show that such correlation is too fuzzy eventually remaining unknown. It is only known that m_1 should be set greater for the more complex dataset, like CIFAR-10 compared to EEACL26. The rest of the appropriate numbers of filters should be set based on m_1: they are set at integers which are multiples of m_1, where multipliers make a sequence similar to a progression. The progression is not necessarily geometric, e.g., it may be 1, 3, 9, 15 (12, 15, 18, 21, ...) or 1, 2, 8, 16 (12, 16, 20, ...), continuing with (9) and (10). This property for filters (receptive fields) can be naively imagined as "splinters after a shot" (Fig. 5), where the reception of visual information should progress.

Such a rule-of-progression (RoP) for the most appropriate number of filters includes the P2R. Obviously, the RoP can be regarded as general (acceptable for a wide range of ICPs) owing to heterogeneity and diversity of the CIFAR-10 and EEACL26 datasets, although sizes (2), (3), (4) of receptive fields are close to each other. The research may be continued to ascertain the deepness of the progression. Besides, it must be considered whether a transition from the deepest ConvL to the final ConvL (which is a fully-connected layer) should be so abrupt. A question of inserting another fully-connected layer for smoothing this abruptness is open as well.

References

[1] H. H. Aghdam and E. J. Heravi, *Guide to convolutional neural networks: a practical application to traffic-sign detection and classification*. Cham, Switzerland: Springer, 2017. https://doi.org/10.1007/978-3-319-57550-6

[2] A. Gibson and J. Patterson, *Deep Learning: A Practitioner's Approach*. O'Reilly Media, Inc., 2017.

[3] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *Computer Vision and Pattern Recognition*, arXiv:1409.1556v6 [cs.CV], 2015.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017. https://doi.org/10.1145/3065386

[5] P. Tang, H. Wang, and S. Kwong, "G-MS2F: GoogLeNet based multi-stage feature fusion of deep CNN for scene recognition," *Neurocomputing*, vol. 225, pp. 188–197, 2017. https://doi.org/10.1016/j.neucom.2016.11.023

[6] V. Campos, B. Jou, and X. Giró-i-Nieto, "From pixels to sentiment: Fine-tuning CNNs for visual sentiment prediction," *Image and Vision Computing*, vol. 65, pp. 15–22, 2017. https://doi.org/10.1016/j.imavis.2017.01.011

[7] Z. Bai, L. L. C. Kasun, and G.-B. Huang, "Generic object recognition with local receptive fields based extreme learning machine," *Procedia Computer Science*, vol. 53, pp. 39–399, 2015. https://doi.org/10.1016/j.procs.2015.07.316

[8] V. V. Romanuke, "Appropriateness of Dropout layers and allocation of their 0.5 rates across convolutional neural networks for CIFAR-10, EEACL26, and NORB datasets," *Applied Computer Systems*, vol. 22, no. 1, pp. 54–63, 2017. https://doi.org/10.1515/acss-2017-0018

[9] M. Ranzato, C. Poultney, S. Chopra, and Y. L. Cun, "Efficient Learning of Sparse Representations with an Energy-Based Model," *Advances in Neural Information Processing Systems*, vol. 19, pp. 1137–1144, 2006.

[10] V. V. Romanuke, "Training data expansion and boosting of convolutional neural networks for reducing the MNIST dataset error rate," *Research Bulletin of the National Technical University of Ukraine "Kyiv Polytechnic Institute"*, no. 6, pp. 29–34, 2016. https://doi.org/10.20535/1810-0546.2016.6.84115

[11] E. Kussul and T. Baidyk, "Improved method of handwritten digit recognition tested on MNIST database," *Image and Vision Computing*, vol. 22, no. 12, pp. 971–981, 2004. https://doi.org/10.1016/j.imavis.2004.03.008

[12] P. Date, J. A. Hendler, and C. D. Carothers, "Design index for deep neural networks," *Procedia Computer Science*, vol. 88, pp. 131–138, 2016. https://doi.org/10.1016/j.procs.2016.07.416

[13] D. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, vol. 2, pp. 1237–1242, 2011. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-210

[14] V. V. Romanuke, "Two-layer perceptron for classifying flat scaled-turned-shifted objects by additional feature distortions in training," *Journal of Uncertain Systems*, vol. 9, no. 4, pp. 286–305, 2015.

[15] V. V. Romanuke, "Boosting ensembles of heavy two-layer perceptrons for increasing classification accuracy in recognizing shifted-turned-scaled flat images with binary features," *Journal of Information and Organizational Sciences*, vol. 39, no. 1, pp. 75–84, 2015.

[16] V. V. Romanuke, "An attempt of finding an appropriate number of convolutional layers in CNNs based on benchmarks of heterogeneous datasets," *Electrical, Control and Communication Engineering*, vol. 14, no. 1, pp. 51–57, 2018. https://doi.org/10.2478/ecce-2018-0006

[17] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

[18] V. V. Romanuke, "Appropriate number and allocation of ReLUs in convolutional neural networks," *Research Bulletin of the National Technical University of Ukraine "Kyiv Polytechnic Institute"*, no. 1, pp. 69–78, 2017. https://doi.org/10.20535/1810-0546.2017.1.88156

[19] Z. Liao and G. Carneiro, "A deep convolutional neural network module that promotes competition of multiple-size filters," *Pattern Recognition*, vol. 71, pp. 94–105, 2017. https://doi.org/10.1016/j.patcog.2017.05.024

[20] J. Mutch and D. G. Lowe, "Object class recognition and localization using sparse features with limited receptive fields," *International Journal of Computer Vision*, vol. 80, no. 1, pp. 45–57, 2008. https://doi.org/10.1007/s11263-007-0118-0

Vadim V. Romanuke was born in 1979. The higher education was received in 2001. In 2006, he received the Degree of Candidate of Technical Sciences in Mathematical Modeling and Computational Methods. The Candidate Dissertation suggested a way of increasing interference noise immunity of data transferred over radio systems. The degree of Doctor of Technical Sciences in Mathematical Modeling and Computational Methods was received in 2014. The Dissertation to be granted the degree of Doctor of Science solved a problem of increasing efficiency of identification of models for multistage technical control and run-in under multivariate uncertainties of their parameters and relationships. In 2016, he received the status of a Full Professor.

He is a Professor at the Faculty of Navigation and Naval Weapons at Polish Naval Academy. His research interests concern decision-making, game theory, statistical approximation, and control engineering based on statistical correspondence. Vadim Romanuke has good programming skills in MATLAB. For practical implementations, Vadim Romanuke uses Python.

Address for correspondence: 69 Śmidowicza Street, Gdynia, Poland, 81-127.

E-mail: romanukevadimv@gmail.com

ORCID ID: https://orcid.org/0000-0003-3543-3087