Computer Methods and Programs in Biomedicine 209 (2021) 106318

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

TEM virus images: Benchmark dataset and deep learning classification

Damian J. Matuszewski[a,*], Ida-Maria Sintorn[a,b]

[a] Department of Information Technology, Uppsala University, Uppsala, Sweden
[b] Vironova AB, Gävlegatan 22, Stockholm, Sweden

ARTICLE INFO

Article history:
Received 11 May 2021
Accepted 24 July 2021

Keywords:
CNN
Convolutional neural networks
Transmission electron microscopy
Virus recognition
Transfer learning
Dataset curation

ABSTRACT

Background and Objective: To achieve the full potential of deep learning (DL) models, such as understanding the interplay between model (size), training strategy, and amount of training data, researchers and developers need access to new dedicated image datasets; i.e., annotated collections of images representing real-world problems with all their variations, complexity, limitations, and noise. Here, we present, describe and make freely available an annotated transmission electron microscopy (TEM) image dataset. It constitutes an interesting challenge for many practical applications in virology and epidemiology; e.g., virus detection, segmentation, classification, and novelty detection. We also present benchmarking results for virus detection and recognition using some of the top-performing (large and small) networks as well as a handcrafted very small network. We compare and evaluate transfer learning and training from scratch hypothesizing that with a limited dataset, transfer learning is crucial for good performance of a large network whereas our handcrafted small network performs relatively well when training from scratch. This is one step towards understanding how much training data is needed for a given task.
Methods: The benchmark dataset contains 1245 images of 22 virus classes. We propose a representative data split into training, validation, and test sets for this dataset. Moreover, we compare different established DL networks and present a baseline DL solution for classifying a subset of the 14 most-represented virus classes in the dataset.
Results: Our best model, DenseNet201 pre-trained on ImageNet and fine-tuned on the training set, achieved a 0.921 F1-score and 93.1% accuracy on the proposed representative test set.
Conclusions: Public and real biomedical datasets are an important contribution and a necessity to increase the understanding of shortcomings, requirements, and potential improvements for deep learning solutions on biomedical problems or deploying solutions in clinical settings. We compared transfer learning to learning from scratch on this dataset and hypothesize that for limited-sized datasets transfer learning is crucial for achieving good performance for large models. Last but not least, we demonstrate the importance of application knowledge in creating datasets for training DL models and analyzing their results.

1. Introduction

Deep learning (DL) shows great promise in various biomedical and microscopy image analysis applications as shown e.g. in these recent reviews on deep learning in medical image analysis [1], in medical and biomedical (pathology) applications [2], in image (microscopy) based cell analysis [3], and in different aspects of electron microscopy, also including biomedical applications [4]. However, for these methods to achieve their full potential and gain acceptance in a clinical and diagnostic setting, the knowledge and understanding of how and when they can be trusted need to be increased. Machine learning researchers and application specialists need access to representative image datasets, to fulfill in other image domains, the amazing results achieved on natural scene images and videos from the internet which is currently driving the machine learning field. Datasets from different imaging techniques and application domains are required to identify problems and biases with current methods and commonly used benchmark datasets and applications. That is, annotated images representing real-world problems with all their real imperfections and complexity such as variations in imaging condition, noise, sample preparation, limitation in size, etc. are desirable not only for the application itself but also for theoretical method development [5,6].

* Corresponding author.
E-mail address: damian.matuszewski@it.uu.se (D.J. Matuszewski).

https://doi.org/10.1016/j.cmpb.2021.106318

Electron microscopy applications are just a tiny part of the reported biomedical deep-learning-related research. The recent review [4], shows the potential of deep learning in the many various aspects relating to electron microscopy imaging and applications (not only in biomedicine), such as denoising, superresolution, segmentation, image classification, protein structure determination, etc. It also contains a section about TEM datasets and states: "…of which most are small, esoteric and not partitioned for machine learning." Recently, the same author made two large TEM and STEM image datasets with partitions for machine learning publicly available [7]. The images of highly varied content and for different purposes and applications were acquired within research projects at Warwick University. In addition, EM imagery was also used in some of the seminal papers of deep learning in the biomedical field [8,9] with data from the ISBI 2012 challenge [10,11].

In the field of automatic virus recognition in transmission electron microscopy (TEM) images, there is only a handful of published papers. Many of which come from our group [12,13,14,15,16] and some from others [19,20,21,22]. This low number of publications is likely due to the shortage of publicly available image datasets suitable for machine learning. Datasets or databases of single or a few TEM images of viruses and virus-infected cells as diagnostic reference/support exists for example from the Robert Koch Institute [23], but for datasets of a size and design suitable for developing and evaluating machine learning purposes, only the KylbergDataset [24] is publicly available to the best of our knowledge. It consists of cut-outs of viruses resampled to a fixed size, so it is useful for a local pattern or texture analysis but not for semantic segmentation or combined segmentation and recognition methods. This lack of datasets for automating the recognition process is somewhat surprising considering the proven importance of manual TEM as a rapid and reliable virus diagnostic tool in emergencies to take adequate measures [25], as well as to identify the virus in emerging or zoonotic virus outbreaks or bioterror attacks. The importance of TEM, and the risk of new and emerging viruses slipping through the nets of molecular detection, was for example experienced in a previous SARS-CoV outbreak [26], and the Monkeypox outbreak in the US, in 2003, where the viruses were identified using TEM [27]. The current outbreak of SARS-CoV2 will likely lead to an increased interest also in image-based automated diagnosis.

In this paper, we present and make freely available an annotated transmission electron microscopy (TEM) image dataset. The dataset contains images of 22 virus classes along with extracted image patches centered on virus particles. Although parts of this dataset have been used in previously published research [12,13,14,15,16,28], this is the first time the whole virus image collection is made publicly available. This after careful curation and preparation/partitioning into a benchmark dataset suitable for machine learning method development and evaluation. Moreover, we present DL classification results on virus image patches for a handcrafted small network and some of the top-performing (large and small) networks for the 14 virus classes with the highest number of virus particles in the dataset. We compare and evaluate transfer learning and training from scratch hypothesizing that with a limited dataset, transfer learning of a large network may be a better choice than training a custom network from scratch if the real-world deployment situation allows for a larger network. This is one step towards understanding how much training data is needed for a task. We also compare different data splits (which images are used for training and testing) and show that variation in the dataset split in an unfortunate (or correct) way may impact the results. This highlights the importance of not only machine learning domain expertise but also application domain expertise for understanding and correct interpretation of results and method (DL) performance.

2. Materials and methods

2.1. Context virus dataset

The dataset contains in total 1245 images of 22 virus classes captured with two different electron microscopes: an LEO (Zeiss, Oberkochen, Germany) with a Morada (Olympus) camera and a Tecnai 10 (FEI, Hillsboro, OR, USA) with a MegaView III (Olympus, Münster, Germany) camera. Before imaging, all samples were treated with 10% phosphate-buffered saline, placed on carbon-coated TEM grids, and stained with 2% phosphotungstic acid following standard procedures. As mentioned in the introduction, parts of the dataset have been used in previous publications. Here, we make the full dataset available with manual annotations after having cleaned the dataset by removing overlapping images.

The virus classes in the dataset are strongly unbalanced both regarding the number of images (from 9 to 129) and in the number of virus particles (from 38 to 1934). The sizes of all images are either 1376 × 1032 or 2048 × 2048 pixels (depending on with which electron microscope they were captured) but the pixel sizes vary from 0.26 to 5.57 nm, i.e., they were acquired at different magnifications.

Each virus particle is annotated only with its approximate center, i.e., a single point for isolated spherical particles or a centerline in case of clustered viruses (beyond visual recognition of individual particles) or elongated virus. The annotations are in the form of coordinate points stored in a separate text file for each image. The virus image dataset is challenging due to many reasons: limited annotation (a center point/line and not a full segmentation mask), a relatively small number of images per class, diffuse virus boundaries, imperfect focus, noise, different magnifications, and a large variation of the virus, background, and debris appearance.

In our previous study [16], we observed that when dealing with virus classes represented by relatively few images with a highly varied number of particles per image, it is very important to make sure that the images are carefully split between training, validation, and test sets to make the sets as representative as possible. Therefore, before splitting the image dataset, we first ordered the images in each class by the number of virus particles they contained. Next, we assigned the images in a repeating sequence to the training, test, training, validation, and training sets until there were no more images left for that class. This resulted in a split of images roughly corresponding to 60-20-20 % for training, validation, and test sets, respectively. In this way, each set received some images with few particles and some with many particles of a given class. Table 1 presents the number of particles and images in each virus class and the corresponding number selected for the training, validation, and test sets.

2.2. Data preprocessing

Many CNNs require input images to be of the same size. Therefore, before training our CNN models, we decided to rescale and crop the original images. We used Lanczos-3 kernel interpolation to rescale all images so that their pixel size corresponds to 1 nm. Next, we cropped the images to patches of 256 × 256 pixels (256 × 256 nm) around the annotation points: the manually selected center points for spherical virus particles and all center-line vertices for elongated virus particles, except for the elongated particles that were annotated with a center-line composed of only 2 points (this would usually indicate an oval-shaped virus particle such as e.g. Orf) – in this case, we selected a new midway point for the center of the patch. This resulted in more image patches than particles in virus classes with elongated particles, particularly in Marburg, Ebola, Influenza, Lassa, and Nipah. E.g., the test set contains only 24 Marburg virus particles but there are 173 patches

Table 1
TEM virus dataset. The virus types used in the classification are marked in bold.

Virus	av. particle size [μm]	# images				# particles			
		Train	val.	test	Total	train	val.	test	Total
Adenovirus	**80**	**40**	**13**	**14**	**67**	**160**	**44**	**86**	**290**
Astrovirus	**25**	**46**	**15**	**15**	**76**	**266**	**86**	**66**	**418**
CCHF	**120**	**44**	**15**	**15**	**74**	**249**	**100**	**86**	**435**
Cowpox	**270**	**30**	**10**	**10**	**50**	**189**	**70**	**57**	**316**
Dengue	45	19	6	7	32	72	15	44	131
Ebola	**80**	**65**	**22**	**22**	**109**	**173**	**70**	**60**	**303**
Guanarito	140	13	4	4	21	31	10	7	48
Influenza	**110**	**56**	**19**	**19**	**94**	**365**	**158**	**125**	**648**
Lassa	**140**	**62**	**21**	**21**	**104**	**222**	**92**	**80**	**394**
LCM	120	22	7	8	37	31	10	14	55
Machupo	120	20	6	7	33	69	16	23	108
Marburg	**80**	**57**	**19**	**19**	**95**	**78**	**28**	**24**	**130**
Nipah	**95**	**25**	**8**	**8**	**41**	**75**	**22**	**17**	**114**
Norovirus	**30**	**32**	**11**	**11**	**54**	**231**	**104**	**84**	**419**
Orf	**145**	**38**	**13**	**13**	**64**	**92**	**76**	**31**	**199**
Papilloma	**55**	**19**	**6**	**6**	**31**	**693**	**227**	**187**	**1107**
Pseudocowpox	145	20	7	7	34	46	21	16	83
Rift Valley	**90**	**77**	**26**	**26**	**129**	**1140**	**408**	**386**	**1934**
Rotavirus	**80**	**22**	**7**	**7**	**36**	**169**	**48**	**40**	**257**
Sapovirus	30	8	3	3	14	19	11	8	38
TBE	50	25	8	8	41	39	12	11	62
WestNile	50	5	2	2	9	88	66	34	188
Total		***745***	***248***	***252***	***1245***	***4497***	***1694***	***1486***	***7677***

(Marburg viruses are very elongated). We used mirror padding for cropping patches that would partially lay outside the original image border. As many annotation points were placed relatively close to each other (due to natural clustering of the virus particles and/or complex shapes of the elongated particles), the patches cropped from the same image sometimes overlap with each other to some degree. However, this did not lead to a data leakage between the training, validation, and test sets because they were established at the image level and special care was taken to remove images from the dataset that captured the same virus particles (i.e., overlapping images).

The virus classes in this dataset are strongly unbalanced in the number of virus particles and, thus, the number of extracted image patches. Therefore, we augmented the training set image patches by flipping and multiple 90 degrees rotations so that each class contains 736 input samples. Those classes that originally contained more than 736 patches were randomly reduced to this number. Finally, before passing the images to the models we normalized their intensities by subtracting the mean from each image patch and dividing by the standard deviation.

We make the following variants of the virus image dataset publicly available at [17]:

1) raw images – the original images from the TEMs with the corresponding annotations and metadata,
2) image patches – cropped from images rescaled to 1 nm per pixel,
3) selected classes – the collection of selected virus classes used in the classification research presented in this paper; including the augmented training set.

All four datasets are split into train, validation, and test sets as described above. Fig. 1 presents sample image patches of all virus classes in the dataset. For the classification problem that we present in the rest of this paper, we excluded the 8 least populous classes (in terms of either image or particle number) from the dataset. Too few samples would likely not represent all possible data variations within these classes and thus be insufficient for good generalization and effective training.

2.3. Convolutional neural networks

We designed a custom-made network inspired by ResNet [29]. It is composed of 4 residual blocks of double convolutional layers with batch normalization, max-pooling, and dropout, followed by a classification block composed of the final convolutional layer, and 3 fully connected layers with dropout between the first two layers. Fig. 2 presents our CNN architecture.

In addition, we fine-tuned some of the most popular networks via transfer learning. The framework architecture and the list of the used CNNs are presented in Fig. 2. All networks were pre-trained on ImageNet [30], and we replaced their original top layers with 3 fully connected layers with interposed dropouts. Some networks (the best-performing ones in the different tests) were also trained from scratch, i.e., not using transfer learning.

We trained the models from scratch for 300 epochs and the fine-tuned (transfer learning) models for 50 epochs. We used the categorical cross-entropy loss function and early stopping based on the validation performance. The models achieved the best performance after around 250 and 30 epochs for the training from scratch and fine-tuning, respectively.

All models were implemented using TensorFlow [31] and Keras [32]. Our code and the best-trained models are available here [18].

3. Virus recognition benchmarking

The top part (above the double line) of Table 2 presents the performance scores of our models on the test set. Besides the classification accuracy, we report precision, recall, and F1-score that were averaged across all classes in the test dataset regardless of the number of instances in each class. Thus, we weigh each class equally, and hence, do not hide errors in a problematic but small class. Figs. 3 and 4 present the test set confusion matrices of our custom architecture model and the best model – DenseNet201 transferred from ImageNet, respectively.

To investigate and demonstrate the sensitivity of data split strategy on the performance, we trained and tested our custom model and the two top-performing models from the first experiment on data that was split into the training, validation, and test sets in the worst possible manner with respect to the representativeness of particle number per image. That is, we sorted the im-

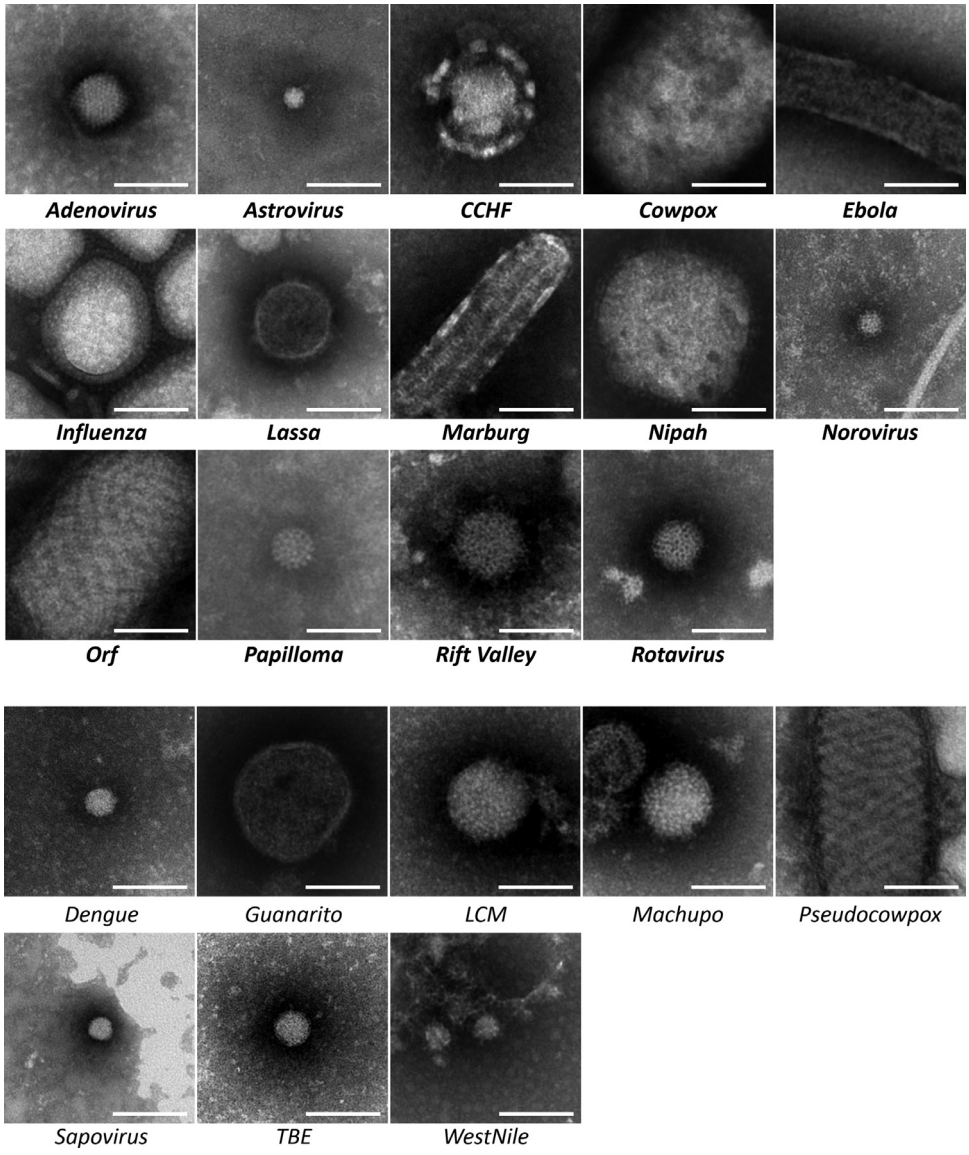

Fig. 1. Example virus patches from the training dataset. The scale bars are 100 nm. The two bottom rows present virus classes that were not used in the classification tests reported in this paper.

Table 2
Deep Learning performance.

Model	Trainable params	Accuracy	Precision	Recall	F1-score
Custom (from scratch)	14,651,401	0.901	0.895	0.898	0.893
DenseNet201 [33] (transfer)	**144,987,022**	**0.931**	**0.926**	**0.921**	**0.921**
DenseNet201 (from scratch)	144,987,022	0.811	0.775	0.821	0.791
InceptionV3 [34] (transfer)	98,330,798	0.895	0.887	0.894	0.889
MobileNetV2 [35] (transfer)	87,174,926	0.846	0.803	0.849	0.813
ResNet50 [29] (transfer)	158,817,294	0.898	0.883	0.881	0.877
ResNet50V2 [36] (transfer)	158,802,062	0.887	0.862	0.885	0.869
VGG16 [37] (transfer)	49,334,094	0.923	0.906	0.916	0.908
VGG19 [38] (transfer)	54,643,790	0.918	0.909	0.924	0.912
Xception [39] (transfer)	156,089,654	0.895	0.887	0.900	0.891
Custom (from scratch) – bad data split	14,649,867	0.765	0.751	0.619	0.640
VGG16 (transfer) – bad data split	49,334,094	0.832	0.822	0.726	0.737
VGG16 (from scratch) – bad data split	49,334,094	0.678	0.604	0.560	0.559
DenseNet201 (transfer) – bad data split	144,987,022	0.805	0.794	0.705	0.711
DenseNet201 (from scratch) – bad data split	144,987,022	0.689	0.639	0.556	0.558

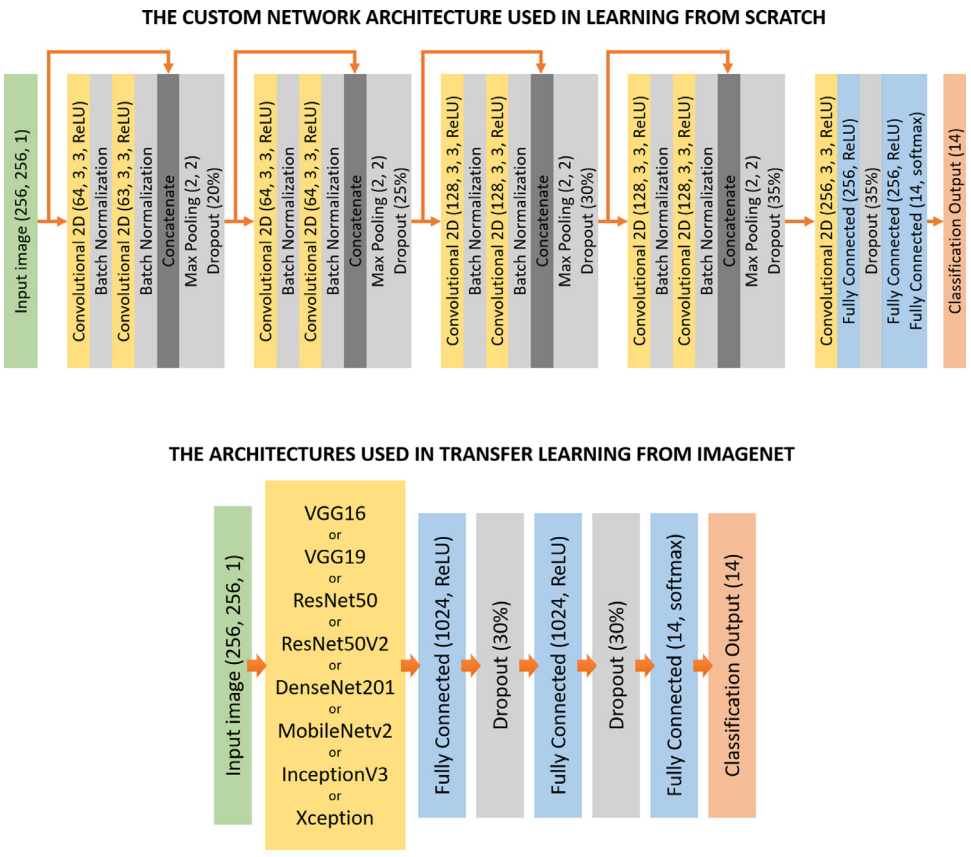

Fig. 2. The custom CNN architecture (top) and the established network architectures (bottom) that were used in the virus classification.

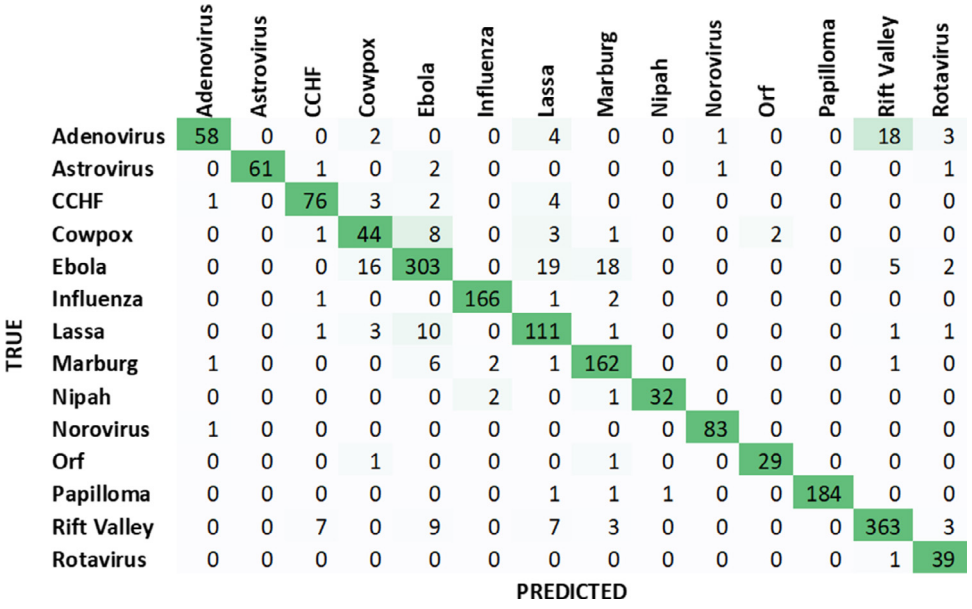

Fig. 3. The test set confusion matrix of our custom architecture model.

ages in each virus class according to the number of the virus particles and then split them by taking the first 60% of images to the training set, the next 20% to the validation set, and the last 20% to the test set. Also, for this case, we trained the well-established models both using transfer learning and from scratch. The "bad data split" results are presented in the last five rows of Table 2.

4. Discussion

The performance of our custom architecture model trained from scratch is comparable to much larger fine-tuned models transferred from ImageNet. In fact, our network achieves substantially better results than the best model (DenseNet201) trained

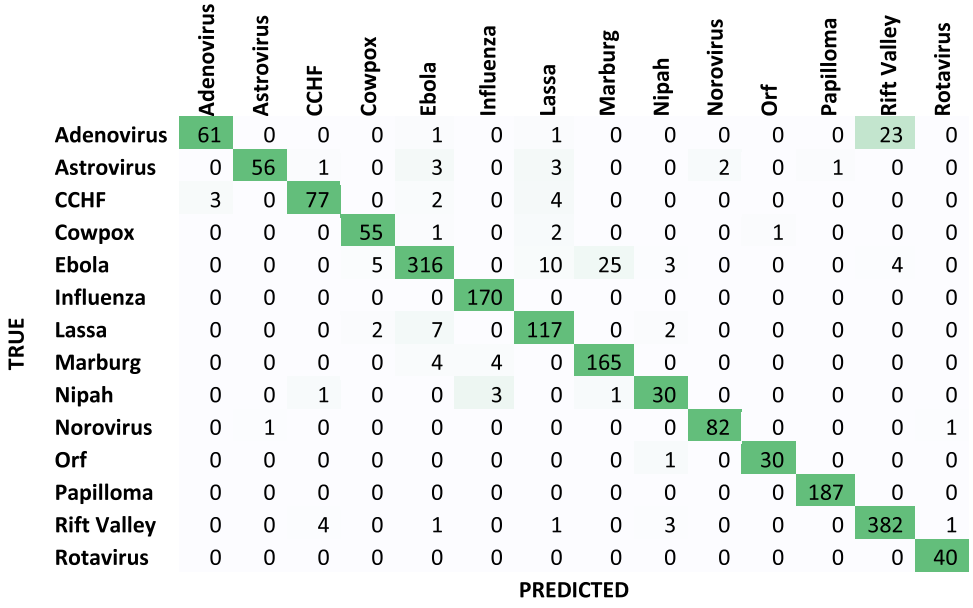

Fig. 4. The test set confusion matrix of the best model: DenseNet201 transferred from ImageNet.

from scratch on the same data despite being 10 times smaller in terms of trainable weights. In this case, we can observe that DenseNet201 performed much better when fine-tuned from ImageNet. This could indicate that the virus dataset is too small to successfully train large networks from scratch. Typically, the larger the network, the more data samples are necessary. With too little data large neural networks tend to overfit, i.e., they memorize individual data samples rather than learn the underlying patterns and dependencies.

From the confusion matrices in Figs. 3 and 4 we can deduce that a group of Ebola and Adenovirus instances were particularly difficult to classify for both models. Adenovirus and Rift Valley Virus particles have different textures (especially when it comes to the virus membrane) but they have somewhat similar sizes and shapes. Ebola and Marburg are two related and similar viruses (both being very elongated). Nevertheless, we believe that the main causes for these misclassifications are limited dataset and image quality: virus particle decomposition, clustered particles, staining artifacts, and imperfect image acquisition. Indeed, visual inspection and error tracing revealed that most of the misclassified particles come from a couple of problematic images in the test set (i.e., these image/particle appearances are not represented in the training set and have low overall image quality). Fig. 5 presents the typical images of Ebola, Marburg, Adenovirus, and Rift Valley in the training set. We also show the problematic images from the test set with the virus particles that were erroneously classified. It is clear that had we excluded these images from the test set (and included them in the training set instead) the overall performance of the models would increase. This, however, would lie about the practical usefulness of the network when dealing with problematic cases. On the other hand, the quality of these images can be questioned. It is common for clinical image analysis tools to include quality control image pre-processing that could potentially exclude such images and ask the technician preparing the samples to repeat the sample preparation and/or imaging. Nevertheless, developing such quality control is not trivial and there would always be complicated border cases. Therefore, we decided to keep these images in the test set and report the original results leaving space for the future development of methods that may be able to handle these difficult cases. One such alternative would be incorporating pre- and post-processing methods to optimally prepare the images for their classification and to refine the results. These classical image analysis methods have been shown to improve deep learning performance in various applications [39]. However, the scope of this work was only the classification of virus image patches. We have published virus detection and segmentation methods in [13] and [16]. If this was to be considered a complete system, many additional steps should be added including: sample handling, automatic imaging, image quality control, sliding window operation, and pre-/post-processing.

In our previous paper [16], we observed that the number of virus particles per image and whether they are separate or clustered are important features learned by CNNs trained on this dataset. Therefore, we included the tests on badly split data. The resulting models achieved substantially worse performance scores than the same networks trained on the representative dataset split. This leads to the conclusion that for a realistic (and limited) dataset the application knowledge and understanding of the image variation are very important when splitting the data to train and test DL models. Moreover, such knowledge is crucial when evaluating and investigating their performance: to understand which training examples should be added or what precautions/measures (such as prescreening and removing poor quality images) should be performed to improve the model. Analyzing the details of the errors and the patterns therein allows us to comprehensively evaluate a model and understand/predict its performance and shortcomings in a real setting. Moreover, in this way we can understand what performance level is good or acceptable for the dataset, i.e., what is actually meaningful to strive for, thus extending our knowledge about the dataset.

5. Conclusions

We publish and describe a benchmark dataset with TEM virus images. Public and real biomedical datasets are an important contribution and a necessity to increase the understanding of shortcomings, requirements, and potential improvements for deep learning solutions on biomedical problems or deploying solutions in clinical settings. We also presented baseline classification models with the best one achieving a 0.921 F1-score and 93.1% accuracy on the proposed representative test set. The dataset constitutes an interesting challenge for many practical applications

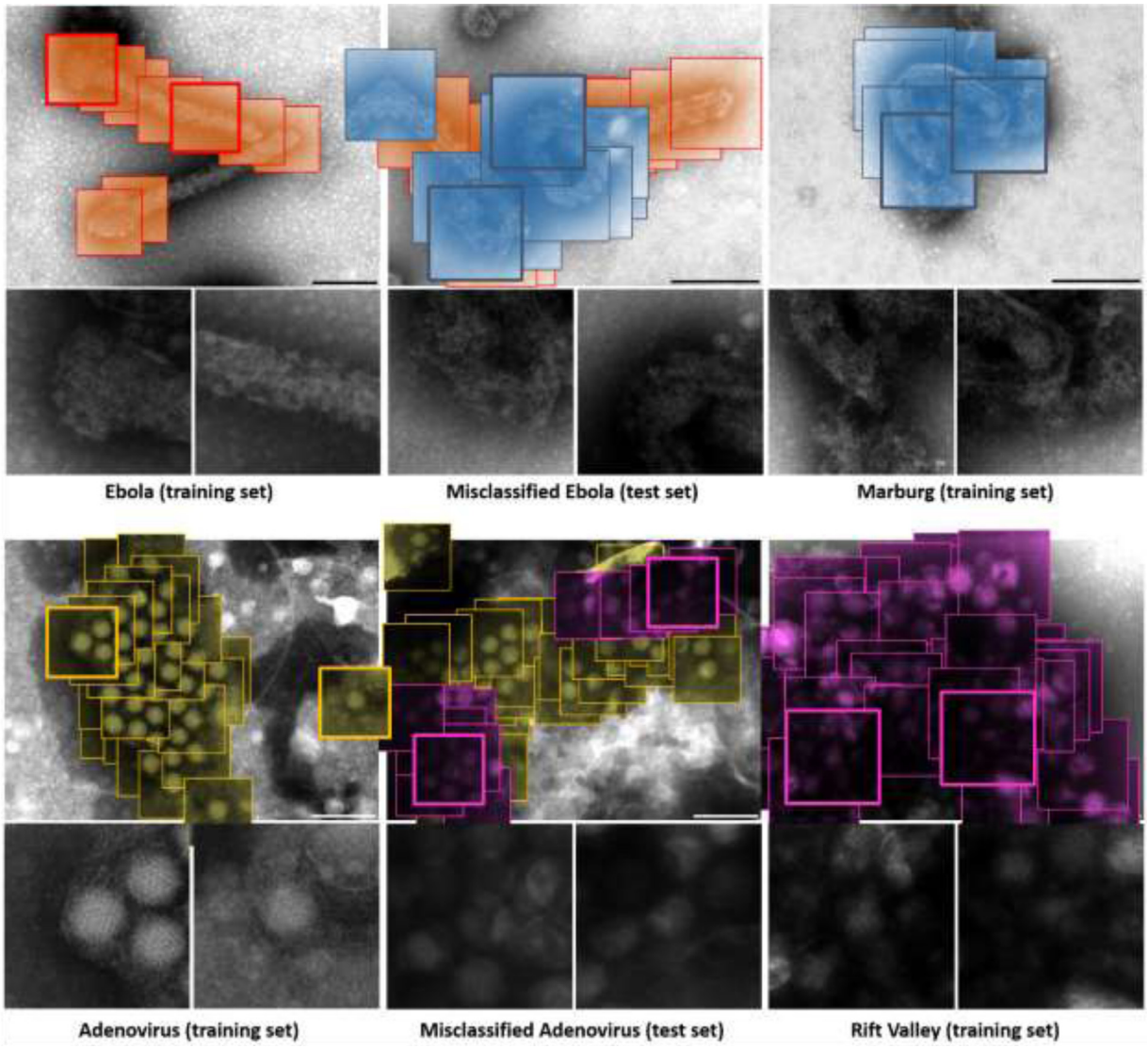

Fig. 5. Problematic images in the test set and the corresponding reference images from the training set. The image patches with bold frames are shown below the original images. The colors of the patches correspond to the best model (DenseNet201 transferred from ImageNet) classification results: Ebola – orange, Marburg – blue, Adenovirus – yellow, and Rift Valley – pink. The scale bars represent 250 nm (the patches are 256 × 256 nm).

in virology and epidemiology; e.g., virus detection, segmentation, classification, and novelty detection. We compare transfer learning to learning from scratch and hypothesize that for limited-sized datasets transfer learning is crucial for achieving good performance for large models. The performance drop is similar for both DenseNet201 with ∼145 M weights and VGG16 with 49M weights. Last but not least, we demonstrate the importance of application knowledge in creating datasets for training DL models and analyzing their results.

Declaration of Competing Interest

None.

Acknowledgment

This work was funded by the Swedish e-science initiative eSSENCE and the Uppsala University initiative AI4Research. The authors have no further relevant financial or non-financial interests or competing interests to disclose.

References

[1] H.P. Chan, R.K. Samala, L.M. Hadjiiski, C. Zhou, Deep learning in medical image analysis, in: Deep Learning in Medical Image Analysis, Springer, 2020, pp. 3–21.
[2] G. Litjens, T. Kooi, B.E. Bejnordi, A.A.A. Setio, F. Ciompi, M. Ghafoorian, J.A. van der Laak, B. Van Ginneken, C.I. Sánchez, A survey on deep learning in medical image analysis, Med. Image Anal. 42 (2017) 60–88.
[3] A. Gupta, P.J. Harrison, H. Wieslander, N. Pielawski, K. Kartasalo, G. Partel, L. Solorzano, A. Suveer, A.H. Klemm, O. Spjuth, I.-M. Sintorn, C. Wählby, Deep learning in image cytometry: a review, Cytometry Part A 95 (4) (2019) 366–380.
[4] J.M. Ede, Deep Learning in Electron Microscopy, Machine Learn. (2020).
[5] H. Kerner, Too many AI researchers think real-world problems are not relevant, Opinion. MIT Technology Review (2020) https://www.technologyreview.com/2020/08/18/1007196/ai-research-machine-learning-applications-problems-opinion/. [last visited on 17-04-2021].
[6] K.L. Wagstaff, Machine learning that matters, in: Proceeding of the 29th International Conference on Machine Learning, 2012, pp. 1851–1856.
[7] J.M. Ede, Warwick electron microscopy datasets, Machine Learn. (2020).
[8] D. Ciresan, A. Giusti, L.M. Gambardella, J. Schmidhuber, Deep neural networks segment neuronal membranes in electron microscopy images, Adv. Neural Inform. Process. Syst. (2012) 2843–2851.
[9] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Proceeding of the Conference on Medical Image Computing and Computer-Assisted Intervention, 2015, pp. 234–241.

[10] ISBI Challenge: Segmentation of neuronal structures in EM stacks http://brainiac2.mit.edu/isbi_challenge [last visited on 17-04-2021]
[11] I. Arganda-Carreras, S.C. Turaga, D.R. Berger, D. Ciresan, A. Giusti, L.M. Gambardella, J. Schmidhuber, D. Laptev, S. Dwivedi, J.M. Buhmann, T. Liu, M. Seyedhosseini, T. Tasdizen, L. Kamentsky, R. Burget, V. Uher, X. Tan, C. Sun, T.D. Pham, E. Bas, M.G. Uzunbas, A. Cardona, J. Schindelin, H.S. Seung, Crowdsourcing the creation of image segmentation algorithms for connectomics, Front. Neuroanatomy 9 (142) (2015).
[12] G. Kylberg, M. Uppström, I.-M. Sintorn, Virus texture analysis using local binary patterns and radial density profiles, in: Proceeding of the Iberoamerican Congress on Pattern Recognition, 2011, pp. 573–580.
[13] G. Kylberg, M. Uppström, K.-O. Hedlund, G. Borgefors, I.-M. Sintorn, Segmentation of virus particle candidates in transmission electron microscopy images, J. Microsc. 245 (2) (2012) 140–147.
[14] I.-M. Sintorn, G. Kylberg, Virus recognition based on local texture, in: Proceeding of the 22nd International Conference on Pattern Recognition (ICPR), 2014, pp. 3227–3232.
[15] D.J. Matuszewski, I.-M. Sintorn, Minimal annotation training for segmentation of microscopy images, Proceeding of the 15th International Symposium on Biomedical Imaging (ISBI), 2018.
[16] D.J. Matuszewski, I.-M. Sintorn, Reducing the u-net size for practical scenarios: Virus recognition in electron microscopy images, Comput. Methods Programs Biomed. 178 (2019) 31–39.
[17] D.J. Matuszewski, I.-M. Sintorn, TEM virus dataset, Mendeley Data 3 (2021), doi:10.17632/x4dwwfwtw3.3.
[18] D.J. Matuszewski, I.-M. Sintorn, TEM virus images: benchmark dataset and deep learning classification – CODE, Mendeley Data 2 (2021), doi:10.17632/kxsvzhcfgs.2.
[19] F.L.C. dos Santos, M. Paci, L. Nanni, S. Brahnam, J. Hyttinen, Computer vision for virus image classification, Biosystems Eng. 138 (2015) 11–22.
[20] Z. Wen, Z. Li, Y. Peng, S. Ying, Virus image classification using multi-scale completed local binary pattern features extracted from filtered images by multi-scale principal component analysis, Pattern Recognit. Lett. 79 (2016) 25–30.
[21] E. Ito, T. Sato, D. Sano, E. Utagawa, T. Kato, Virus particle detection by convolutional neural network in transmission electron microscopy images, Food Environ. Virol. (2018) 1–8.
[22] C. Xiao, X. Chen, Q. Xie, G. Li, H. Xiao, J. Song, H. Han, Virus identification in electron microscopy images by residual mixed attention network, Comput. Methods Programs Biomed. 198 (2021).
[23] M. Laue, L. Möller, The virusexplorer DEM - a database for diagnostic electron microscopy of viruses, Zenodo (2016).
[24] G. Kylberg. Virus Texture Dataset v. 1.0. (2012). http://www.cb.uu.se/∼gustaf/virustexture/index.html [last visited on 17-04-2021]
[25] H.R. Gelderblom, D. Madeley, Rapid viral diagnosis of Orthopoxviruses by electron microscopy: optional or a must? Viruses 10 (142) (2018).
[26] T.G. Ksiazek, D. Erdman, C.S. Goldsmith, S.R. Zaki, T. Peret, S. Emery, S. Tong, C. Urbani, J.A. Comer, W. Lim, P.E. Rollin, A novel coronavirus associated with severe acute respiratory syndrome, N. Engl. J. Med. 348 (20) (2003) 1953–1966.
[27] K.D. Reed, J.W. Melski, M.B. Graham, R.L. Regnery, M.J. Sotir, M.V. Wegner, J.J. Kazmierczak, E.J. Stratman, Y. Li, J.A. Fairley, G.R. Swain, The detection of monkeypox in humans in the Western Hemisphere, N. Engl. J. Med. 350 (4) (2004) 342–350.
[28] D.J. Matuszewski, Image and Data Analysis for Biomedical Quantitative Microscopy, Acta Universitatis Upsaliensis, 2019.
[29] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016), 2021, pp. 770–778.
[30] J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 248–255.
[31] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, R. Jozefowicz, Y. Jia, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, M. Schuster, R. Monga, S. Moore, D. Murray, C. Olah, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems (2015). Software available from tensorflow.org.
[32] F. Chollet, et al. Keras, 2015. Software available from https://keras.io.
[33] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR, 2017, pp. 4700–4708.
[34] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR, 2016, pp. 2818–2826.
[35] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 4510–4520.
[36] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: Proceeding of the European Conference on Computer Vision (ECCV) (2016), 2021, pp. 630–645.
[37] K. Simonyan, A. Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint (2014) arXiv:1409.1556.
[38] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR, 2017, pp. 1251–1258.
[39] M. Salvi, U.R. Acharya, F. Molinari, K.M. Meiburger, The impact of pre-and post-image processing techniques on deep learning frameworks: a comprehensive review for digital pathology image analysis, Comput. Biol. Med. (2020) 104129.

PhD Damian J. Matuszewski received his MSc in Computer Science at the University of São Paulo, Brazil in 2014 and PhD in Computerized Image Processing at Uppsala University, Sweden in 2019. Currently he is working as a post-doc researcher at the Department of Informoation Technology, Uppsala University, Sweden. His main research interests are image analysis, machine learning and deep learning. In particular, he is driven by the practical use of these technologies and tools in multidisciplinary applications, especially in biomedical microscopy.

Associate Professor Ida-Maria Sintorn received her PhD in image processing and remote sensing at the Swedish Agricultural University 2005, spent two years as an image analyst at the scientific research institute CSIRO in Sydney, Australia, before returning to Sweden. She has since then shared her time between academia (Swedish University of Agricultural Sciences and Uppsala University), and as Chief Technology Officer at Vironova AB, a biotech company in the field of electron microscopy and analysis. Her main research interests are image processing and machine learning for automated microscopy imaging and analysis.