

accuracy values are 93.41, 95.96, and 96.42, respectively. Also, it shows that the addition of 64 filters is better than the others according to the average validation accuracy value.

We are going to see how the pruning maintains the accuracy. Firstly, for the addition just of FC₄ with pruning point at activation-24, we can see that pruning reduces the number of parameters up to 90.06% (from an almost not pruning network) can maintain accuracy up to 98.5%. For the addition of 32 and 64 filters, the reduction rates are 90.76% and 90.62%, maintaining up to 99.0 and 99.3%, respectively. The addition of 64 filters can maintain accuracy up to 100% for the reduction rate of 74.38% at the pruning point of activation-33.

The best result was the network that can maintain accuracy up to 100% like the unpruned network. It could be reached using the pruning point at Act33 (activation-33) layer and 64 filters for dimensionality reduction. In this case, the reduction rate was 74.38%, which means it reduced from 23.7 million to 6.14 million in the number of parameters (see Table I). We also got a higher reduction rate for some pruning points, but the accuracy decreases slightly (99.0% and 99.3%), for example, at activation-24 pruning points with 32 and 64 filters.

IV. CONCLUSION

The implementation of the depth pruning of Resnet50 could work pretty well in the tobacco leaf pest dataset. The depth pruning often produces many feature maps at the end of the network. They need to be reduced to decrease the number of parameters. We applied a 1x1 kernel convolutional layer as a downsampling or dimensional reduction for the number of feature maps. We used the validation dataset to show the performance. We got the best performance as an unpruned network using pruning point at Activation-33 layer and 64 filters for downsampling. At the best performance, we accuracy up to 100% like the unpruned network. It reduces the number of parameters from 23.7 million (unpruned network) reduced up to 6.14 million, or about a 74.38% reduction rate.

ACKNOWLEDGMENT

The tobacco leaf data from this study was supported by PT Perkebunan Nusantara X (PTPN X) Jember, East Java. PTPN X is a state-owned enterprise with the main business, namely the sugar and tobacco industry.

REFERENCES

- [1] C. S. Marzan and C. R. Ruiz, "Automated tobacco grading using image processing techniques and a convolutional neural network," *Int. J. Mach. Learn. Comput.*, vol. 9, no. 6, pp. 807–813, 2019, doi: 10.18178/ijmlc.2019.9.6.877.
- [2] A. Harjoko, A. Prahara, T. W. Supardi, I. Candradewi, R. Pulungan, and S. Hartati, "Image processing approach for grading tobacco leaf based on color and quality," *Int. J. Smart Sens. Intell. Syst.*, vol. 12, no. 1, pp. 1–10, 2019, doi: 10.21307/ijssis-2019-010.
- [3] D. S. Guru, P. B. Mallikarjuna, S. Manjunath, and M. M. Shenoi, "Machine Vision Based Classification Of Tobacco Leaves For Automatic Harvesting," *Intell. Autom. Soft Comput.*, vol. 18, no. 5, pp. 581–590, 2012, doi: 10.1080/10798587.2012.10643267.
- [4] Y. Sun, H. Q. Wang, Z. Y. Xia, J. H. Ma, and M. Z. Lv, "Tobacco-disease Image Recognition via Multiple-Attention Classification Network," *J. Phys. Conf. Ser.*, vol. 1584, no. 1, 2020, doi: 10.1088/1742-6596/1584/1/012008.
- [5] D. I. Swasono, H. Tjandrasa, and C. Fatichah, "Classification of tobacco leaf pests using VGG16 transfer learning," *Proc. 2019 Int. Conf. Inf. Commun. Technol. Syst. ICTS 2019*, pp. 176–181, 2019, doi: 10.1109/ICTS.2019.8850946.
- [6] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, "Gradient-based learning applied to document recognition", *Proc. IEEE* 86 (11) (1998) 2278–2324, 1998.
- [7] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, "Going deeper with convolutions", In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015.
- [8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition", *IEEE Conf. Comput. Vis. Pattern Recognit.* 770–778 (2016), doi:10.1109/CVPR.2016.90, 2016.
- [9] J. Zou, T. Rui, Y. Zhou, C. Yang, and S. Zhang, "Convolutional neural network simplification via feature map pruning," *Comput. Electr. Eng.*, vol. 70, pp. 950–958, 2018, doi: 10.1016/j.compeleceng.2018.01.036.
- [10] G. Li, J. Wang, H. W. Shen, K. Chen, G. Shan, and Z. Lu, "CNNPruner: Pruning convolutional neural networks with visual analytics," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1364–1373, 2021, doi: 10.1109/TVCG.2020.3030461.
- [11] B. O. Ayinde, T. Inanc, and J. M. Zurada, "Redundant feature pruning for accelerated inference in deep neural networks," *Neural Networks*, vol. 118, pp. 148–158, 2019, doi: 10.1016/j.neunet.2019.04.021.
- [12] C. Yang *et al.*, "Structured Pruning of Convolutional Neural Networks via L1 Regularization," *IEEE Access*, vol. 7, pp. 106385–106394, 2019, doi: 10.1109/ACCESS.2019.2933032.
- [13] P. Singh, V. K. Verma, P. Rai, and V. P. Nambodiri, "Acceleration of Deep Convolutional Neural Networks Using Adaptive Filter Pruning," *IEEE J. Sel. Top. Signal Process.*, vol. 14, no. 4, pp. 838–847, 2020, doi: 10.1109/JSTSP.2020.2992390.
- [14] C. Liu and H. Wu, "Channel pruning based on mean gradient for accelerating Convolutional Neural Networks," *Signal Processing*, vol. 156, pp. 84–91, 2019, doi: 10.1016/j.sigpro.2018.10.019.
- [15] A. H. Ashouri, T. S. Abdelrahman, and A. Dos Remedios, "Retraining-free methods for fast on-the-fly pruning of convolutional neural networks," *Neurocomputing*, vol. 370, no. xxxx, pp. 56–69, 2019, doi: 10.1016/j.neucom.2019.08.063.
- [16] F. E. Fernandes and G. G. Yen, "Pruning Deep Convolutional Neural Networks Architectures with Evolution Strategy," *Inf. Sci. (Ny)*, vol. 552, pp. 29–47, 2021, doi: 10.1016/j.ins.2020.11.009.
- [17] S. K. Yeom *et al.*, "Pruning by explaining: A novel criterion for deep neural network pruning," *Pattern Recognit.*, vol. 115, 2021, doi: 10.1016/j.patcog.2021.107899.
- [18] A. Jordao, M. Lie, and W. R. Schwartz, "Discriminative Layer Pruning for Convolutional Neural Networks," *IEEE J. Sel. Top. Signal Process.*, vol. 14, no. 4, pp. 828–837, 2020, doi: 10.1109/JSTSP.2020.2975987.
- [19] F. Tung and G. Mori, "Deep Neural Network Compression by In-Parallel Pruning-Quantization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 3, pp. 568–579, 2020, doi: 10.1109/TPAMI.2018.2886192.
- [20] Y. Liang, W. Liu, S. Yi, H. Yang, and Z. He, "Filter pruning-based two-step feature map reconstruction," *Signal, Image Video Process.*, vol. 15, no. 7, pp. 1555–1563, 2021, doi: 10.1007/s11760-021-01888-4.
- [21] C. Qi *et al.*, "An efficient pruning scheme of deep neural networks for Internet of Things applications," *EURASIP J. Adv. Signal Process.*, vol. 2021, no. 1, 2021, doi: 10.1186/s13634-021-00744-4.
- [22] E. Jeczminek and P. A. Kowalski, "Flattening layer pruning in convolutional neural networks," *Symmetry (Basel)*, vol. 13, no. 7, pp. 1–13, 2021, doi: 10.3390/sym13071147.
- [23] S. Zhang, G. Wu, J. Gu, and J. Han, "Pruning convolutional neural networks with an attention mechanism for remote sensing image classification," *Electron.*, vol. 9, no. 8, pp. 1–19, 2020, doi: 10.3390/electronics9081209.