OUTLINE

Pendekatan Deep learning attentive recurrent network dengan model compression dan quantization untuk mengurangi kebisingan agar meningkatkan kejelasan suara manusia



RESEARCH

*Shofyan*

*2301980064*

Program Pascasarjana Ilmu Komputer

PROGRAM STUDI TEKNIK INFORMATIKA JENJANG S2

UNIVERSITAS BINA NUSANTARA

JAKARTA

2022

## Latar Belakang

Pandemi *Covid-19* pada akhirnya mengubah cara bekerja yang sebelumnya betatap muka langsung di kantor, kini harus bekerja dari rumah (*Work from Home*), komunikasi menggunakan media *video call* sudah menjadi bagian dari koordinasi dengan rekan kerja. Keadaan ruangan dirumah tidak seperti di kantor yang sudah diatur sedemikian rupa sehingga bebas dari gannguan suara seperti kendaraan yang lewat, anak kecil yang teriak, suara pedagang asongan menjajakan makanan, suara tetangga sedang memperbaiki rumah dan lain lain. Hal ini menimbulkan gangguan suara ketika berkomunikasi dengan rekan kerja. Pada kasus ini kita membutuhkan penghilang kebisingan suara ketika bekomunikasi sehingga pesan yang disampaikan dapat diterima dengan jelas.

Pendekatan statistika dalam menghilangkan kebisingan sudah menjadi masalah yang menarik bagi para peneliti mulai dari metode *Wiener filtering* (Scalart, 1996), *minimum mean-square error short-time spectral amplitude* (MMSE-STSA) *estimator* (Ephraim, 1984), dan *MMSE log-spectral amplitude* (MMSE-LSA) *estimator* (Epraim, 1985). Metode ini sering dibuat degan beberapa asumsi yang mendasar stasioner, dan tidak memiliki kemampuan untuk menangani kebisingan non-stasioner (Zhang, 2019).

Dalam beberapa tahun terakhir, metode peningkatan kualitas suara manusia *supervised* berbasis *deep neural network* (DNN) telah menunjukkan kemajuan besar. Dalam metode peningkatan kualitas suara manusia yang *supervised*, desain target pelatihan memainkan peran penting. Banyak metode DNN telah diadopsi dari pemrosesan gambar, ada perbedaan penting antara domain pada audio. Sampel audio mentah membentuk sinyal deret waktu satu dimensi yang pada dasarnya berbeda dari gambar yang membentuk dua dimensi. Sinyal audio biasanya diubah menjadi representasi waktu-frekuensi dua dimensi untuk diproses, tetapi dua sumbu, waktu dan frekuensi, tidak homogen seperti sumbu horizontal dan vertikal dalam sebuah gambar. Gambar adalah *snapshot* instan dari target dan sering dianalisis secara keseluruhan atau sebagian dengan sedikit batasan urutan; namun sinyal audio harus dipelajari secara berurutan dalam urutan kronologis (Purwins, 2019). Properti ini memunculkan solusi khusus audio yang berdasarkan urutan waktu.

Pendekatan metode *convolutional recurrent neural network* (Tan, 2018) untuk memisahkan kebisingan membutuhkan banyak biaya baik dari sumber daya komputasi maupun penggunaan *memory* (Tan, 2021). Baru-baru ini, model yang disebut *temporal convolutional network* (TCN) (Shaojie, 2018), yang menggunakan pembelajaran residual yang digabungkan dengan unit konvolusi yang melebar, dapat menangkap ketergantungan kontekstual jangka panjang dan mengungguli penelitian , (Jitong, 2017) yang mengusulkan penggunaan *recurrent neural network* (RNN) dengan empat *hidden long shortterm memory* (LSTM). Penggunaan metode TCN ini juga membutuhkan tenaga komputasi yang besar dan juga *memory*  yang banyak. Penelitian kali ini mengusulkan kompresi pada metode TCN tanpa mengurangi kualitas secara signifikan dengan mengkompresi hidden layer TCN

## Rumusan Masalah

Mengacu pada temuan masalah pada latar belakang di atas maka berikut adalah rumusan masalah dalam penelitian ini adalah:

Bagaimana membuat model *deep learning attentive recurrent network* yang lebih efisien pada penggunaan *resource* dengan *model compression* dan *quantization* dalam mengurangi kebisingan dan meningkatkan kualitas suara manusia?

## Tujuan Penelitian

Berdasarkan perumusan masalah di atas, tujuan penelitian disusun sebagai berikut:

1. Mengembangkan sistem pengurangan kebisingan dan peningkatan kualitas suara manusia dengan penggunaan *deep learning attentive recurrent network* dengan *model compression* dan *quantization* agar dapat digunakan pada kasus nyata di kehidupan sehari-hari

## Manfaat Penelitian

Dari penelitian yang dilakukan diharapkan memberikan kontribusi manfaat sebagai berikut:

1. Memberikan rancangan dan arsitektur model untuk pengurangan kebisingan dan peningkatan kualitas suara manusia yang dapat diimplementasikan dikehidupan sehari hari.
2. Menambah kejelasan suara manusia ketika pertemuan menggunakan *video conference* walaupun ada kebisingan suara.

## Ruang Lingkup

Ruang lingkup penelitian ini adalah:

* Sumber suara berasar dari *monaural* (satu*-*mikropon)
* Suara kebisingan dibuat dengan menggabungkan suara manusia dengan sumber suara kebisingan
* Dataset yang digunakan diambil dari Librispeech (Panayotov, 2015)

## Metodologi Penelitian

Pengurangan kebisingan dan meningkatkan kejelasan suara merupakan aspek penting yang dapat digunakan di banyak bidang. Teknologi ini dapat diterapkan di berbagai bidang pekerjaan misalnya pada *video call meeting*. Penelitian tentang Pengurangan kebisingan dan meningkatkan kejelasan suara telah ada sejak lama. Ada banyak model yang ada untuk mendeteksi kebisingan suara. Penelitian ini bertujuan untuk menghasilkan model yang lebih baik dari beberapa model yang ada dari segi efektifitas dan penggunaan ongkos komputasi.

Model yang diusulkan menggunakan *Deep learning* dengan kombinasi ekstraksi tingkat fitur dari dataset yang berbeda karena metode yang diusulkan memiliki kemungkinan untuk mengungguli kinerja dari model yang diusulkan sebelumnya . Garis besar penelitian dapat dilihat pada Gambar 3

Diagram

Description automatically generated

Gambar 1 Metode Penelitian

### Pengumpulan dan Prapemrosesan Dataset

Kami mengevaluasi semua model dengan cara independen speaker, noise, dan corpus. Kami menggunakan semua ucapan dari set pelatihan korpus LibriSpeech (Panayotov, 2015) untuk menghasilkan campuran pelatihan. Ini terdiri dari sekitar 280 ribu ucapan pidato lebih dari 2000 pembicara. LibriSpeech telah terbukti menjadi korpus yang efektif untuk generalisasi lintas korpus karena direkam oleh banyak sukarelawan di seluruh dunia, dan karenanya terdiri dari ucapan-ucapan yang direkam dalam kondisi akustik yang berbeda. Ucapan pelatihan yang bising dihasilkan secara online selama pelatihan dengan cara berikut. Untuk setiap sampel dalam kelompok tertentu, kami secara acak mengambil sampel ucapan ucapan, mengekstrak potongan acak 4 detik darinya, dan menambahkan potongan acak kebisingan ke dalamnya dengan *signal to noise ratio* (SNR) acak dari {−5, 4, 3, 2, 1, 0} dB. Pidato sampel digunakan tidak berubah jika durasinya lebih kecil dari 4 detik. Satu set 224 suara non-suara manusia dari perpustakaan efek suara (docbox.etsi.org) digunakan sebagai suara pelatihan.

Semua model dievaluasi pada tiga corpora yang berbeda: WSJ-SI-84 (WSJ) (Paul dkk, 1992), TIMIT (Garofolo, 1993), dan IEEE (Rothauser, 1996), yang tidak digunakan selama pelatihan. Kami menggunakan ucapan satu pembicara pria dan satu pembicara wanita dari IEEE. Perangkat tes WSJ terdiri dari 150 ucapan dari 6 pembicara yang berbeda. Set tes TIMIT terdiri dari 192 ucapan di set tes inti. IEEE Pria dan IEEE Wanita masing-masing terdiri dari 144 ucapan yang dipilih secara acak. Kami menghasilkan ucapan berisik menggunakan empat jenis suara yang berbeda: celoteh, kafetaria, pabrik, dan mesin, tidak ada yang digunakan selama pelatihan. Ucapan uji dihasilkan pada 6 SNR berbeda 5, 2, 0, 2, dan 5 dB

### Ekstraksi Fitur

Sinyal suara bising x didefinisikan sebagai jumlah dari sinyal suara bersih s dan sinyal suara n, digambarkan dengan Persamaan 6.1

x=s+n

Persamaan 6.1 Sinyal suara bising

{x,s,n} RM×1, dan M adalah jumlah sampel dalam sinyal suara. Algoritme peningkatan suara manusia bertujuan untuk mendapatkan perkiraan yang mendekati, s, dari s yang diberikan x.Tujuan dari algoritma peningkatan ucapan domain waktu adalah untuk menghitung s langsung dari x daripada menggunakan representasi T-F dari x. Peningkatan suara manusia domain waktu menggunakan DNN dapat dirumuskan sebagai Persamaan 6.2

=fθ(x)

Persamaan 6.2 Peningkatan suara manusia domain waktu

di mana fθ menunjukkan fungsi yang diwakili oleh DNN diparametrisasi oleh θ

Secara umum, algoritma peningkatan suara dirancang untuk memproses bingkai sinyal suara. Diberikan sinyal noise x, pertama-tama dipotong menjadi bingkai yang tumpang tindih yang kemudian diproses pada tingkat bingkai oleh model peningkatan suara. Biarkan X RT ×L menunjukkan matriks yang berisi bingkai sinyal x dan xt RL×1 bingkai ke-t. xt didefinisikan sebagai Persamaan 6.3

xt[k]=x[(t−1)·J+k], k=0,···,L−1

Persamaan 6.3 Matriks berisi bingkai sinyal

di mana T adalah jumlah bingkai, L adalah panjang bingkai, dan J adalah pergeseran bingkai. T diberikan oleh M , di mana menunjukkan J fungsi langit-langit. x diisi dengan nol jika M tidak habis dibagi J. Pemrosesan bingkai menggunakan DNN dapat didefinisikan sebagai Persamaan 6.4

t = fθ(xt−T1,··· ,xt−1,xt,xt+1,··· ,xt+T2)

Persamaan 6.4 Pemrosessan binkai

di mana t dihitung menggunakan xt, T1 frame masa lalu, dan T2 masa depan bingkai. Algoritme peningkatan ucapan tingkat bingkai dianggap kausal jika estimasi dari s bingkai yang diberikan dihitung menggunakan bingkai yang bising pada waktu yang kurang dari atau sama dengan t. Untuk peningkatan bicara kausal dimodifikasi sebagaimana digambarkan pada Persamaan 6.5

t =fθ(xt−T1,···,xt−1,xt)

Persamaan 3.6 Peningkatan bicara kausal

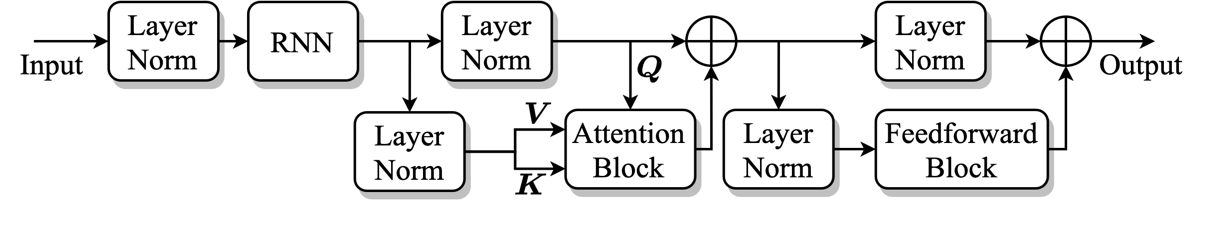
di mana t dihitung menggunakan xt dan T1 melewati frame. Kausalitas adalah persyaratan yang diperlukan untuk peningkatan kemampuan berbicara secara real-time. Lebih lanjut, kami mengamati bahwa algoritme kausal menunjukkan degradasi yang lebih besar pada korpora yang tidak terlatih dibandingkan dengan algoritme non-kausal yang sesuai. Oleh karena itu, kami juga mengembangkan dan membandingkan algoritma kausal.

### Pemisahan Data

Dataset yang digunakan oleh model dipisahkan menjadi set data splitting. Penelitian ini mengikuti metode tradisional dengan membagi dataset menjadi tujuan pelatihan dan pengujian. Langkah pertama dari pemisahan data adalah mengacak dataset untuk membuat data yang lebih merata. Setelah itu, kami membagi kereta dan menguji dengan rasio 8:2, 8 untuk kereta dan 2 untuk validasi untuk menghindari overfitting atau underfitting. Rasio 8:2 diterapkan untuk semua model yang diusulkan.

### Permodelan Attentive Recurrent Network

Sebuah diagram blok ARN diberikan pada Gambar 3.2. Blok bangunan ARN adalah normalisasi lapisan, RNN, blok self-attention, dan blok feedforward. Selanjutnya, kami menjelaskan blok bangunan ini satu per satu.



Gambar 6.2 Diagram ARN.

* + 1. ***Layer Normalization***

*Layer Normalization* adalah teknik normalisasi populer yang digunakan dalam DNN untuk meningkatkan generalisasi dan memfasilitasi pelatihan yang lebih cepat (Ba, 2016). Hal ini diusulkan sebagai alternatif untuk normalisasi *batch* (Ioffe, 2015), yang ditemukan lebih sensitif terhadap pelatihan *batch* sebagaimana digambarkan di Persamaan 6.6

Diagram

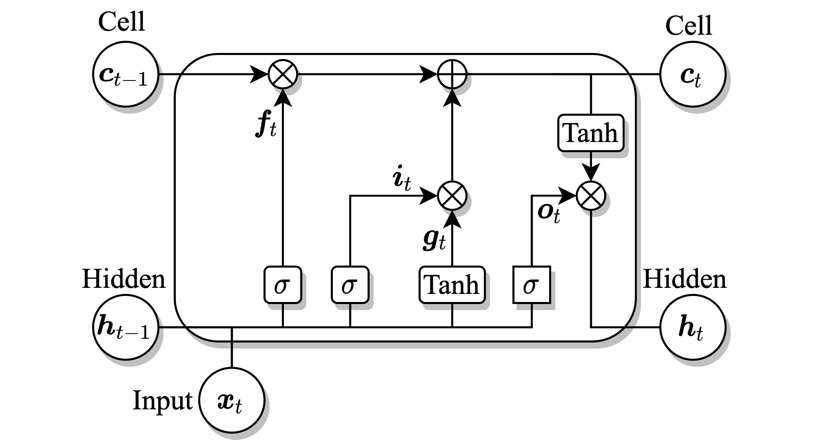
Description automatically generated with medium confidence

Persamaan 6.6 *Layer normalization*

dimana xt dan , berturut-turut adalah mean dan varians dari . Simbol dan adalah parameter yang dapat dilatih dengan ukuran yang sama dengan xt, ⊙ menunjukkan perkalian elemen, dan adalah konstanta positif kecil yang digunakan untuk menghindari pembagian dengan nol. Menunjukkan perkalian elemen, dan adalah konstanta positif kecil yang digunakan untuk menghindari pembagian dengan nol.

* + 1. **Recurrent Neural Network**

Kami menggunakan *long short-term memory* (LSTM) RNN di ARN. Sebuah diagram ilustrasi dari LSTM ditunjukkan pada Gambar. 6.3.



Gambar 6.3 Diagram ilustrasi dari LSTM

Diberikan urutan vektor input {x1,··· ,xt−1,xt,xt+1,··· ,xT}, keadaan tersembunyi pada waktu t, ht ,dihitung sebagai Persamaan 6.7

Text

Description automatically generated

Persamaan 6.7 Tanh(s)

Di mana xt, gt, dan ct masing-masing mewakili input, input blok, dan status memori (sel) pada waktu t. Selain itu, ft, dan ot masing-masing adalah gerbang yang dikenal sebagai gerbang input, gerbang lupa dan gerbang keluaran. W dan b menunjukkan bobot dan bias yang dapat dilatih.

* + 1. **Self-attention Block**

Mekanisme *self attention* didefinisikan menggunakan tiga komponen: kunci K RT×R, nilai V RT×S, dan kueri Q RT×R. Pertama, skor korelasi antara pasangan baris dari Q dan K, {Qi,Kj}, dimana i,j {1,···,T}, dihitung dengan menggunakan Persamaan 6.8

W = QKT

Persamaan 6.8 *Correlation score*

Output akhir dari *self attention block* dihitung dengan Persamaan 6.9 dengan lapisan *softmax*  yang di representasikan dengan Persamaan 6.10

Text

Description automatically generated

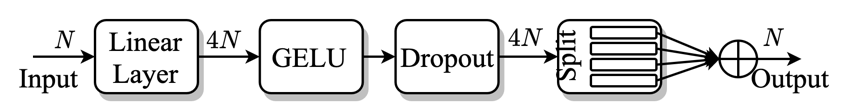
Persamaan 3.9 *Output self attention block*

A = Softmax(W ′)V ′

Persamaan 3.10 *Softmax*

* + 1. **Feedforward Block**

Blok feedforward di ARN ditunjukkan pada Gambar 6.4. Sebuah input yang diberikan dari ukuran N diproyeksikan ke ukuran 4N menggunakan lapisan linier, yang diikuti oleh unit linier kesalahan Gaussian (GELU) [50] dan lapisan putus sekolah [51]. Akhirnya, keluaran berukuran 4N dibagi menjadi empat vektor berukuran N, yang dijumlahkan untuk mendapatkan keluaran akhir.



Gambar 6.4 *Feedforward Block*

Dengan blok bangunan yang dijelaskan, kami sekarang menyajikan aliran pemrosesan ARN yang ditunjukkan pada Gambar. 1. Input ke ARN pertama-tama dinormalisasi dan kemudian diproses menggunakan RNN. Output dari RNN dinormalisasi menggunakan dua normalisasi lapisan paralel. Aliran pertama digunakan sebagai Q dan aliran kedua digunakan sebagai K dan V untuk blok perhatian berikut. Output dari blok perhatian ditambahkan ke Q untuk membentuk koneksi residual. Sekali lagi, output dinormalisasi menggunakan dua normalisasi lapisan paralel. Aliran pertama diproses menggunakan blok *feed forward* dan aliran kedua ditambahkan ke *output blok feed forward* untuk membentuk sambungan sisa.

### Klasifikasi dan Evaluasi

Kami menggunakan *short-time objective intelligibility* (STOI) dan erceptual evaluation of speech quality (PESQ) sebagai metrik evaluasi untuk membandingkan model yang dilatih di Librispeech. STOI memiliki kisaran tipikal [0, 1], yang secara kasar mewakili persen benar. PESQ memiliki rentang [−0.5,4.5], di mana skor yang lebih tinggi menunjukkan kualitas ucapan yang lebih baik. Kedua metrik tersebut biasanya digunakan untuk mengevaluasi algoritma peningkatan bicara

# DAFTAR PUSTAKA

Agrios, G. N. (1988). *Plat Pathology* (Third; G. N. Agrios, Ed.). https://doi.org/https://doi.org/10.1016/C2012-0-01423-8

Brahimi, M., Arsenovic, M., Laraba, S., Sladojevic, S., Boukhalfa, K., & Moussaoui, A. (2018). Deep Learning for Plant Diseases: Detection and Saliency Map Visualisation. *Computational Economics*, 1–21. https://doi.org/10.1007/978-3-319-90403-0\_6

Brahimi, M., Boukhalfa, K., & Moussaoui, A. (2017). Deep Learning for Tomato Diseases: Classification and Symptoms Visualization. *Applied Artificial Intelligence*, *31*(4), 299–315. https://doi.org/10.1080/08839514.2017.1315516

Cao, Q., Balasubramanian, N., & Balasubramanian, A. (2017). *MobiRNN: Efficient Recurrent Neural Network Execution on Mobile GPU*. 1–6. https://doi.org/10.1145/3089801.3089804

Castro, W., Oblitas, J., Maicelo, J., & Avila-George, H. (2018). Evaluation of Expert Systems Techniques for Classifying Different Stages of Coffee Rust Infection in Hyperspectral Images. *International Journal of Computational Intelligence Systems*, *11*(1), 86. https://doi.org/10.2991/ijcis.11.1.8

Chemura, A., Mutanga, O., & Dube, T. (2016). Separability of coffee leaf rust infection levels with machine learning methods at Sentinel-2 MSI spectral resolutions. *Precision Agriculture*, *18*(5), 859–881. https://doi.org/10.1007/s11119-016-9495-0

Chemura, A., Mutanga, O., Sibanda, M., & Chidoko, P. (2018). Machine learning prediction of coffee rust severity on leaves using spectroradiometer data. *Tropical Plant Pathology*, *43*(2), 117–127. https://doi.org/10.1007/s40858-017-0187-8

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*, *145*(September 2017), 311–318. https://doi.org/10.1016/j.compag.2018.01.009

Garcia, J., & Barbedo, A. (2018). Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. *Computers and Electronics in Agriculture*, *153*(August), 46–53. https://doi.org/10.1016/j.compag.2018.08.013

Goodfellow, I., Bengio, Y., & Courville, A. (2013). *Deep Learning*. Retrieved from www.deeplearningbook.org

Gould, S., Fulton, R., & Koller, D. (2009). Decomposing a Scene into Geometric and Semantically Consistent Regions. *International Conference on Computer Vision (ICCV)*, (Iccv). https://doi.org/10.1109/ICCV.2009.5459211

Ham, M., Moon, J. J., Lim, G., Song, W., Jung, J., Ahn, H., … Kim, H.-S. (2019). *NNStreamer: Stream Processing Paradigm for Neural Networks, Toward Efficient Development and Execution of On-Device AI Applications*. Retrieved from http://arxiv.org/abs/1901.04985

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Identity mappings in deep residual networks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9908 LNCS*, 630–645. https://doi.org/10.1007/978-3-319-46493-0\_38

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., … Adam, H. (2017). *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. https://doi.org/10.1016/S1507-1367(10)60022-3

Hughes, D. P., & Salathe, M. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. *ArXiv*. https://doi.org/10.1111/1755-0998.12237

Huynh, L. N., Lee, Y., & Balan, R. K. (2017). DeepMon: Mobile GPU-based Deep Learning Framework for Continuous Vision Applications. *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services - MobiSys ’17*, 82–95. https://doi.org/10.1145/3081333.3081360

Karpathy, A. (2018). CS231n Convolutional Neural Networks for Visual Recognition. Retrieved from http://cs231n.github.io/convolutional-networks/

KOMINFO. (2017). Rayakan Hari Kopi, Kemenperin Terus Tingkatkan Ekspor Kopi Nasional. Retrieved December 15, 2018, from https://kominfo.go.id/content/detail/10775/rayakan-hari-kopi-kemenperin-terus-tingkatkan-ekspor-kopi-nasional/0/artikel\_gpr

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *ImageNet Classification with Deep Convolutional Neural Networks*. *12*, 04015009. https://doi.org/10.1061/(ASCE)GT.1943-5606.0001284

LeCun, Y. (1989). Generalization and network design strategies. *Univ. of Toronto*, 143–155. https://doi.org/10.1242/jeb.02092

LeCun, Y., & Bengio, Y. (1995). *Convolutional Networks for Images, Speech, and Time-Series*. (November). Retrieved from internal-pdf://189.226.74.163/Convolutional Networks for Images, Speech, and.pdf

Lecun, Y., Bengio, Y., & Hinton, G. (2015). *Deep learning*. *521*. https://doi.org/10.1038/nature14539

Liu, C., Zoph, B., Neumann, M., Shlens, J., Hua, W., Li, L. J., … Murphy, K. (2018). Progressive Neural Architecture Search. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11205 LNCS*, 19–35. https://doi.org/10.1007/978-3-030-01246-5\_2

Loussaief, S., & Abdelkrim, A. (2018). Convolutional Neural Network Hyper-Parameters Optimization based on Genetic Algorithms. *International Journal of Advanced Computer Science and Applications*, *9*(10), 252–266. https://doi.org/10.14569/ijacsa.2018.091031

Mahfud, & Cholil, M. (2012). *Teknologi dan Strategi Pengendalian Penyakit Karat Daun Untuk Meningkatkan produksi Kopi Nasional*. *5*(1), 44–57.

Mahfud, M., Siti, N., Ismiyati, & Ardiansyah. (2010). Kajian penerapan teknologi produksi pada usahatani kopi robusta di lokasi prima tani kabupaten pasuruan. *Jurnal Pengkajian Dan Pengembangan Teknologi Pertanian*, *13*(2), 141–147.

Mengistu, A. D., Alemayehu, D. M., & Mengistu, S. G. (2016). Ethiopian Coffee Plant Diseases Recognition Based on Imaging and Machine Learning Techniques. *International Journal of Database Theory and Application*, *9*(4), 79–88. https://doi.org/10.14257/ijdta.2016.9.4.07

Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). *Using Deep Learning for Image-Based Plant Disease Detection*. *7*(September), 1–10. https://doi.org/10.3389/fpls.2016.01419

Salar, S., Oskouei, L., Golestani, H., & Hashemi, M. (2016). CNNdroid : GPU-Accelerated Execution of Trained Deep Convolutional Neural Networks on Android Comparing Mobile and Desktop GPUs. *Proceedings of the 24th ACM International Conference on Multimedia*, 1201–1205. https://doi.org/10.1145/2964284.2973801

Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Network For Large-Scale Image Recognation. *ICLR 2015: Very Deep Convolutional Network For Large-Scale Image Recognation*. https://doi.org/10.2146/ajhp170251

Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Computational Intelligence and Neuroscience*, *2016*, 11. https://doi.org/10.1155/2016/3289801

Sugiarti, L. (2017). *Analisis Tingkat Keparahan Penyakit Karat Winaya Mukti Tanjungsari*. *1*(2), 80–89.

Suhartono, D., Aditya, W., Lestari, M., & Yasin, M. (2013). Expert System in Detecting Coffee Plant Diseases. *International Journal of Electrical Energy*, *1*(3), 156–162. https://doi.org/10.12720/ijoee.1.3.156-162

Szegedy, C., Ioffe, S., Vanchouke, V., & Alemi, A. (2016). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *Journal of Urology*. https://doi.org/10.1016/S0022-5347(17)50340-7

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Pierre, S., Reed, S., … Andrew, R. (2015). Going Deeper with Convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–9. https://doi.org/10.1002/jctb.4820

Tan, M., Chen, B., Pang, R., Vasudevan, V., & Le, Q. V. (2018). *MnasNet: Platform-Aware Neural Architecture Search for Mobile*. Retrieved from http://arxiv.org/abs/1807.11626

Vanetti, M. (2007). Confusion Matrix Online Calculator. Retrieved April 27, 2019, from Confusion Matrix Online Calculator website: http://www.marcovanetti.com/pages/cfmatrix/?noc=8%0A

Yanai, K., Tanno, R., & Okamoto, K. (2016). *Efficient Mobile Implementation of A CNN-based Object Recognition System*. 362–366. https://doi.org/10.1145/2964284.2967243

Zhang, X., Qiao, Y., Meng, F., Fan, C., & Zhang, M. (2018). Identification of maize leaf diseases using improved deep convolutional neural networks. *IEEE Access*, *6*, 30370–30377. https://doi.org/10.1109/ACCESS.2018.2844405

Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. V. (2017). *Learning Transferable Architectures for Scalable Image Recognition*. https://doi.org/10.1044/1092-4388(2002/060