

**DETEKSI KEMATANGAN TANDAN BUAH SEGAR (TBS) KELAPA SAWIT  
BERDASARKAN KOMPOSISI WARNA MENGGUNAKAN DEEP LEARNING**

**Muhammad Rifqi<sup>1</sup>, Suharjito<sup>2</sup>**

<sup>1,2</sup> Computer Science Department, BINUS Graduate Program

<sup>1,2</sup> Universitas Bina Nusantara

<sup>1,2</sup> Jakarta, Indonesia 11480

*E-mail:* <sup>1</sup>[muhammad.rifqi008@binus.ac.id](mailto:muhammad.rifqi008@binus.ac.id), <sup>2</sup>[suharjito@binus.edu](mailto:suharjito@binus.edu)

**ABSTRACT**

**Artikel:**

Diterima: 25 November, 2021

Direvisi: 05 Januari, 2022

Diterbitkan: 07 Januari, 2022

**\*Alamat Korespondensi:**

[muhammad.rifqi008@binus.ac.id](mailto:muhammad.rifqi008@binus.ac.id)

Classification of oil palm fresh fruit bunch (FFB) based on maturity is very important for estimating oil content. Traditional methods using human vision to observe color changes during ripening and counting the number of fruits that fall from FFB are not effective. Research for neural architectures to design new network bases and improve them resulted in a set of models called EfficientNet. The most important function is the optimizer. This function repeatedly increases the parameters to reduce loss. In this study, the EfficientNetB0 and B1 models were developed to detect oil palm maturity into 6 classes, Raw, Ripe, Overripe, Underripe, abnormal, and empty bunch using optimizer RMSprop and SGD. From the research results, obtained the highest accuracy using the RMSprop optimizer of 0.9955 using the EfficientNetB0 model and 0.9949 using the EfficientNetB1 model. While using the SGD optimizer, the accuracy achieved is 0.918 using the EfficientNetB0 model and 0.9079 using the EfficientNetB1 model.

**Keywords:** *Maturity, Palm oil, EfficientNet, Optimizer*

**ABSTRAK**

Klasifikasi tandan buah segar (TBS) kelapa sawit berdasarkan kematangan sangat penting untuk memperkirakan kandungan minyak. Cara tradisional menggunakan penglihatan manusia untuk mengamati perubahan warna dalam masa pematangan dan menghitung jumlah buah yang jatuh dari TBS tidaklah efektif. Pencarian arsitektur saraf untuk merancang dasar jaringan baru dan meningkatkannya sehingga menghasilkan sekumpulan model yang disebut EfficientNet. Fungsi yang paling penting adalah *optimizer*. Fungsi ini secara berulang-ulang meningkatkan parameter-parameter untuk mengurangi loss. Pada penelitian ini, dikembangkan model EfficientNetB0 dan B1 untuk mendeteksi kematangan sawit ke dalam 6 kelas, yaitu TBS kurang masak, TBS masak, TBS mentah, TBS terlalu

masak, TBS abnormal dan janjang kosong menggunakan *optimizer* RMSprop dan SGD. Dari hasil penelitian, didapat akurasi tertinggi dengan menggunakan *optimizer* RMSprop sebesar 0.9955 menggunakan model EfficientNetB0 dan 0.9949 menggunakan model EfficientNetB1. Sementara menggunakan *optimizer* SGD, akurasi yang diraih sebesar 0.918 menggunakan model EfficientNetB0 dan 0.9079 menggunakan model EfficientNetB1.

**Kata Kunci:** *Kematangan, Kelapa sawit, EfficientNet, Optimizer*

## I. PENDAHULUAN

*Elaeis guinensis jack* atau yang lebih dikenal dengan sebutan kelapa sawit. Nama tersebut berasal dari bahasa Yunani, yaitu *elaia* yang berarti zaitun. [1]. Hal yang menjadi daya tarik pada perkebunan kelapa sawit terletak pada keuntungan yang berlimpah karena merupakan sumber minyak nabati dan bahan agroindustri [2].

Salah satu indikator kualitas TBS adalah kadar asam lemak bebas (ALB), dimana kadar ini tidak boleh melebihi 2-3% [3]. Perbedaan kematangan ini dapat dilihat dari perubahan warnanya, dimana warna merah menandakan matang, oranye kemerahan menandakan terlewat matang, hitam kemerahan kurang matang, dan hitam keunguan menunjukkan mentah [4]. Cara tradisional menggunakan penglihatan manusia untuk mengamati perubahan warna dalam masa pematangan dan menghitung jumlah buah yang jatuh dari TBS tidaklah efektif, kurang objektif, dan memakan biaya pekerja yang tinggi. [5].

Banyak riset yang sudah didedikasikan untuk menentukan kematangan dan kualitas kelapa sawit, khususnya menggunakan metode *computer vision*. Salah satu pendekatan terbaik untuk melakukan penentuan kematangan buah kelapa sawit dengan menggunakan model *deep learning* [6].

*Deep learning* telah menunjukkan hasil yang menjanjikan untuk mengklasifikasikan gambar, bahkan untuk kasus yang kompleks sekalipun. Dengan penekanan khusus pada penggunaan di sektor kelapa sawit, dimana penggunaan informasi dari kecerdasan buatan dalam proses produksi minyak dapat mengarah pada perbaikan besar terhadap dampak ekonomi dan lingkungan [7].

Convolutional Neural Networks (CNN) dirancang secara khusus untuk bekerja pada gambar. Dengan CNN, kernel atau filter adalah matriks persegi kecil yang digunakan untuk menampakkan fitur tertentu dari gambar yang dimasukkan. Sehingga pada akhirnya, hasilnya akan dijumlahkan untuk membuat nilai tunggal pada *output* [8].

Berdasarkan pengamatan bahwa akurasi dan efisiensi dapat lebih baik diperoleh dengan memberlakukan keseimbangan antara semua dimensi jaringan. Berdasarkan penelitian, kualitas yang lebih baik bisa dicapai dengan menyeimbangkan kedalaman jaringan, luasnya dan resolusinya. [9]. Lebih lanjut, [9] menggunakan pencarian arsitektur saraf untuk merancang dasar jaringan baru dan meningkatkannya sehingga menghasilkan sekumpulan model yang disebut EfficientNet yang dinamakan dengan EfficientNetB0 – EfficientNetB7. Semakin tinggi nilai EfficientNet yang digunakan, maka akan semakin kompleks model yang terbuat, namun akan dapat menangani gambar dengan resolusi yang lebih luas.

Perbedaan lainnya dari EfficientNet ini adalah, jumlah parameter yang digunakan. EfficientNet tidak hanya memiliki akurasi klasifikasi yang baik, tapi juga memiliki parameter model yang lebih sedikit. Hal ini membuat model yang berbasis EfficientNet lebih cocok untuk kumpulan data yang kecil. Karena memiliki jumlah model yang sedikit, EfficientNet juga lebih mudah untuk di *deploy* ke sistem sesungguhnya [10]. EfficientNet bisa lebih efektif dalam memanfaatkan sumber daya yang terbatas untuk menghasilkan hasil klasifikasi yang lebih akurat, lebih cepat dari metode yang ada sebelumnya seperti VGG16 dan ResNet [11].

Setelah melakukan penambahan *layer* kedalam model, kita perlu mengatur *score function* dan *loss function*, dan algoritma pengoptimalan. Fungsi yang paling penting adalah optimizer. Fungsi ini secara berulang-ulang meningkatkan parameter-parameter (nilai filter kernel, *weight*, bias dari neuron untuk mengurangi *loss*) [12]. Beberapa optimizer yang dapat digunakan diantaranya adalah RMSprop, Adam, Nadam [13].

Berdasarkan masalah diatas, maka dilakukanlah penelitian untuk mengembangkan model guna mengetahui hasil dari kematangan TBS kelapa sawit yaitu pada TBS kurang masak, TBS masak, TBS mentah, TBS terlalu masak, TBS abnormal dan janjang kosong. Kemudian dilakukan pengklasifikasian menggunakan metode CNN [14] menggunakan model EfficientNet. Model EfficientNet dipilih karena EfficientNet cocok digunakan untuk kumpulan data yang kecil, karena jumlah data yang digunakan sebanyak 8976 data. Model EfficientNet yang digunakan pada penelitian ini adalah EfficientNetB0 dan EfficientNetB1 untuk melihat pengaruh jumlah parameter terhadap akurasi dari model yang dikembangkan. Untuk meningkatkan performa model yang dihasilkan, maka akan digunakan *optimizer* RMSprop yang mendapatkan hasil yang lebih baik dibandingkan optimizer lainnya berdasarkan [15].

Untuk mendeteksi buah yang mentah dan matang, [16] melakukan penelitian dengan mengumpulkan gambar kelapa sawit untuk diketahui tingkat kematangannya dengan menggunakan fitur warna HSI. Nilai HSI didapat dari mengkonversi nilai RGB. [17] menggunakan tiga metode ekstraksi warna, yaitu warna histogram, momen warna, dan warna correlogram untuk kemudian diekstraksi masing-masing ciri warnanya untuk mengetahui perbedaan tingkat kematangannya berdasarkan perbedaan warnanya.

Dengan menggunakan ekstraksi fitur warna yang berbeda, [18] menggunakan warna rona (*hue*), karena pengukuran rona terbukti menjadi yang terbaik untuk buah kelapa sawit dibandingkan dengan RGB dan nilai CIE<sub>x,y</sub>. Penelitian lainnya dilakukan oleh [19] untuk mendeteksi kematangan sawit yang mentah, kurang matang, dan matang. Gambar diakuisi dengan menggunakan kamera digital dari ponsel pintar. [20] juga melakukan pendeteksian kematangan TBS kelapa sawit menggunakan ekstraksi fitur warna, yaitu RGB.

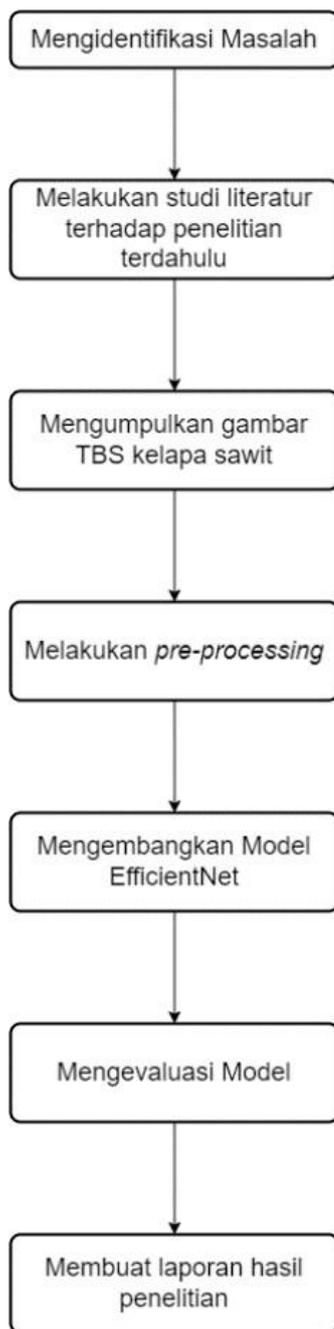
Nilai RGB dijadikan sebagai *input* dari sistem penilaian. Selanjutnya, sistem akan memproses gambar, terdiri dari pemrosesan gambar, dan menghitung rata-rata nilai RGB. Penelitian menggunakan ekstraksi warna RGB dan HSV dilakukan oleh [21] untuk menentukan tingkat kematangan TBS kelapa sawit. Dengan menggunakan algoritma *K-Means Clustering* dilakukan proses pengklasifikasian 3 tingkat kematangan kelapa sawit, yaitu matang, cukup matang dan mentah.

Studi lainnya untuk mendeteksi kematangan sawit menggunakan fitur ekstraksi warna dilakukan oleh [22]. Metode ekstraksi warna yang digunakan adalah HSV. [23] melakukan pembuatan aplikasi berbasis mobile, untuk mengklasifikasikan tingkat kematangan kelapa sawit menggunakan CNN. Pada penelitian ini, penulis melakukan proses *transfer learning* menggunakan ImageNet pada 4 model CNN *lightweight*, dengan augmentasi data yang dinamakan “*9-angle crop*”. Dari penelitian tersebut, didapatkan akurasi sebesar 0.898 menggunakan model EfficientNetB0.

## II. METODOLOGI

### 2.1 Tahapan Penelitian

Pada penelitian ini terdapat 7 tahapan. Diantaranya; Mengidentifikasi masalah, melakukan studi literatur terhadap penelitian terdahulu, mengumpulkan gambar TBS kelapa sawit, melakukan *pre-processing*, mengembangkan model EfficientNet, melakukan evaluasi dari model yang telah dikembangkan dan membuat laporan hasil penelitian. Ilustrasi tahapan penelitian dapat dilihat pada gambar 1



Gambar 1. Tahapan penelitian

Tahap awal dari penelitian ini mengidentifikasi masalah yang terjadi. Dimana pada saat ini klasifikasi TBS kelapa sawit berdasarkan kematangan sangat penting untuk memperkirakan kandungan minyak. Cara tradisional menggunakan penglihatan manusia untuk mengamati perubahan warna dalam masa pematangan dan menghitung jumlah buah yang jatuh dari TBS tidaklah efektif, kurang objektif, dan memakan biaya pekerja yang tinggi.

Sehingga dilakukanlah studi literatur terhadap penelitian terdahulu. Berdasarkan studi literatur, salah satu pendekatan terbaik

untuk melakukan penentuan kematangan buah kelapa sawit dengan menggunakan model deep learning. Deep learning telah menunjukkan hasil yang menjanjikan untuk mengklasifikasikan gambar, bahkan untuk kasus yang kompleks sekalipun.

Kemudian, dilakukan pengambilan gambar yang akan dijadikan sebagai data training yang akan digunakan untuk penelitian. Gambar diambil menggunakan kamera ponsel RealMe 5i yang memiliki resolusi sebesar 12 megapiksel. Pengambilan gambar dilakukan mulai dari jam 7.00 pagi sampai jam 17.00 sore. Hal ini dilakukan agar gambar mendapatkan cahaya yang cukup. Gambar sawit yang digunakan adalah TBS yang telah dipanen, agar memudahkan proses pengambilan gambar. Pengambilan gambar TBS kelapa sawit dilakukan dengan mengambil gambar tampak depan, belakang, kiri dan kanan, agar mendapatkan gambar yang bervariasi dan banyak.

Agar gambar yang didapat bisa digunakan, gambar harus melalui tahap pre-processing terlebih dahulu. Diantaranya melakukan pelabelan data yang telah didapat berdasarkan pengetahuan yang didapat oleh narasumber pada penelitian ini yang merupakan asisten kebun di salah satu perkebunan kelapa sawit yang ada di provinsi Riau. Selanjutnya, gambar yang didapat akan dilakukan proses resizing, menjadi 224x224x3. Dikarenakan, gambar yang bisa digunakan dengan model yang akan dikembangkan pada penelitian ini menerima gambar dengan resolusi maksimal 224x224x3. Selanjutnya, untuk mendapatkan data yang cukup untuk pemrosesan data, dilakukan augmentasi data untuk mendapatkan gambar lebih banyak dari gambar yang sudah ada.

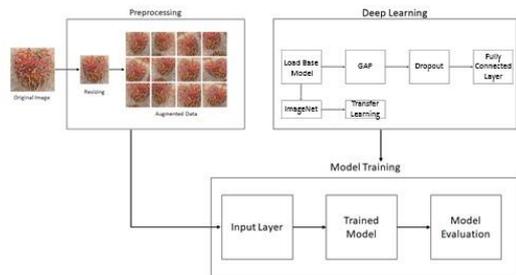
Selanjutnya, dengan data training yang ada, kemudian dikembangkanlah model yang akan digunakan yaitu EfficientNet. EfficientNet merupakan kumpulan model yang telah dikembangkan oleh [9] yang terdiri dari EfficientNetB0-B7. Pada penelitian ini menggunakan model EfficientNetB0 dan EfficientNetB1 untuk diketahui akurasi modelnya. Penelitian ini akan melihat apakah jumlah parameter yang berbeda akan berpengaruh terhadap hasil akurasi model yang didapat.

Dengan hasil yang didapat, kemudian dilakukan analisa terkait model yang telah dibangun. Analisa yang dilakukan dengan melihat hasil akurasi dari model yang telah

dikembangkan. Analisa tersebut diharapkan dapat memberi kontribusi baik kedunia penelitian maupun dunia perkebunan kelapa sawit.

## 2.2 Model Yang Diusulkan

Model yang diusulkan adalah pada penelitian ini terdiri dari beberapa tahapan yang dapat dilihat dari Gambar 2.



Gambar 2. Model yang diusulkan

Pertama dengan melakukan instalasi paket yang dibutuhkan. Kemudian, model yang akan digunakan dimuat, pada hal ini EfficientNetB0 dan EfficientNetB1 yang dilengkapi dengan transfer learning, selanjutnya ditambahkan model sequential yang berguna untuk memastikan setiap lapisan memiliki satu tensor *input* dan satu tensor *output*. Kemudian ditambahkan beberapa *layer* untuk menyesuaikan dengan data yang ada. Diantara *layer GlobalMaxPooling2D* yang digunakan untuk menyeragamkan input kedalam 3 nilai maksimal untuk setiap *channel*. Untuk mencegah terjadinya *overfitting*, ditambahkan *layer Dropout* yang secara acak menetapkan unit input ke 0 dengan frekuensi laju pada setiap Langkah pelatihan. Dengan demikian akan didapatkan jumlah parameter yang tersedia dari parameter yang dibangun.

Kemudian, data yang telah melalui tahap preprocessing, akan dimuat agar bisa diproses oleh model. Sebelum dimuat, data terlebih dahulu di bagi kedalam folder training, validation dan testing dengan rasio 7:12. Selanjutnya, untuk mengurangi loss, ditambahkan optimizer yaitu sebuah metode atau algoritma yang digunakan untuk mengubah attribut dari neural network seperti weight dan learning ratenya.

Selanjutnya, model telah dapat dibangun dengan menggunakan data training dan data validasi. Dengan menentukan epoch, yaitu hyperparameter yang menentukan berapa kali algoritma pembelajaran akan bekerja melalui

seluruh training dataset. Semakin tinggi nilai epoch-nya, maka akan semakin baik akurasi yang didapat, namun akan semakin Panjang prosesnya dan tentu saja akan memakan waktu lebih lama. Hyperparameter lainnya yang digunakan adalah learning rate, yaitu parameter yang akan menentukan seberapa cepat jaringan memperbaharui parameternya. Jika nilai learning rate rendah, akan memperlambat proses pembelajaran tetapi dapat memperbaharui jaringan dengan lancar. Sebaliknya, jika nilai learning rate tinggi, akan mempercepat proses pembelajaran tetapi lambat dalam memperbaharui jaringan. Lalu ada batch size, yaitu jumlah sub sample yang diberikan ke dalam jaringan setelah pembaharuan jaringan terjadi. Nilai default untuk *batch\_size* adalah 32. Setelah model dibangun, maka akan didapat nilai akurasi dan loss dari model yang dibangun terhadap data yang digunakan. Kemudian model yang telah dibangun, dapat dievaluasi untuk didapatkan nilai akurasinya.

## III. HASIL DAN PEMBAHASAN

### 3.1 Dataset

Dataset yang digunakan terdiri dari 6 class, diantaranya TBS kurang masak, TBS masak, TBS mentah, TBS terlalu masak, TBS upnormal dan janjang kosong. Jumlah keseluruhan data yang digunakan sebanyak 8976 gambar dengan masing-masing class memiliki 1496 gambar. Representasi yang digunakan pada penelitian ini dapat dilihat pada gambar berikut



Gambar 3. Dataset sawit yang digunakan

Gambar-gambar tersebut adalah TBS yang telah diambil dari pohonnya. Namun, gambar yang didapat masih jauh dari kata cukup untuk diproses menggunakan deep learning, sehingga perlu dilakukan proses augmentasi.

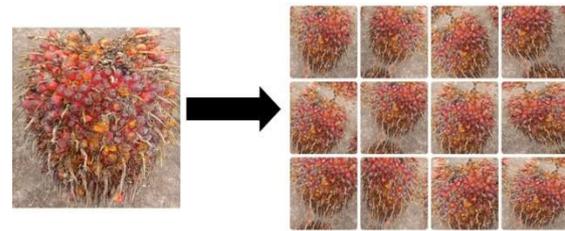
### 3.2 Pre-Processing

Dari dataset yang dimiliki, dataset tersebut dibagi kedalam 3 folder yaitu train, val dan test dengan rasio pembagian 7:1:2. Untuk mendukung pembuatan model yang lebih baik, perlu dilakukan metode augmentasi untuk memperbanyak gambar. Dengan ketentuan sebagai berikut:

*Tabel 1. Detail augmentasi gambar*

Parameter	Nilai
Rotation_range	45
Width_shift_range	0.2
Height_shift_range	0.2
Shear_range	0.2
Zoom_range	0.2
Horizontal_flip	True
Fill mode	nearest

Proses augmentasi gambar diawali dengan menyediakan gambar yang akan diaugmentasi seperti yang ditunjukkan pada Gambar 4.



Gambar 4. Data sebelum dan sesudah di augmentasi

Selanjutnya, gambar akan akan diaugmentasi dengan nilai variable seperti yang ada pada tabel 1. Gambar akan diputar secara acak sebesar 45 derajat. *Width\_shift\_range* berarti gambar akan menggeser posisi gambar ke kiri atau ke kanan (horizontal) sebesar 20%. Sementara *height\_shift\_range* berarti gambar akan digeser posisinya secara ke atas dan kebawah (vertical) sebesar 20%. *Shear\_range* berarti bahwasanya gambar akan terdistorsi sepanjang sumbu sebesar 20%.

*Zoom\_range* berarti bahwa gambar akan diperbesar secara acak sebesar 20%. Sementara jika *horizontal\_flip* bernilai *true*, maka gambar hasil augmentasi akan diputar balik secara horizontal. *Fill\_mode* adalah proses pengisian gambar yang kosong, terdiri dari 4, yaitu *constant*, *nearest*, *reflect* dan *wrap*. *Nearest* dipilih, karena jika gambar melalui proses augmentasi, bagian gambar yang kosong akan disesuaikan dengan piksel terdekat. Hasil augmentasi dapat dilihat pada gambar 4.

### 3.3 Hasil Evaluasi

Performa dari setiap model dibandingkan dengan menggunakan 2 matriks, yaitu klasifikasi akurasi dan klasifikasi *loss*. Dimana model yang terbaik akan didapat dari melihat hasil akurasi. Untuk mendapatkan model dengan akurasi yang baik, perlu dilakukan penyesuaian model. Mulai dari menambahkan layer untuk menyesuaikan dengan menambahkan beberapa layer, diantaranya model *sequential* yang berguna untuk memastikan setiap lapisan memiliki satu tensor *input* dan satu tensor *output*. Kemudian ditambahkan beberapa *layer* untuk menyesuaikan dengan data yang ada. Diantara *layer GlobalMaxPooling2D* yang digunakan untuk menyeragamkan input kedalam 3 nilai maksimal untuk setiap *channel*. Untuk mencegah terjadinya *overfitting*, ditambahkan *layer Dropout* yang secara acak menetapkan unit input ke 0 dengan frekuensi laju pada setiap

Langkah pelatihan. Dari model yang telah dibangun, didapat kesimpulan pada tabel 2:

*Tabel 2. Rangkuman model yang digunakan*

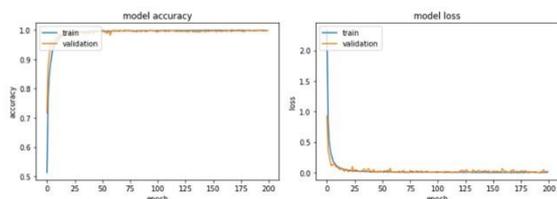
Model	Total Parameter	Trainable Parameter	Non-trainable parameter
EfficientNet B0	4,057,257	4,015,234	42,023
EfficientNet B1	6,582,925	6,520,870	62,055

Selanjutnya, untuk mengurangi *loss*, ditambahkan *optimizer* yaitu sebuah metode atau algoritma yang digunakan untuk mengubah atribut dari *neural network* seperti *weight* dan *learning ratenya*. Untuk konfigurasi hyperparameter yang digunakan oleh penulis dapat dilihat pada Tabel 3.

*Tabel 3. Konfigurasi Hyperparameter*

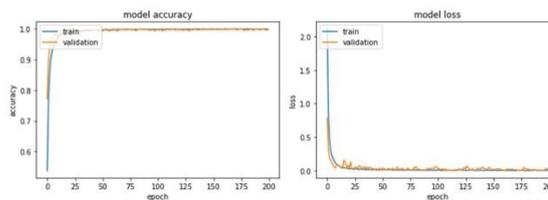
Hyperparameter	Nilai yang digunakan
Optimizer	RMSprop
Learning rate	1e-4
Matriks	Acc
Epoch	200
Batch Size	20

Dengan *data training* yang ada, kemudian dibangunlah model untuk diketahui akurasi modelnya, pada penelitian ini model yang digunakan adalah EfficientNetB0 dan EfficientNetB1. Perbandingan hasil model yang dibangun antara optimizer yang berbeda, yakni RMSprop dan SGD juga dilakukan.



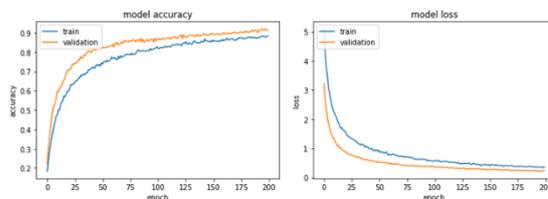
Gambar 5. Model accuracy dan model loss EfficientNetB0 dengan Optimizer RMSprop

Berdasarkan Gambar 5, model EfficientNetB0 yang dibangun menggunakan optimizer RMSprop menunjukkan mulai *convergence* saat memasuki epoch lebih dari 25. Begitu pun untuk nilai *loss* mulai *convergence* di epoch lebih dari 25.



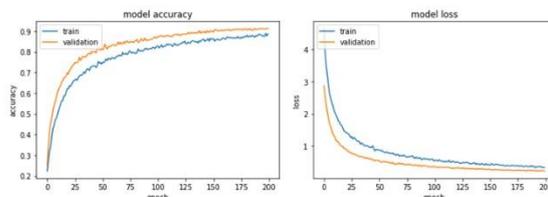
Gambar 6. Model accuracy dan model loss EfficientNetB1 dengan Optimizer RMSprop

Berdasarkan gambar 6, model EfficientNetB1 yang dibangun menggunakan optimizer RMSprop menunjukkan mulai *convergence* saat memasuki epoch lebih dari 25. Begitu pun untuk nilai *loss* mulai *convergence* di epoch lebih dari 25.



Gambar 7. Model Accuracy dan Model Loss EfficientNetB0 dengan Optimizer SGD

Berdasarkan gambar III.4, model EfficientNetB0 yang dibangun menggunakan optimizer SGD menunjukkan mulai *convergence* saat memasuki epoch lebih dari 75. Begitu pun untuk nilai *loss* mulai *convergence* di epoch lebih dari 75.



Gambar 8. Model accuracy dan model loss EfficientNetB1 dengan Optimizer SGD

Berdasarkan Gambar 8, model EfficientNetB0 yang dibangun menggunakan optimizer SGD menunjukkan mulai *convergence* saat memasuki epoch lebih dari 75. Begitu pun untuk nilai *loss* mulai *convergence* di epoch lebih dari 75.

Dari Gambar 5-8 dapat dilihat jika model yang dibangun menggunakan optimizer RMSprop menunjukkan hasil yang lebih baik, lebih awal mengalami *convergence* dibandingkan dengan model yang dibangun menggunakan optimizer SGD.

### 3.4 Performa Evaluasi dari Model yang Diusulkan

Untuk melakukan uji performa dari model yang diusulkan, digunakan nilai akurasi menggunakan Bahasa pemrograman python yang penulis gunakan untuk membangun model pada penelitian ini. Dengan menggunakan kode yang terdapat pada Gambar 9.

```
score = model.evaluate(test_generator, verbose=1)
```

Gambar 9. Kode nilai akurasi

Hasil penghitungan akurasi nantinya akan dibandingkan dengan model EfficientNetB0 yang telah dilakukan oleh [23]. Perbandingan hasil akurasi berbagai jenis model dengan variasi optimizer yang digunakan dapat dilihat pada tabel tabel 4:

Tabel 4. Hasil Akurasi

Model Yang Digunakan	Akurasi
EfficientNetB0 oleh [23]	0.898
EfficientNetB0, Optimizer RMSprop	0.9955
EfficientNetB1, Optimizer RMSprop	0.9949
EfficientNetB0, Optimizer SGD	0.918
EfficientNetB1, Optimizer SGD	0.9079

Dari Tabel 4 diatas, dapat dilihat hasil yang didapat oleh model EfficientNetB0 dan EfficientNetB1 memiliki akurasi tertinggi saat menggunakan optimizer RMSprop dengan hasil 0.9955 dengan EfficientNetB0 dan 0.9949. Dibanding dengan menggunakan optimizer SGD akurasi yang didapat sebesar 0.918 dengan EfficientNetB0 dan 0.9079 menggunakan EfficientNetB1.

Dari 4 model yang telah dikembangkan, didapat perbandingan hasil kecepatan evaluasi antar model tersebut, yang dapat dilihat pada Tabel 5:

Tabel 5. Kecepatan evaluasi model

Model	Kecepatan Evaluasi
EfficientNetB0, Optimizer RMSprop	148 detik
EfficientNetB0, Optimizer RMSprop	210 detik

EfficientNetB0, Optimizer SGD	149 detik
EfficientNetB1, Optimizer SGD	210 detik

Berdasarkan Tabel 5, jumlah parameter yang membedakan antara EfficientNetB0 dengan EfficientNetB1 berpengaruh terhadap kecepatan evaluasi dari model yang telah dikembangkan. Dimana model EfficientNetB1 62 detik lebih lama dibandingkan model EfficientNetB0 dengan menggunakan *optimizer* RMSprop. Sedangkan menggunakan *optimizer* SGD, EfficientNetB1 61 detik lebih lama dibandingkan model EfficientNetB1.

## IV. PENUTUP

Dari penelitian yang telah dilakukan, didapatkan hasil Model EfficientNet telah berhasil dibangun dengan menggunakan dataset yang terdiri dari TBS kurang masak, TBS masak, TBS mentah, TBS terlalu mentah, TBS upnormal dan janjang kosong. Dari model EfficientNet yang dibangun, didapat hasil evaluasi model EfficientNetB0 0.9893 dan model EfficientNetB1 0.9949. jumlah parameter yang membedakan antara EfficientNetB0 dengan EfficientNetB1 berpengaruh terhadap kecepatan evaluasi dari model yang telah dikembangkan. Dimana model EfficientNetB1 62 detik lebih lama dibandingkan model EfficientNetB0 dengan menggunakan *optimizer* RMSprop. Sedangkan menggunakan *optimizer* SGD, EfficientNetB1 61 detik lebih lama dibandingkan model EfficientNetB1.

## DAFTAR PUSTAKA

- [1] I. Pahan, *Panduan Teknis Budidaya Kelapa Sawit Untuk Praktisi Perkebunan*, I. Jakart: Penebar Swadaya, 2015.
- [2] S. ITN, *58 Kiat Meningkatkan Produktivitas dan Mutu Kelapa Sawit*, I. Jakarta: Penebar Swadaya, 2008.
- [3] A. W. Krisdiarto, L. Sutiarto, and K. H. Widodo, "Optimasi Kualitas Tandan Buah Segar Kelapa Sawit dalam Proses Panen-Angkut Menggunakan Model Dinamis," *Agritech*, vol. 37, no. 1, p. 102, 2017, doi: 10.22146/agritech.17015.
- [4] Z. Ibrahim, N. Sabri, and D. Isa, "Palm

- oil fresh fruit bunch ripeness grading recognition using convolutional neural network,” *J. Telecommun. Electron. Comput. Eng.*, vol. 10, no. 3–2, pp. 109–113, 2018.
- [5] M. Shiddiq, Fitmawati, R. Anjasmara, N. Sari, and Hefniati, “Ripeness detection simulation of oil palm fruit bunches using laser-based imaging system,” *AIP Conf. Proc.*, vol. 1801, no. January 2017, 2017, doi: 10.1063/1.4973101.
- [6] H. Herman, A. Susanto, T. W. Cenggoro, S. Suharjito, and B. Pardamean, “Oil Palm Fruit Image Ripeness Classification with Computer Vision using Deep Learning and Visual Attention,” *J. Telecommun. Electron. Comput. Eng.*, vol. 12, no. 2, pp. 21–27, 2020.
- [7] I. Bonet, F. Caraffini, A. Pena, A. Puerta, and M. Gongora, “Oil Palm Detection via Deep Transfer Learning,” *2020 IEEE Congr. Evol. Comput. CEC 2020 - Conf. Proc.*, 2020, doi: 10.1109/CEC48606.2020.9185838.
- [8] L. T. Duong, P. T. Nguyen, C. Di Sipio, and D. Di Ruscio, “Automated fruit recognition using EfficientNet and MixNet,” *Comput. Electron. Agric.*, vol. 171, no. February, p. 105326, 2020, doi: 10.1016/j.compag.2020.105326.
- [9] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.
- [10] P. Zhao and L. Huang, “MULTI-ASPECT SAR TARGET RECOGNITION BASED ON EFFICIENTNET AND GRU,” pp. 1651–1654, 2020.
- [11] Z. Jiahao, Y. Jiang, R. Huang, and J. Shi, “EfficientNet-Based Model with Test Time Augmentation for Cancer Detection,” *2021 IEEE 2nd Int. Conf. Big Data, Artif. Intell. Internet Things Eng. ICBAIE 2021*, no. Icbaie, pp. 548–551, 2021, doi: 10.1109/ICBAIE52039.2021.9389825.
- [12] R. V. K. Reddy, B. Srinivasa Rao, and K. P. Raju, “Handwritten Hindi Digits Recognition Using Convolutional Neural Network with RMSprop Optimization,” *Proc. 2nd Int. Conf. Intell. Comput. Control Syst. ICICCS 2018*, no. Iccics, pp. 45–51, 2019, doi: 10.1109/ICCONS.2018.8662969.
- [13] I. Joshi, M. Desai, S. Bookseller, R. Mohod, C. N. Paunwala, and B. Vaidya, “Performance of Different Optimizers,” *2019 10th Int. Conf. Comput. Commun. Netw. Technol.*, pp. 1–7, 2019.
- [14] I. M. Wismadi, D. C. Khrisne, and I. M. A. Suyadnya, “Detecting the Ripeness of Harvest-Ready Dragon Fruit using Smaller VGGNet-Like Network,” *J. Electr. Electron. Informatics*, vol. 3, no. 2, p. 35, 2020, doi: 10.24843/jeei.2019.v03.i02.p01.
- [15] R. Zaheer and H. Shaziya, “A Study of the Optimization Algorithms in Deep Learning,” *Proc. 3rd Int. Conf. Inven. Syst. Control. ICISC 2019*, no. Icisc, pp. 536–539, 2019, doi: 10.1109/ICISC44355.2019.9036442.
- [16] M. K. Shabdin, A. R. M. Shariff, M. N. A. Johari, N. K. Saat, and Z. Abbas, “A study on the oil palm fresh fruit bunch (FFB) ripeness detection by using Hue, Saturation and Intensity (HSI) approach,” *IOP Conf. Ser. Earth Environ. Sci.*, vol. 37, no. 1, 2016, doi: 10.1088/1755-1315/37/1/012039.
- [17] N. Sabri, Z. Ibrahim, S. Syahlan, N. Jamil, and N. N. A. Mangshor, “Palm oil fresh fruit bunch ripeness grading identification using color features,” *J. Fundam. Appl. Sci.*, vol. 9, no. 4S, p. 563, 2018, doi: 10.4314/jfas.v9i4s.32.
- [18] S. A. Ghazali, H. Selamat, Z. Omar, and R. Yusof, “Image Analysis Techniques for Ripeness Detection of Palm Oil Fresh Fruit Bunches,” *Elektr. J. Electr. Eng.*, vol. 18, no. 3, pp. 57–62, 2019, doi: 10.11113/elektrika.v18n3.192.
- [19] A. Septiarini, H. Hamdani, H. R. Hatta, and A. A. Kasim, “Image-based processing for ripeness classification of oil palm fruit,” *Proceeding - 2019 5th Int. Conf. Sci. Inf. Technol. Embrac. Ind. 4.0 Towar. Innov. Cyber Phys. Syst. ICSITech 2019*, pp. 23–26, 2019, doi: 10.1109/ICSITech46713.2019.8987575.
- [20] N. F. Sulaiman, O. Pauline, L. W. Kiow, L. K. Huong, and G. S. Fong, “CCAM Communications in Computational and Applied Automatic Grading System for Oil Palm Fruit Ripeness,” *Commun.*

- Comput. Appl. Math.*, vol. 2, no. 1, pp. 7–11, 2020, [Online]. Available: <http://fazpublishing.com/ccam/index.php/ccam/article/view/32>.
- [21] E. F. Himmah, M. Widyaningsih, and M. Maysaroh, “Identifikasi Kematangan Buah Kelapa Sawit Berdasarkan Warna RGB Dan HSV Menggunakan Metode K-Means Clustering,” *J. Sains dan Inform.*, vol. 6, no. 2, pp. 193–202, 2020, doi: 10.34128/jsi.v6i2.242.
- [22] A. W. Setiawan and O. E. Prasetya, “Palm Oil Fresh Fruit Bunch Grading System Using Multispectral Image Analysis in HSV,” *2020 IEEE Int. Conf. Informatics, IoT, Enabling Technol. ICIoT 2020*, pp. 85–88, 2020, doi: 10.1109/ICIoT48696.2020.9089431.
- [23] G. N. Elwirehardja, J. S. Prayoga, and Suharjito, “Oil palm fresh fruit bunch ripeness classification on mobile devices using deep learning approaches,” *Comput. Electron. Agric.*, vol. 188, no. March, p. 106359, 2021, doi: 10.1016/j.compag.2021.106359.