Plant Disease Classification Using Deep Learning with Two Color Aware Branch

Darrell Cornellius Rivaldo   
*C14210025*  
*Universitas Kristen Petra*Surabaya, Indonesia  
c14210025@john.petra.ac.id

Nicholas Gunawan   
*C14210099*   
*Universitas Kristen Petra*Surabaya, Indonesia  
c14210099@john.petra.ac.id

Nico Samuelson Tjandra   
*C14210017   
Universitas Kristen Petra*Surabaya, Indonesia  
c14210017@john.petra.ac.id

*Abstract*— Seiring dengan bertumbuhnya populasi dunia, dibutuhkan upaya dan inovasi untuk meningkatkan produksi pangan secara berkelanjutan. *Crop diseases* atau penyakit pada tanaman berkontribusi terhadap 20 hingga 40% kegagalan panen pertanian global. Yang dimana, penyakit daun pada tanaman atau plant leaf disease merupakan permasalahan utama yang terjadi pada tanaman pokok, yang dimana memiliki potensi untuk menyebabkan kegagalan panen. Pada penelitian ini, kami melakukan percobaan untuk mengimplementasikan konsep parallel branching yang dikemukakan pada model EfficientNetV2. Selain itu, kami menggunakan ruang warna CIELAB yang lebih efektif dalam mengisolasi fitur level abu-abu pada *channel* "L" dan fitur terkait warna pada *channel* "AB. Penelitian ini menggunakan model *deep learning* EfficientNetV2 B0 yang dimodifikasi sehingga memiliki dua cabang, yaitu cabang akromatik (L) dan cabang kromatik (AB) pada *stem* dan hingga 4 *block* pertama. Kami menguji model yang telah dikembangkan pada dua dataset, yaitu Plant Village Dataset dan Cropped-PlantDoc Dataset. Dari eksperimen yang kami lakukan pada dataset PlantVillage, model EfficientNetV2 yang dimodifikasi ini mampu menghasilkan akurasi terbaik ketika dilakukan percabangan pada 4 block pertama. Meskipun model yang kami kembangkan belum dapat mengalahkan akurasi dari model InceptionV3 milik Schuler, model kami memiliki jumlah parameter yang jauh lebih kecil sehingga waktu yang dibutuhkan untuk training juga jauh lebih cepat. Eksperimen kami juga mendukung hasil eksperimen yang dilakukan oleh Schuler dimana kontribusi AB channel yang lebih besar akan memberikan hasil model yang lebih baik. Berdasarkan eksperimen kami, hasil terbaik didapatkan jika 50% - 80% filter dimasukkan pada AB channel.

Keywords—parallel branching, EfficientNet, Plant Leaf Disease Classification, CIELab

# introduction

Seiring dengan bertumbuhnya populasi dunia, dibutuhkan upaya dan inovasi untuk meningkatkan produksi pangan secara berkelanjutan. Pertanian merupakan elemen penting dari pembangunan berkelanjutan, terutama di negara-negara berkembang. Hal ini dikarenakan kemajuan dalam bidang pertanian merupakan dasar bagi transformasi struktural ekonomi negara-negara berkembang [[1]](#one). Indonesia adalah negara agraris, yang dimana sektor pertanian berperan besar terhadap perekonomian negara [[2]](#two). Oleh karena itu, sangat penting bagi suatu negara untuk melindungi dan memajukan sektor pertaniannya agar tetap stabil dan terhindar dari kegagalan.

*Crop diseases* atau penyakit pada tanaman berkontribusi terhadap 20 hingga 40% kegagalan panen pertanian global [[3]](#three). Yang dimana, penyakit daun pada tanaman atau plant leaf disease merupakan permasalahan utama yang terjadi pada tanaman pokok, yang dimana memiliki potensi untuk menyebabkan kegagalan panen [[4]](#four). Cara tradisional untuk mendeteksi dan melakukan klasifikasi penyakit pada tanaman adalah dengan melakukan pengamatan fisik dari daun tanaman. Namun, pendekatan ini dinilai kurang baik dikarenakan pengamatan fisik tidak dapat selalu diandalkan dan dapat mengakibatkan penurunan produksi pertanian secara signifikan [[5]](#five).

Demilie [[6]](#six) melakukan perbandingan dari beberapa metode *machine learning* dan *deep learning* untuk *tugas plant leaf disease classification*. Untuk *machine learning*, Demilie menggunakan beberapa algoritma seperti *Naïve Bayes Classifier, K-Nearest Neighbors, Decision Tree, Support Vector Machine*, dan *Random Forest*. Sedangkan untuk *Deep Learning*, Demilie menggunakan algoritma *Convolutional Neural Network* (CNN) dan *Artificial Neural Networks* (ANN). Dalam penelitian ini, ditemukan bahwa teknik Deep Learning menggunakan CNN adalah yang terbaik untuk mengklasifikasikan penyakit daun pada tanaman. CNN terbukti sangat efektif dibandingkan berbagai pendekatan *machine learning* karena fleksibilitasnya dan kemampuan untuk mengekstraksi fitur secara otomatis.  Sehingga, CNN dapat menemukan *extra features* yang dapat membantu algoritma ini menghasilkan hasil yang *superior*.

Chaudary et al. [[7]](#seven) mempelajari *plant disease spot segmentation* dalam *color space* YCbCr, HSI, dan CIE Lab. Dalam penelitian ini, ditemukan bahwa *color channel* "A" dari ruang warna CIE Lab memberikan hasil yang paling akurat dibandingkan dengan *color channel* lain yang diuji. Selain itu, penelitian ini juga menunjukkan bahwa *noise* pada gambar, yang berasal dari latar belakang, pencahayaan urat daun, dan *flash* kamera, dapat diminimalkan dengan menggunakan ruang warna CIE Lab. Hal ini bermanfaat untuk mengklasifikasi gambar yang diambil di lapangan, di mana gambar sering kali memiliki berbagai jenis noise. Schuler et al. [[8]](#eight) menemukan bahwa dengan mentransformasi *color channel* RGB ke CIE Lab dapat secara efektif mengisolasi fitur *gray-level* pada "L" channel dan fitur terkait warna pada channel "AB".

Schuler et al. [[8]](#eight) melakukan modifikasi terhadap arsitektur model *deep learning* InceptionV3 yang sebelumnya telah dikembangkan oleh Toda & Okura [[9]](#nine). Schuler et al. memodifikasi tiga lapisan konvolusi pertama dengan membuat cabang paralel yang menerima data *achromatic* dan *chromatic* menggunakan CIE Lab *color space* (L *channel* & AB *channel*). Konsep *parallel branching* ini memungkinkan model mempelajari fitur warna dengan lebih baik. L branch memahami fitur *achromatic* seperti tekstur, tepi, dan area rusak, sementara AB *branch* mempelajari fitur *chromatic* seperti lesi dan warna umum daun. Sehingga, model yang dikembangkan akan lebih tahan terhadap berbagai noise yang terdapat pada gambar ketika melakukan *plant leaf disease classification*. Dari penelitian ini, ditemukan jika model yang telah dimodifikasi dengan parallel branching memiliki performance yang lebih baik dibanding model yang hanya menggunakan single branch dalam melakukan klasifikasi penyakit tanaman. Penggunaan *decorrelated color space* juga membantu mengurangi jumlah learning parameter dan floating point operations akibat adanya filter weight yang redundant.

Atila et al. [[10]](#ten) melakukan evaluasi komprehensif terhadap berbagai model *deep learning* dalam tugas klasifikasi penyakit daun tanaman dengan menggunakan PlantVillage *dataset*. Penelitian ini membandingkan beberapa model *deep learning state-of-the-art*, termasuk AlexNet, VGG16, ResNet50, InceptionV3, dan EfficientNet, dengan memanfaatkan *transfer learning* dari ImageNet. Penelitian ini menggunakan dua versi dari PlantVillage *dataset*, yaitu versi *normal* dan versi *augmented* dari Geetharamani dan Pandian [[11]](#eleven). Hasil penelitian menunjukkan bahwa model EfficientNet B5 mencapai akurasi tertinggi pada dataset normal dengan akurasi sebesar 99,91%, sedangkan model EfficientNet B4 mencapai akurasi tertinggi pada dataset *augmented* dengan akurasi sebesar 99,97%. Selain itu, *time per epoch* pada proses training untuk EfficientNet B4 maupun B5 selalu konsisten lebih rendah dibandingkan InceptionV3, menunjukkan efisiensi waktu dan performa yang lebih baik pada model EfficientNet dalam tugas klasifikasi ini.

Dalam penelitian ini, kami bertujuan untuk mengimplementasikan konsep parallel branching yang dikemukakan oleh Schuler et al. [[8]](#eight) pada model EfficientNet. Menurut penelitian Atila [[10]](#ten), model EfficientNet tidak hanya memberikan akurasi yang lebih tinggi, tetapi juga menawarkan efisiensi waktu yang lebih baik. Dengan menerapkan *parallel branching*, diharapkan model dapat mempelajari fitur warna dengan lebih baik dan menjadi lebih tahan terhadap berbagai jenis *noise* yang ada. Selain itu, kami menggunakan ruang warna CIELAB yang lebih efektif dalam mengisolasi fitur level abu-abu pada *channel* "L" dan fitur terkait warna pada *channel* "AB". Penelitian ini menggunakan model *deep learning* EfficientNetV2 B0 yang dimodifikasi sehingga memiliki dua cabang, yaitu cabang akromatik (L) dan cabang kromatik (AB) pada *stem* dan hingga 4 *block* pertama. Kami menguji model yang telah dikembangkan pada dua dataset, yaitu Plant Village Dataset [[12]](#twelve) dan Cropped-PlantDoc Dataset [[13]](#thirteen).

Penelitian ini diharapkan dapat menjawab beberapa rumusan masalah berikut:

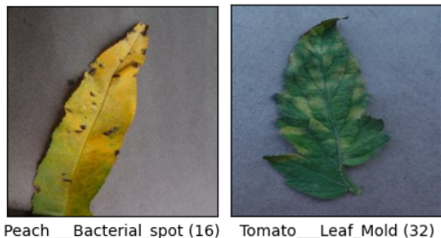
1. Bagaimana performa arsitektur EfficientNetV2 jika dibandingkan dengan arsitektur InceptionV3?

2. Berapa jumlah *block* pertama dari EfficientNetV2 yang harus dipecah untuk mencapai akurasi model yang optimal?

3. Berapa proporsi filter yang tepat antara kedua cabang EfficientNetV2 untuk menghasilkan akurasi model yang paling baik?

# penjelasan dataset

Dataset yang digunakan dalam eksperimen ini ada dua yaitu PlantVillage Dataset [[12]](#twelve) dan Cropped PlantDoc Dataset [[13]](#thirteen). Kedua *dataset* tersebut terdiri dari gambar-gambar daun yang sehat dan memiliki penyakit disertai dengan nama dari penyakitnya. Kedua *dataset* tersebut sama-sama menggunakan tiga channel warna RGB. Pada *dataset* PlantVillage yang digunakan, terdapat 54305 gambar dengan 38 kategori gambar. Setiap gambar pada PlantVillage resolusinya kemudian diubah menjadi 128x128. Gambar [1](#gambar1) ini menunjukkan contoh gambar dari *dataset* PlantVillage.

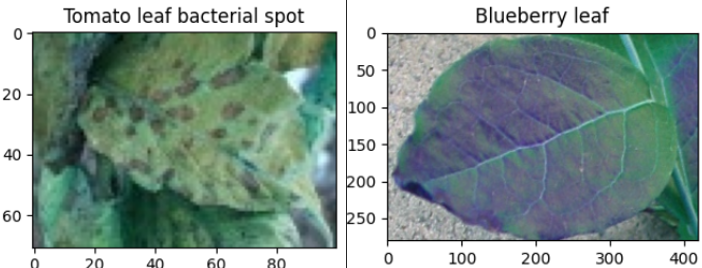


Gambar 1

|  |  |
| --- | --- |
| **Class Name** | **Number of Images** |
| Apple with scab | 630 |
| Apple with black rot | 621 |
| Apple with cedar apple rust | 275 |
| Healthy apple | 1645 |
| Blueberry with healthy | 1502 |
| Cherry with powdery mildew | 1052 |
| Cherry with healthy | 854 |
| Corn with grey leaf spot | 513 |
| Corn with common rust | 1192 |
| Corn with northern leaf blight | 985 |
| Healthy corn | 1162 |
| Grape with black rot | 1180 |
| Grape with black measles | 1383 |
| Grape with leaf blight | 1076 |
| Healthy grape | 423 |
| Orange with Huanglongbing | 5507 |
| Peach with bacterial spot | 2297 |
| Healthy peach | 360 |
| Pepper with bacterial spot | 997 |
| Healthy pepper | 1478 |
| Potato with early blight | 1000 |
| Healthy potato | 152 |
| Potato with late blight | 1000 |
| Healthy raspberry | 371 |
| Healthy soybean | 5090 |
| Squash with powdery mildew | 1835 |
| Healthy strawberry | 456 |
| Strawberry with leaf scorch | 1109 |
| Tomato with bacterial spot | 2127 |
| Tomato with early blight | 1000 |
| Healthy tomato | 1591 |
| Tomato with late blight | 1909 |
| Tomato with leaf mold | 952 |
| Tomato with septoria leaf spot | 1771 |
| Tomato with two spotted spider mite | 1676 |
| Tomato with target spot | 1404 |
| Tomato with mosaic virus | 373 |
| Tomato with yellow leaf curl virus | 5357 |
| **Total gambar** | **54,305** |

**Tabel 1:** Class dari PlantVillage Dataset

Berbeda dengan PlantVillage yang seperti diambil dalam sebuah lab atau keadaan (*background*) yang sederhana, gambar pada *dataset* Cropped PlantDoc diambil secara langsung dalam keadaan nyata seperti di kebun, taman, dan lain-lain. Dalam *dataset* ini terdapat 8883 gambar dengan 29 kategori gambar. Setiap resolusi dari gambar kemudian juga diubah menjadi ukuran 128x128. *Gambar* [2](#gambar2) merupakan contoh gambar dari dataset Cropped PlantDoc.



Gambar 2

|  |  |
| --- | --- |
| **Class Name** | **Number of Images** |
| Blueberry leaf | 849 |
| Tomato leaf yellow virus | 829 |
| Peach leaf | 620 |
| Raspberry leaf | 556 |
| Strawberry leaf | 492 |
| Tomato Septoria leaf spot | 436 |
| Tomato leaf | 396 |
| Corn leaf blight | 372 |
| Potato leaf early blight | 333 |
| Bell\_pepper leaf | 323 |
| Tomato mold leaf | 293 |
| Tomato leaf bacterial spot | 280 |
| Soybean leaf | 266 |
| Bell\_pepper leaf spot | 264 |
| Tomato leaf mosaic virus | 261 |
| Squash Powdery mildew leaf | 257 |
| Potato leaf late blight | 250 |
| Apple leaf | 247 |
| Cherry leaf | 240 |
| Tomato leaf late blight | 221 |
| Grape leaf | 220 |
| Tomato Early blight leaf | 214 |
| Apple rust leaf | 179 |
| Apple Scab Leaf | 171 |
| Grape leaf black rot | 133 |
| Corn rust leaf | 127 |
| Corn Gray leaf spot | 79 |
| Potato leaf | 11 |
| Tomato two spotted spider mites leaf | 2 |
| **Total gambar** | **8883** |

**Tabel 2 :** Class dari PlantDoc Dataset

# proposed method

Metode yang diajukan dalam eksperimen ini adalah pertama mengubah *color space* dari gambar yang awalnya adalah RGB menjadi CIE LAB sesuai yang dilakukan oleh Schuler et al. [[8]](#eight). Gambar [3](#gambar3) dan [4](#gambar4) menunjukkan contoh gambar dari PlantVillage dan Cropped PlantDoc yang sudah diubah menjadi CIE LAB dan pisah menjadi L dan AB channel. Kemudian model EfficientNetV2 akan dibuat menjadi dua cabang, cabang pertama akan diberikan input dari L *channel* (akromatik), sedangkan cabang kedua akan diberikan input dari AB *channel* (kromatik). Gambar [5](#gambar5) menunjukkan arsitektur dari EfficientNetV2-B0 dengan *width coefficient* 0.5 dan *depth coefficient* 0.5 yang sudah dimodifikasi menjadi bercabang untuk eksperimen ini.

A close-up of a leaf

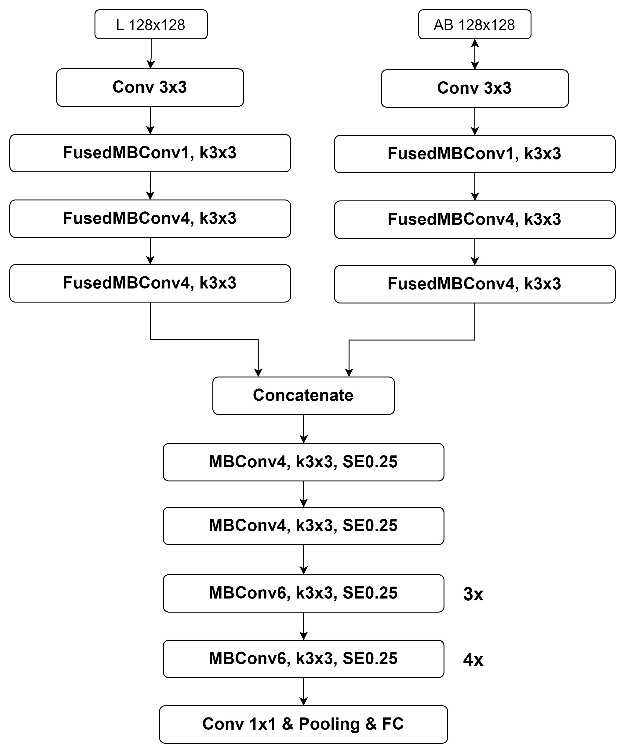
Description automatically generated

Gambar 3: contoh gambar PlantVillage yang sudah diubah menjadi CIE LAB dan dipisah menjadi L dan AB channel

A close-up of a leaf

Description automatically generated

Gambar 4: contoh gambar Cropped PlantDoc yang sudah diubah menjadi LAB CIE dan dipecah menjadi L dan AB channel



Gambar 5: Arsitektur EfficientNetV2 yang dimodifikasi dengan dibuat bercabang

Berbeda dengan arsitektur InceptionV3 yang memiliki beberapa layer konvolusi pada bagian *stem*-nya, arsitektur EfficientNetV2 hanya mempunyai satu layer konvolusi, sehingga jika mengikuti percobaan Schuler et al. [[8]](#eight) yang hanya memecah bagian *stem*-nya saja, EfficientNetV2 hanya maksimal dapat dipecah sebanyak satu layer saja. Oleh karena itu pada eksperimen ini, model EfficientNetV2 akan dipecah juga pada bagian MBConv/FusedMBConv *block*-nya sehingga memungkinkan untuk dicari seberapa banyak jumlah *block* yang perlu dicabang untuk mendapatkan hasil yang terbaik. Dalam eksperimen ini akan dicoba jumlah *block* yang dicabang mulai dari satu *block* hingga empat *block* pertama*.* Setelah itu akan dilakukan *concatenate* terhadap hasil konvolusi dari *block* paling terakhir dari cabang dan kemudian diteruskan secara serial ke *block-block­* berikutnya.

Model EfficientNetV2 yang dibuat bercabang ini juga memiliki parameter baru yaitu L *ratio*, sama seperti desain model pada percobaan Schuler et al. [[8]](#eight). Parameter L *ratio* akan menentukan jumlah distribusi filter pada masing-masing cabang. Dengan mengatur parameter L *ratio* ini, kontribusi dari tiap cabang (akromatik dan kromatik) terhadap penentuan klasifikasi penyakit dapat ditentukan. Dalam eksperimen ini akan dicoba 5 konfigurasi, yaitu L ratio 0% (AB ratio 100%), L ratio 20% (AB ratio 80%), L ratio 50% (AB ratio 50%), L ratio 80% (AB ratio 20%), dan L ratio 100% (AB ratio 0%).

Dalam implementasinya, konfigurasi dari pembagian *random* *dataset training, validation, test* yang digunakan untuk semua eksperimen dengan data PlantVillage secara berturut-turut adalah 70%, 20%, 10%. *Batch size* yang digunakan adalah 32. Optimizer yang digunakan adalah Adam dengan *learning rate* 0.01. *Training* pada data PlantVillage dilakukan sebanyak 25 epoch. Pada Cropped PlantDoc hanya terdapat perbedaan pada pembagian data, yaitu data *training* 65%, *validation* 15%, dan *testing* 20%, serta untuk jumlah epoch yang digunakan dalam training adalah 50. Setiap proses *training* dalam eksperimen ini juga diberikan *checkpoint* untuk menyimpan model dengan akurasi validasi terakhir yang paling baik.

Semua eksperimen dilakukan pada *platform* Kaggle dengan menggunakan GPU T4 yang disediakan dalam *platform* tersebut. Selain itu, model yang digunakan dalam eksperimen ini adalah model EfficientNetV2-B0 dari Keras Tensorflow [[14]](#fourteen). Model tersebut dimodifikasi dengan dibuat bercabang dan parameter yang digunakan dalam model untuk setiap eksperimen adalah *width coefficient* 0.5, *depth coefficient* 0.5, dan *dropout rate* 0.5.

# hasil eksperimen

Berangkat dari penelitian yang dilakukan oleh Schuler et al. [[8]](#eight) yang berhasil membuktikan bahwa penggunaan color space CIE Lab dan konsep model InceptionV3 yang bercabang bisa memberikan hasil yang lebih baik, kami berusaha mencari tahu apakah metode serupa bisa diterapkan di EfficientNetV2.

Kami melakukan dua eksperimen utama, yaitu untuk mencari jumlah cabang yang dapat memberikan hasil yang paling optimal serta mencari jumlah proporsi pembagian filter akromatik dan kromatik yang paling optimal. Pengujian dilakukan pada kedua dataset yan digunakan, yaitu PlantVillage dan Cropped PlantDoc. Kami kemudian melakukan pencatatan terhadap hasil akurasi yang didapat pada data testing beserta total waktu yang dibutuhkan untuk training masing-masing model.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Test Accuracy (%) | Train time (seconds) | Total parameters |
| EfficientNetV2  (0 block) | 95.77 | **1448** | 803K |
| EfficientNetV2  (1 block) | 95.64 | **1448** | 802K |
| EfficientNetV2  (2 block) | 96.17 | 1453 | 802K |
| EfficientNetV2  (3 block) | 96.60 | 1488 | 799K |
| EfficientNetV2  (4 block) | 97.36 | 1463 | **795K** |
| Inception V3  (Schuler et al.) | **99.22** | 3651 | 4.84M |

**Tabel 3:** Test accuracy dan training time untuk PlantVillage Dataset

|  |  |  |  |
| --- | --- | --- | --- |
| Architecture | Test Accuracy (%) | Train time (seconds) | Total parameters |
| 0%L + 100%AB | 95.35 | 1385 | 803K |
| 20%L + 80%AB  (4 block) | **97.36** | 1463 | 797K |
| 50%L + 50%AB  (4 block) | 96.30 | **1487** | 792K |
| 80%L + 20%AB  (4 block) | 95.50 | 1479 | 795K |
| 100%L + 0%AB | 91.00 | 1381 | **803K** |

**Tabel 4:** Efek perbandingan filter pada PlantVillage Dataset

Dari eksperimen yang kami lakukan pada dataset PlantVillage, model EfficientNetV2 mampu menghasilkan akurasi terbaik ketika dilakukan percabangan pada 4 block pertama. Meskipun belum dapat mengalahkan model InceptionV3 milik Schuler, model kami memiliki jumlah parameter yang jauh lebih kecil sehingga waktu yang dibutuhkan untuk training juga jauh lebih cepat. Hal ini menjadi penting mengingat kemampuan device yang digunakan di lapangan sangat bervariasi.

Eksperimen kami juga mendukung hasil eksperimen yang dilakukan oleh Schuler dimana kontribusi AB channel yang lebih besar akan memberikan hasil model yang lebih baik. Berdasarkan eksperimen kami, hasil terbaik didapatkan jika 50% - 80% filter dimasukkan pada AB channel. Kami juga mencoba menggunakan salah satu cabang saja, 100%L serta 100%AB. Hasil yang kami dapatkan membuktikan bahwa AB channel memberikan pengaruh yang lebih besar dalam kemampuan model melakukan klasifikasi.

Berdasarkan kombinasi terbaik yang sudah ditemukan tersebut (4 *block* cabang dan 20%L 80%AB), kami juga mencoba melakukan training pada model EfficientNetV2-B0 yang *width coefficient* dan *depth coefficient*-nya tidak kami ubah, yaitu keduanya satu. Dengan jumlah parameter yang jauh lebih banyak yaitu 5.937.228 dan lama *training* 2529 detik, hasil akurasi tes yang didapatkan masih lebih rendah 1% daripada EfficientNetV2-B0 dengan *width* dan *depth coefficient* 0.5 yang waktu training dan jumlah parameternya jauh lebih rendah.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Test Accuracy (%) | Train time (seconds) | Total parameters |
| EfficientNetV2  (0 block) | 28.41 | 278 | 797K |
| EfficientNetV2  (1 block) | 28.86 | **214** | 797K |
| EfficientNetV2  (2 block) | 30.91 | 264 | 796K |
| EfficientNetV2  (3 block) | 30.13 | 258 | 793K |
| EfficientNetV2  (4 block) | 29.13 | 275 | **790K** |
| InceptionV3 (Schuler et al.) | **64.90** | 4005 | 4.83M |

**Tabel 5:** Test accuracy and training time untuk Cropped PlantDoc Dataset

|  |  |  |  |
| --- | --- | --- | --- |
| Architecture | Test Accuracy (%) | Train time (seconds) | Total parameters |
| 0%L + 100%AB | 25.55 | 229 | 797K |
| 20%L + 80%AB | **30** | 288 | 788K |
| 50%L + 50%AB | 28 | 278 | **785K** |
| 80%L + 20%AB | 30.5 | 275 | 790K |
| 100%L + 0%AB | 25.48 | **231** | 797K |

**Tabel 6:** Efek perbandingan filter pada Cropped PlantDoc Dataset

Berdasarkan percobaan kami, model EfficientNetV2 masih belum mampu bekerja dengan baik pada Cropped PlantDoc dataset. Kami menduga hal ini terjadi karena perbedaan arsitektur yang cukup jauh antara EfficientNetV2 dan Inception V3.

# conclusion

Pada dataset PlantVillage, arsitektur EfficientNetV2 masih belum mampu mengungguli model InceptionV3 milik Schuler et al. [[8]](#eight) dalam hal akurasi. Namun, keunggulan utama EfficientNetV2 terletak pada waktu training yang jauh lebih cepat dibandingkan InceptionV3, menunjukkan efisiensi yang lebih tinggi dalam komputasi. Di sisi lain, pada dataset PlantDoc, EfficientNetV2 mengalami overfitting yang signifikan, dengan akurasi hanya mencapai 30%, jauh di bawah InceptionV3 yang berhasil mencapai 65%.

Berdasarkan penelitian yang kami lakukan, memecah 4 *block* pertama dari EfficientNetV2 terbukti memberikan hasil yang paling optimal. Jumlah *block* ini memungkinkan model untuk memanfaatkan informasi dari channel kromatik dan akromatik secara lebih efektif, sehingga meningkatkan akurasi keseluruhan model dalam klasifikasi penyakit tanaman.

Penelitian menunjukkan bahwa distribusi filter yang paling efektif antara kedua cabang EfficientNetV2 adalah dengan mengalokasikan 50% hingga 80% filter ke cabang AB. Proporsi ini memberikan keseimbangan yang optimal antara pemrosesan informasi kromatik dan akromatik, yang pada akhirnya menghasilkan peningkatan akurasi model yang signifikan dalam tugas klasifikasi penyakit tanaman.

##### References

1. A. Adelaja and J. George, “Food and Agricultural Security: An introduction to the special issue,” *Sustainability*, vol. 13, no. 21, p. 12129, Nov. 2021, doi: 10.3390/su132112129.
2. G. A. Gina, N. A. Mariya, N. C. Natalia, N. S. Nispuana, N. M. F. Wijaya, and N. M. Y. Phalepi, “THE ROLE OF THE AGRICULTURAL SECTOR ON ECONOMIC GROWTH IN INDONESIA,” *Indonesian Journal of Multidisciplinary Sciences*, vol. 2, no. 1, pp. 167–179, Jun. 2023, doi: 10.59066/ijoms.v2i1.325.
3. B. Richard, A. Qi, and B. D. L. Fitt, “Control of crop diseases through Integrated Crop Management to deliver climate‐smart farming systems for low‐ and high‐input crop production,” *Plant Pathology*, vol. 71, no. 1, pp. 187–206, Oct. 2021, doi: 10.1111/ppa.13493.
4. S. Mathulaprangsan, K. Lanthong, D. Jetpipattanapong, S. Sateanpattanakul, and S. Patarapuwadol, “Rice diseases recognition using effective deep learning models,” *2020 Joint International Conference on Digital Arts, Media and Technology With ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON)*, Mar. 2020, doi: 10.1109/ectidamtncon48261.2020.9090709.
5. H. G. C, E. Dhinesh, and A. Jagan, “Detection of leaf disease using principal component analysis and linear support vector machine,” *2019 11th International Conference on Advanced Computing (ICoAC)*, Dec. 2019, doi: 10.1109/icoac48765.2019.246866.
6. W. B. Demilie, “Plant disease detection and classification techniques: a comparative study of the performances,” *Journal of Big Data*, vol. 11, no. 1, Jan. 2024, doi: 10.1186/s40537-023-00863-9.
7. Chaudhary, P., Chaudhari, A., and Go- dara, S. Color transform based approach for dis- ease spot detection on plant leaf. International Journal of Computer Science and Telecommuni- cations 3 (06 2012), 65–70.
8. J. P. S. Schuler, S. Romani, M. Abdel-Nasser, H. Rashwan, and D. Puig, “Color-Aware Two-Branch DCNN for efficient plant disease classification,” *Mendel ... (Brno. On-line)/Mendel ...*, vol. 28, no. 1, pp. 55–62, Jun. 2022, doi: 10.13164/mendel.2022.1.055.
9. Y. Toda and F. Okura, “How convolutional neural networks diagnose plant Disease,” *Plant Phenomics*, vol. 2019, Jan. 2019, doi: 10.34133/2019/9237136.
10. Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, “Plant leaf disease classification using EfficientNet deep learning model,” *Ecological Informatics*, vol. 61, p. 101182, Mar. 2021, doi: 10.1016/j.ecoinf.2020.101182.
11. G. G and A. P. J, “Identification of plant leaf diseases using a nine-layer deep convolutional neural network,” *Computers & Electrical Engineering*, vol. 76, pp. 323–338, Jun. 2019, doi: 10.1016/j.compeleceng.2019.04.011.
12. A. Ali, “PlantVillage Dataset,” Kaggle, 2019. https://www.kaggle.com/datasets/abdallahalidev/plantvillage-dataset (accessed Jun. 27, 2024).
13. P. Kayal, “pratikkayal/PlantDoc-Object-Detection-Dataset,” GitHub, Nov. 23, 2019. https://github.com/pratikkayal/PlantDoc-Object-Detection-Dataset (accessed Jun. 27, 2024).
14. keras-team, “Keras,” *GitHub*, Mar. 27, 2015. https://github.com/keras-team/keras (accessed Jun. 27, 2024).