Proceedings of the 18th Conference on Computer Science and Intelligence Systems pp. 889–894
DOI: 10.15439/2023F4794
ISSN 2300-5963 ACSIS, Vol. 35

Efficient Deep Learning Approach for Olive Disease Classification

Antonio Bruno
Institute of Information Science and Technologies
National Research Council
Via Moruzzi 1, Pisa, Italy
Email: antonio.bruno@isti.cnr.it

Davide Moroni
Institute of Information Science and Technologies
National Research Council
Via Moruzzi 1, Pisa, Italy
Email: davide.moroni@isti.cnr.it

Massimo Martinelli
Institute of Information Science and Technologies
National Research Council
Via Moruzzi 1, Pisa, Italy
Email: massimo.martinelli@isti.cnr.it

***Abstract*—From ancient times olive tree cultivation has been one of the most crucial agricultural activities for Mediterranean countries. In recent years, the role of Artificial Intelligence in agriculture is increasing: its use ranges from monitoring of cultivated soil, to irrigation management, to yield prediction, to autonomous agricultural robots, to weed and pest classification and management, for example, by taking pictures using a standard smartphone or an unmanned aerial vehicle , and all this eases human work and makes it even more accessible.**

In this work, a method is proposed for olive disease classification, based on an adaptive ensemble of two EfficientNet-b0 models, that improves the state-of-the-art accuracy on a publicly available dataset by 1.6-2.6%. Both in terms of the number of parameters and the number of operations, our method reduces complexity roughly by 50% and 80%, respectively, that is a level not seen in at least a decade. Due to its efficiency, this method is also embeddable into a smartphone application for real-time processing.

I. Introduction

OLIVE tree cultivation represents one of the most important activities of agriculture for the civilizations of the Mediterranean area. Indeed the countries of this area produced roughly 65% of the world's olive oils in the last years [1]. Olive-derived products have shown health benefits due to their compounds [2]. In addition, olive trees are known to adapt to environmental stresses such as salinity, drought, heat and high levels of ultraviolet B rays [3], [4], [5] generating, during the millennia, 600 species within 25 genera [6]. However, even olive trees are affected by diseases: some of them are visible on their fruits and can happen only during specific periods of the year, while others have visible signs on the leaves [7]. The signs of a disease can be different in different hosts and can evolve over time.

Although olive cultivation techniques have been perfected over the centuries, artificial intelligence has only recently entered the olive industry, bringing a series of significant innovations and improving the management of many issues, like as predicting crop yields, plant health monitoring, disease prevention, identification and classification, irrigation management, monitoring and management of agricultural activities [8], [9], [10] (e.g. sowing, harvesting, pruning,...), even for olive disease [11], [12]. We propose here a highly efficient solution that allows to classify olive diseases affecting leaves directly from images taken by standard smartphone cameras. This paper is organized as follows: in Sec. II, the dataset used for experiments is described; in Sec. III, the solutions and the experimental setup are described, while results are shown in Sec. IV. The paper ends with a discussion and conclusion in Sec. VI.

II. Dataset description

To test our solution, the largest publicly available dataset [13] has been used: it is composed of 3400 images representing olive leaves affected by *Alucus olearius* or *Olive peacock spot* or *healthy*. Tab. I shows the distribution of the classes, while a sample of images for each class is shown in Fig. 1.

TABLE I: Data distribution of the dataset used.

Class	Size
Aculus olearius	890
Healthy	1050
Olive peacock Spot	1460

III. Design description

A. EfficientNet

We selected EfficientNet-b0 [14] as the core model because, according to its structure and the obtained results, it has the best accuracy/complexity trade-off. Two main factors give the efficiency of this architecture: the first is the compound scaling (Fig. 2) by which input scaling (i.e. input size), width scaling (i.e. convolutional kernel size) and depth scaling (i.e. the number of layers) are performed in conjunction since, by observation, they are dependent; the second is the use of the inverted bottleneck MBConv (first introduced in MobileNetV2, an efficient model designed to run on smartphones) as a main module, reducing the complexity of convolution by expanding and compressing the channels.

B. Ensembling

The most significant contribution to this work is given by ensembling: it is a technique of combining several models, called *weak* models, in order to provide produce a model

Fig. 1: Samples from the dataset for Aculus olearius (first row), Healthy (second row) and Olive peacock spot (third row)

having better results than a single one [15]. Ensembling is also known to reduce errors and improve the model's generalization capabilities. Due to its resource-consuming nature and the exponential growth of model complexity, however, ensembling is scarcely used in computer vision. By contrast, our method allows performing ensembling in an adaptive and efficient way (Fig. 3):

- we use only two weak models (achieving minimality and efficiency);
- the ensemble is not a typical aggregation function, but it is performed using a linear combination layer, trainable by gradient descent (obtaining adaptivity);
- the ensemble is performed using the deep features instead of the output, excluding redundant operations (for efficiency).

C. Validation pipeline

The validation pipeline can be split into two main phases:

1) 5-fold cross-validation with end-to-end EfficientNet-b0 training, using transfer learning [16] from ImageNet pre-trained models [17], because transfer learning provides faster convergence;
2) 5-fold cross-validation with fine-tuning of the ensemble, using the two best models from the previous phase.

The design choices used during the validation are:

Input size: set to 512×512 because, after a preliminary investigation, it gives the best trade-off between image quality and computational costs.

Batch size: set to the maximum available using our GPU (32GB RAM), which is 50 for the end-to-end and 200 for the fine-tuning.

Regularization: early-stopping with patience of 10 epochs is used, helping to prevent overfitting.

Optimizer: AdaBelief [18] with learning rate $5 \cdot 10^{-4}$, betas (0.9, 0.999), eps 10^{-16}, using weight decoupling without rectifying, in order to have both fast convergence and generalization.

Validation metric: Weighted F1-score which better takes into account both errors and data imbalance.

Dataset split: training and test subsets are preset, in every run of the 5-fold cross-validation, the training set is split 80/20 in train/valid.

Standardization: data are processed in order to belong to a distribution with values around the average and the unit standard deviation, improving stability and convergence of the training.

Obviously, each run of the cross-validation of both phases is associated with a different initialization of the random model

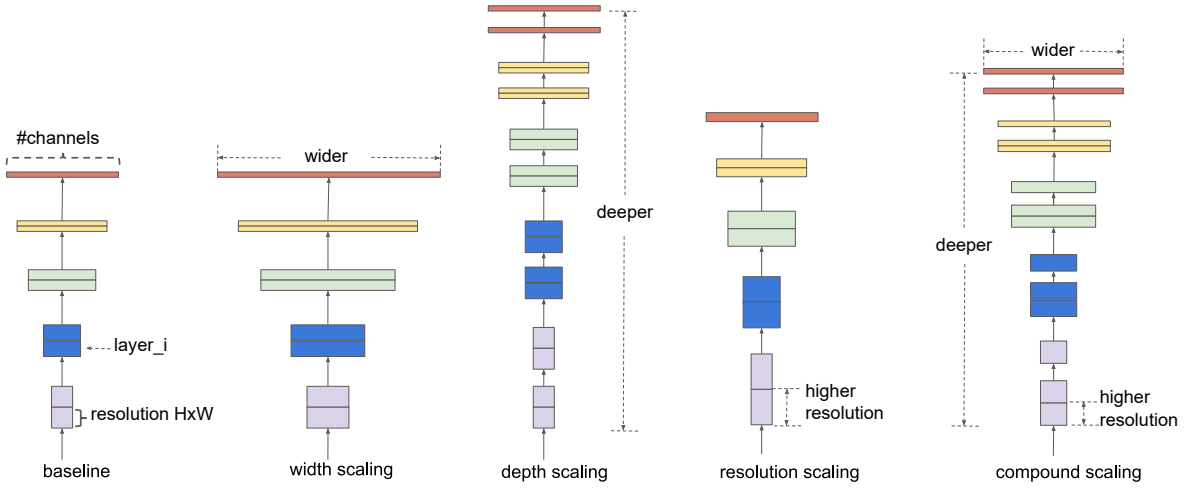

Fig. 2: Example of scaling types, from left to right: a baseline network example, conventional scaling methods that only increase one network dimension (width, depth, resolution) and, at the end, the EfficientNet compound scaling method. Image taken from the original paper [14].

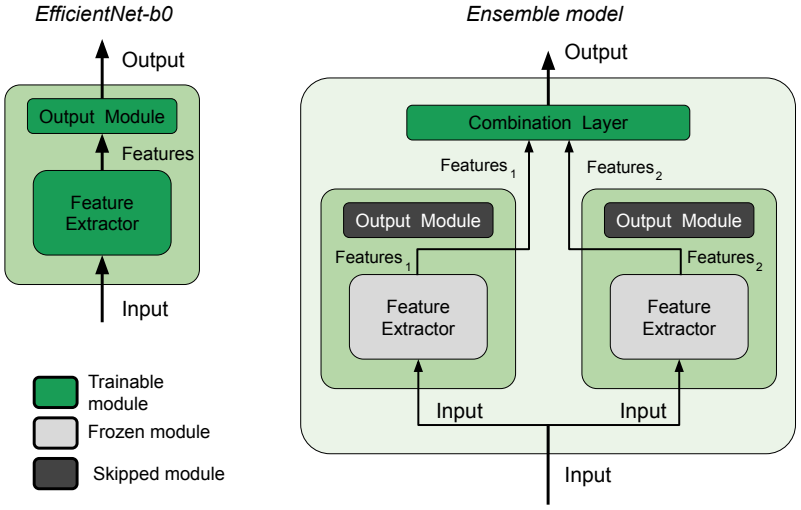

Fig. 3: Graphical scheme of the models used in this work: on the left, an end-to-end trainable EfficientNet-b0; on the right, the fine-tunable adaptive ensemble.

parameters.

IV. Experimental results

According to Tab. II, the EfficientNet-b0 with the selected design choices already provides a good starting point with an average F1-score of 0.969, 0.983 and 0.999 for test, valid and train set, respectively, with high robustness (i.e. low variance). The ensemble further reduces the variance and improves the generalization power (i.e. performance on valid and test) by an average of +1.5% and +1.4% on test and valid, respectively. The final errors are 12 (test: 9; valid: 2; train: 1) and in Fig. 4 the confusion matrix for the test set is shown.

The strength of the proposed solution is even more significant when compared with the State of the Art (SOTA) Tabs. III-IV, indeed the EfficientNet-b0 has the values of the same metric as the best performing SOTA model, and it uses only 52% of parameters and 21% of FLOPs, while considering the Ensemble the complexity (both parameters and FLOPs) is roughly doubled, but it is still lower than the SOTA, for a +1.6% on all the metrics.

V. Discussion

In order to stress our method, we tested an ensemble of five weak models: while using other datasets, generally this improves the results a little at the expense of complexity, as Tab. V shows, in this case, the results don't improve, the errors remain exactly on the same 12 images even if distributed among the different splits.

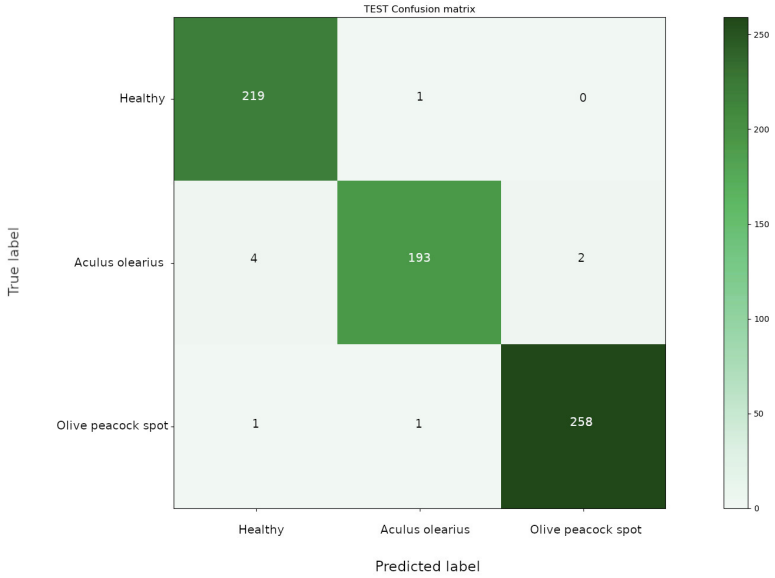

Fig. 4: The confusion matrix on the test split of the best ensemble model.

TABLE II: Metrics (F1-score) on the subset of 5-fold cross-validation runs of both end-to-end weak (left) and fine-tuning ensemble models (right). The ensemble has a twofold contribution: improving generalization performances (+1.5% on test, +1.4% on valid, on average) and robustness (halving the deviation). Data is organized best-to-worst fold (top-to-bottom), and then the models corresponding to the first two rows in the left table are used as weak models for the ensemble.

	Weak				***Ensemble***		
	Test	**Valid**	**Train**		**Test**	**Valid**	**Train**
	0.97206	0.98713	1.00000		0.98676	0.99632	0.99954
	0.97203	0.98534	0.99724		0.98382	0.99816	0.99862
	0.96925	0.97973	0.99862		0.98382	0.99816	0.99862
	0.96777	0.98159	1.00000		0.98382	0.99632	1.00000
	0.96620	0.98529	1.00000		0.98382	0.99632	0.99954
Mean	*0.96946*	*0.98382*	*0.99917*	*Mean*	*0.98441*	*0.99706*	*0.99926*
Std	*0.00231*	*0.00273*	*0.00110*	*Std*	*0.00118*	*0.00090*	*0.00055*

TABLE III: Comparing metrics of the SOTA models. Since, in their papers, the authors did not mention if the values refer either as mean/best or on test only/whole dataset, we reported the mean values (best in brackets) on both test only and whole dataset.

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
VGG-19[19]	0.82	0.75	0.94	0.84
AlexNet[19]	0.84	0.86	0.87	0.86
VGG-16[19]	0.85	0.87	0.86	0.87
AlexNet (genetic)[20]	0.87	0.87	0.87	0.87
ViT Transformer[19]	0.95	0.94	0.98	0.96
DenseNet (genetic)[20]	0.96	0.97	0.96	0.96
ViT+VGG-16[19]	0.96	0.97	0.96	0.96
ResNet (genetic)[20]	0.97	0.97	0.97	0.97
EfficientNet-b0 (test)	0.970 (0.972)	0.970 (0.972)	0.969 (0.972)	0.969 (0.972)
EfficientNet-b0 (whole)	0.990 (0.992)	0.990 (0.992)	0.990 (0.992)	0.990 (0.992)
Ensemble-b0 (test)	0.986 (0.987)	0.985 (0.987)	0.986 (0.987)	0.985 (0.987)
Ensemble-b0 (whole)	0.996 (0.996)	0.996 (0.996)	0.996 (0.996)	0.996 (0.996)

TABLE IV: Comparing complexity (expressed by the number of parameters and FLOPs) of the SOTA models.

Model	#params	FLOPs
VGG-19	≈143.6M	≈19.63G
AlexNet	≈61.1M	≈0.71G
VGG-16	≈138.3M	≈15.47G
ViT Transformer[1]	≈88.2M	≈4.41G
DenseNet[2]	≈7.9M	≈2.83G
ViT+VGG-16	≈226.5M	≈19.88G
ResNet[3]	≈11.6M	≈1.81G
EfficientNet-b0	≈5.2M	≈0.39G
Ensemble-b0	≈10M[4]	≈0.78G[5]

[1] authors did not specify the version they used, metrics are about the lightest one (ViT-B-32).
[2] authors did not specify the version they used, metrics are about the lightest one (DenseNet-121).
[3] authors did not specify the version they used, metrics are about the lightest one (ResNet-18).
[4] the actual trainable parameters are 0.1M (the parameters of the combination layer) and the gradient backward propagation stops at this layer.
[5] the forward pass can be parallelized, having the same execution time of a weak model.

TABLE V: Metrics (F1-score) related to the best ensembles of five weak models.

Test	Valid	Train
0.98529	1.00000	0.99908
0.98529	1.00000	0.99908
0.98235	1.00000	0.99908
0.98235	1.00000	0.99908
0.98235	1.00000	0.99908

This approach to ensembling has been recently introduced and discussed in [21], [22]; it has already proved excellent applicability to AI-based methods for agriculture [23].

Specifically in [21], we tested our method on seven benchmarking datasets, that are: CIFAR-10 [24], CIFAR-100 [24], Stanford Cars [25], Food-101 [26], Oxford 102 Flower [27], CINIC-10 [28] and Oxford-IIIT Pet [26]. The results demonstrated that our novelties improve the SOTA for each dataset by an average of 0.5%, using different kinds of images, reducing complexity in terms of the number of parameters up to sixty times and of FLOPs up to one hundred times. This results in a considerable saving of time and costs compared to most recent models (i.e. Vision Transformers [29]).

In [30], our method was also tested on images of plants taken on the field, in different environments, backgrounds, light conditions and at different stages of growth of the weeds. This defined the baseline for an in-progress work, in which, with the help of farmers taking pictures directly on the field using a mobile app [31], a set of models trained and being continuously extended, are contributing to significantly improving the classification of about a hundred of the main stressors that can interfere with wheat cultivation, such as weeds, pests, diseases and damages.

Another real-world application using this solution on a different domain was presented and discussed in [22]: using a public database of lung ultrasound, the SOTA was reached with 100% of accuracy in classifying healthy from Covid-19 from pneumonia cases.

VI. Conclusions

In this paper, we presented an efficient adaptive ensemble method to classify olive leaf diseases using two EfficientNet-b0 as weak models. The ensemble is performed by a linear layer that combines the features of the weak models. Our method increased the generalization strenghtby about 1.5% and reduced the variance. Moreover, by parallelizing the independent weak models, the complexity is comparable to a single weak model, having 52% of parameters and 21% of FLOPs of the best SOTA solution.

Due to its efficiency, given a significantly smaller architecture in terms of the number of tunable parameters and floating point operations comparable to those of a decade ago, this solution can also be embedded into a smartphone application for real-time classifications.

Further studies will be performed to investigate the use of the efficient adaptive ensemble method with a greater number of weak models.

References

[1] J. Blázquez, "The origin and expansion of olive cultivation," *World Olive Encyclopaedia, Madrid*, pp. 19–20, 1996.
[2] S. Valente, B. Machado, D. C. Pinto, C. Santos, A. M. Silva, and M. C. Dias, "Modulation of phenolic and lipophilic compounds of olive fruits in response to combined drought and heat," *Food chemistry*, vol. 329, p. 127191, 2020.
[3] C. Brito, L.-T. Dinis, J. Moutinho-Pereira, and C. M. Correia, "Drought stress effects and olive tree acclimation under a changing climate," *Plants*, vol. 8, no. 7, p. 232, 2019.
[4] S. Silva, C. Santos, J. Serodio, A. M. Silva, and M. C. Dias, "Physiological performance of drought-stressed olive plants when exposed to a combined heat–uv-b shock and after stress relief," *Functional Plant Biology*, vol. 45, no. 12, pp. 1233–1240, 2018.
[5] L. Regni, A. M. Del Pino, S. Mousavi, C. A. Palmerini, L. Baldoni, R. Mariotti, H. Mairech, T. Gardi, R. D'Amato, and P. Proietti, "Behavior of four olive cultivars during salt stress," *Frontiers in plant science*, vol. 10, p. 867, 2019.

[6] H. K. Obied, P. D. Prenzler, D. Ryan, M. Servili, A. Taticchi, S. Esposto, and K. Robards, "Biosynthesis and biotransformations of phenol-conjugated oleosidic secoiridoids from olea europaea l." *Natural product reports*, vol. 25, no. 6, pp. 1167–1179, 2008.

[7] A. Graniti, R. Faedda, S. O. Cacciola, and G. M. di San Lio, "19. olive diseases in a changing ecosystem," 2011.

[8] L. Benos, A. C. Tagarakis, G. Dolias, R. Berruto, D. Kateris, and D. Bochtis, "Machine learning in agriculture: A comprehensive updated review," *Sensors*, vol. 21, no. 11, 2021. doi: 10.3390/s21113758. [Online]. Available: https://www.mdpi.com/1424-8220/21/11/3758

[9] P. Lameski, E. Zdravevski, V. Trajkovik, and A. Kulakov, "Weed detection dataset with rgb images taken under variable light conditions," in *ICT Innovations 2017*, D. Trajanov and V. Bakeva, Eds. Cham: Springer International Publishing, 2017. ISBN 978-3-319-67597-8 pp. 112–119.

[10] P. Lameski, E. Zdravevski, and A. Kulakov, "Weed segmentation from grayscale tobacco seedling images," in *Advances in Robot Design and Intelligent Control*, A. Rodić and T. Borangiu, Eds. Cham: Springer International Publishing, 2017. ISBN 978-3-319-49058-8 pp. 252–258.

[11] A. Sinha and R. S. Shekhawat, "Olive spot disease detection and classification using analysis of leaf image textures," *Procedia Computer Science*, vol. 167, pp. 2328–2336, 2020. doi: https://doi.org/10.1016/j.procs.2020.03.285 International Conference on Computational Intelligence and Data Science. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050920307511

[12] S. Uğuz and N. Uysal, "Classification of olive leaf diseases using deep convolutional neural networks," *Neural Computing and Applications*, vol. 33, no. 9, pp. 4133–4149, May 2021. doi: 10.1007/s00521-020-05235-5. [Online]. Available: https://doi.org/10.1007/s00521-020-05235-5

[13] "Olive dataset kernel description," https://github.com/sinanuguz/CNN_olive_dataset, accessed: 2023-05-20.

[14] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, Jun. 2019, pp. 6105–6114.

[15] D. W. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," *J. Artif. Intell. Res.*, vol. 11, pp. 169–198, 1999. doi: 10.1613/jair.614. [Online]. Available: https://doi.org/10.1613/jair.614

[16] K. Weiss, T. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, 05 2016. doi: 10.1186/s40537-016-0043-6

[17] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009. doi: 10.1109/CVPR.2009.5206848 pp. 248–255.

[18] J. Zhuang, T. Tang, Y. Ding, S. Tatikonda, N. Dvornek, X. Papademetris, and J. Duncan, "Adabelief optimizer: Adapting stepsizes by the belief in observed gradients," *Conference on Neural Information Processing Systems*, 2020.

[19] H. Alshammari, G. Karim, I. Ben Ltaifa, M. Krichen, L. Ben Ammar, and M. Mahmood, "Olive disease classification based on vision transformer and cnn models," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–10, 07 2022. doi: 10.1155/2022/3998193

[20] H. Alshammari, G. Karim, M. Krichen, L. Ben Ammar, M. Eltaib, A. Boukrara, and M. Mahmood, "Optimal deep learning model for olive disease diagnosis based on an adaptive genetic algorithm," *Wireless Communications and Mobile Computing*, vol. 2022, 03 2022. doi: 10.1155/2022/8531213

[21] A. Bruno, D. Moroni, and M. Martinelli, "Efficient adaptive ensembling for image classification," *accepted for publication by Expert Systems - Wiley on 31st July 2023, arXiv preprint arXiv:2206.07394*, 2023.

[22] A. Bruno, G. Ignesti, O. Salvetti, D. Moroni, and M. Martinelli, "Efficient lung ultrasound classification," *Bioengineering*, vol. 10, no. 5, 2023. doi: 10.3390/bioengineering10050555. [Online]. Available: https://www.mdpi.com/2306-5354/10/5/555

[23] A. Bruno, D. Moroni, R. Dainelli, L. Rocchi, S. Morelli, E. Ferrari, P. Toscano, and M. Martinelli, "Improving plant disease classification by adaptive minimal ensembling," *Frontiers in Artificial Intelligence*, vol. 5, p. 868926, 2022.

[24] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR 2021: The Ninth International Conference on Learning Representations*, 2021.

[25] T. Ridnik, E. Ben-Baruch, A. Noy, and L. Zelnik-Manor, "Imagenet-21k pretraining for the masses," 2021.

[26] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, "Sharpness-aware minimization for efficiently improving generalization," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, 2021.

[27] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang, "Cvt: Introducing convolutions to vision transformers," 2021.

[28] Z. Lu, G. Sreekumar, E. Goodman, W. Banzhaf, K. Deb, and V. N. Boddeti, "Neural architecture transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 9, p. 2971–2989, Sep 2021. doi: 10.1109/tpami.2021.3052758

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023.

[30] R. Dainelli, M. Martinelli, A. Bruno, D. Moroni, S. Morelli, M. Silvestri, E. Ferrari, L. Rocchi, and P. Toscano, *Recognition of weeds in cereals using AI architecture*, ch. 49, pp. 401–407. [Online]. Available: https://www.wageningenacademic.com/doi/abs/10.3920/978-90-8686-947-3_49

[31] M. Massimo, "Agrosat+ project," 2023, http://si.isti.cnr.it/index.php/hid-notcategorized-category-list/228-barilla [Accessed: 31st July 2023].