

RESEARCH ARTICLE

Mendeteksi Kematangan Buah Kelapa Sawit Menggunakan *Convolution Neural Network Deep Learning*

Raihan Triyogi, Rita Magdalena* and Bambang Hidayat

Fakultas Teknik Elektro, Universitas Telkom, Bandung, 40257, Jawa Barat, Indonesia

* Corresponding author: ritamagdalen@telkomuniversity.ac.id

Received on 16 September 2023; accepted on 16 October 2023

Abstrak

Minyak kelapa sawit adalah bahan utama dalam produksi minyak goreng dan juga digunakan dalam produksi sabun, lilin, kosmetik, tinta, dan pasta gigi. Minyak kelapa sawit dihasilkan dari buah sawit yang telah matang. Penentuan kematangan kelapa sawit dilihat dari bentuk dan warnanya, yang bisa ditentukan oleh orang yang berpengalaman. Untuk menghindari ketergantungan terhadap seseorang serta mempercepat proses deteksi kematangan, dirancang sistem dengan metode *Convolutional Neural Network* (CNN) untuk melakukan klasifikasi kematangan buah kelapa sawit. Tugas Akhir ini menggunakan dataset buah kelapa sawit yang terdiri dari 3 kelas, kelas mentah, matang dan busuk dengan masing-masing kelas terdapat 100 citra sehingga total terdapat 300 citra. Dataset tersebut digunakan sebagai data *train*, data *validation*, dan data *test* dengan distribusi persentase sebanyak 65% data *train*, 20% data *validation*, dan 15% data *test*. Penelitian ini menggunakan CNN arsitektur *MobileNet*. Arsitektur *MobileNet* digunakan karena kompleksitasnya rendah dan arsitektur ini sederhana. Pada Tugas Akhir ini pengujian dilakukan menggunakan 5 skenario untuk mendapatkan skenario terbaik. Skenario terbaik yang didapatkan dalam penelitian ini dengan menggunakan citra berukuran 224×224 pixel, optimizer RMSprop, *learning rate* 0.0001, *epoch* 50, dan *batch* 16. Dari skenario terbaik didapatkan hasil performansi terbaik yaitu akurasi data latih 100% dengan *loss* 0,0349, akurasi data uji 100% dengan *loss* 0,0569, dan nilai *recall* 100%, *precision* 100%, dan *f1-score* 100%.

Key words: Kelapa Sawit, *Convolutional Neural Network*, *MOBILENET*.

Pendahuluan

Produsen minyak sawit dunia salah satunya berasal dari Indonesia. Perkebunan kelapa sawit di Indonesia dapat ditemukan di beberapa daerah, khususnya di Aceh, Kalimantan, Sulawesi, Sumatera (terutama daerah Riau, Jambi, Sumatera Selatan), dan Jawa. Iklim tropis dan daerah dengan tingkat curah hujan yang konsisten sangat ideal untuk pertumbuhan tanaman kelapa sawit [1]. Tanaman kelapa sawit banyak tumbuh subur pada daerah yang memiliki tanah jenis gambut seperti pada daerah Pantai Timur Sumatera dan pulau Kalimantan.

Berdasarkan data Kementerian Koordinator Bidang Perekonomian Republik Indonesia pada tahun 2018 dihasilkan 46,68 juta ton minyak sawit terdiri dari *Crude Palm Oil* (CPO) sebanyak 40,57 juta ton, dan *Palm Kernel Oil* (PKO) sebanyak 8,11 juta ton [2]. Penentuan kematangan kelapa sawit secara manual cukup memerlukan waktu, tenaga, dan konsentrasi yang tinggi sehingga diperlukan suatu sistem yang mampu mendeteksi tingkat kematangan kelapa sawit dengan cepat. Metode yang digunakan untuk pendeteksian tingkat kematangan dari buah kelapa sawit ini salah satunya yaitu dengan *Convolutional*

Neural Network (CNN). CNN adalah bagian dari *Deep Neural Network* yang termasuk dalam kelas jaringan saraf tiruan. Penggunaan CNN dipilih karena dapat menghasilkan dan menunjukkan nilai akurasi dalam mendeteksi kematangan kelapa sawit dengan baik, karena dapat berisi seluruh informasi dari skala yang berbeda untuk mendapatkan klasifikasi objek atau gambar yang lebih akurat.

Pada penelitian sebelumnya mengklasifikasikan kematangan pucuk daun teh varietas *Assamica* Klon GMB 7 menggunakan metode CNN dengan membandingkan 2 arsitektur yaitu arsitektur VGGNET19 dan ResNet50 didapatkan hasil nilai akurasi menggunakan arsitektur VGGNET19 sebesar 97,5% [3]. Penelitian lain menentukan tingkat kematangan mentimun menggunakan CNN dengan arsitektur VGG-16. Peneliti mengkategorikan dataset menjadi tiga kategori yaitu matang dan mentah dengan data *test* yang digunakan 200 citra dan data *train* 800 citra memperoleh nilai akurasi sebesar 98,5% [4]. Lalu penelitian klasifikasi terhadap kualitas bibit kelapa sawit untuk meningkatkan penggarapan dari tanaman kelapa sawit dengan metode yang digunakan yaitu CNN menggunakan arsitektur *ResNet50*. Peneliti membagi citra bibit kelapa sawit yang berjumlah 612 citra menjadi 4 kelas

menghasilkan tingkat akurasi sebesar 95% [5]. Untuk menentukan kematangan kelapa sawit, penelitian tugas akhir ini membuat sebuah sistem yang menggunakan pendekatan CNN dengan arsitektur *Mobile-Net*. Penelitian ini mengkategorikan 300 foto menjadi tiga kelas sebagai dataset citra (mentah, matang, dan busuk) dengan pembagian jumlah 195 data latih, 60 data validasi dan 45 data uji. Pendeteksian kematangan kelapa sawit ini akan menggunakan berbagai parameter dan diharapkan hasil terbaik dalam pendeteksian kematangan kelapa sawit.

Tinjauan Pustaka

Kelapa Sawit

Kelapa sawit merupakan tanaman berjenis monokotil sehingga tanaman kelapa sawit memiliki batang tumbuhan yang tidak memiliki cabang dan kambium serta memiliki akar berjenis serabut. Pertumbuhan tanaman kelapa sawit dapat memiliki tinggi maksimal diantara 15 – 18 m dengan pertumbuhan tiap tahunnya sebanyak 25 – 75 cm/tahun dan dapat tumbuh hingga 100 cm jika tumbuh di tanah dengan kondisi yang sesuai. Tanaman kelapa sawit dapat menghasilkan 20-22 tandan per tahun pada kondisi yang produktif sedangkan menghasilkan 12-14 tandan per tahun pada kondisi tanaman yang tua dengan jumlah buah sebanyak 1600 pada setiap tandan [6]. Kelapa sawit merupakan penghasil minyak terefisien dibandingkan dengan penghasil minyak nabati lainnya, karena buah sawit menghasilkan PKO (*Palm Kernel Oil*) dari kernel yang merupakan minyak inti sawit dan CPO (*Crude Palm Oil*) yaitu minyak sawit kasar yang dihasilkan dari daging buah sawit. CPO dihasilkan sebanyak 20-24% dan PKO dihasilkan sebanyak 3-4%. Minyak kelapa sawit dapat dihasilkan sebanyak 20 kg dari setiap 100 kg buah sawit [7].

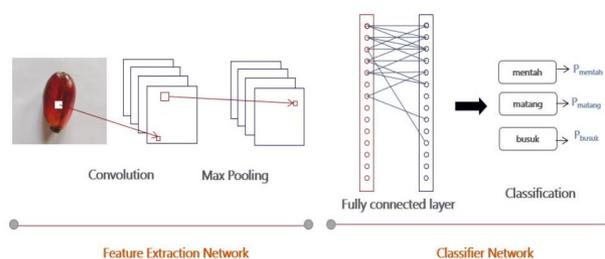
Penentuan kematangan kelapa sawit secara tradisional salah satunya dapat dilihat dari warna buahnya. Antosianin atau pigmen alami pada buah sawit menghasilkan warna merah, ungu, dan biru tergantung dari kondisi pH. Senyawa antosianin seiring bertumbuhnya tingkat kematangan cenderung menurun sehingga pada buah sawit kondisi matang memiliki lebih sedikit kandungan antosianin dari pada sawit kondisi mentah [8]. Buah sawit sangat muda memiliki warna hijau pucat, kemudian warna sawit berubah seiring bertambah tua menjadi warna hijau kehitaman, lalu menjadi warna kuning muda, hingga saat matang warna sawit berubah merah kekuningan atau *orange* dan menjadi warna merah tua saat sawit dalam keadaan terlalu matang.

Citra Digital

Citra digital merupakan citra dari proses rekam menggunakan perangkat digital dan diproses dengan komputer. Komputer dapat langsung memproses citra yang dihasilkan oleh perangkat digital karena sistem kuantisasi dan *sampling* terdapat pada perangkat digital. Sistem kuantisasi adalah proses mengubah intensitas analog menjadi intensitas diskrit untuk menghasilkan gradasi warna yang sesuai dengan kebutuhan sedangkan sistem *sampling* melakukan perubahan pada citra analog menjadi citra diskrit dengan cara membagi M baris dan N kolom pada citra analog [9].

Pengolahan Citra Digital

Pengolahan citra adalah proses penanganan gambar yang masuk dan mengubahnya menjadi gambar yang diinginkan sebagai produk akhir. Salah satu cara untuk meningkatkan kualitas citra agar lebih mudah dipahami baik oleh manusia maupun komputer adalah pengolahan citra digital. Kelebihan dari pengolahan citra digital yaitu mampu mengidentifikasi produk fisik secara objektif. Melalui proses pengolahan citra digital, dapat tercipta citra digital baru dengan kualitas citra yang tinggi [10].



Gambar 1. Arsitektur CNN.

Citra warna (RGB)

Citra warna (RGB) merupakan gambar yang tersusun dari intensitas piksel dengan tiga warna spesifik yang berbeda, yaitu komponen merah (*red*), hijau (*green*), dan biru (*blue*) [9]. Tiga komponen tersebut dikombinasikan untuk menghasilkan intensitas setiap piksel. Citra warna dalam *file* grafis disimpan sebagai citra 24 bit dengan masing-masing 8 bit pada setiap komponen RGB dengan nilai intensitas pada setiap komponen bernilai 0 sampai 255.

Citra Grayscale

Citra *grayscale* merupakan bentuk gambar yang memiliki nuansa abu-abu di dalamnya. Nuansa abu-abu tersebut didapatkan dari campuran warna putih dan hitam yang menghasilkan piksel dengan intensitas keabuan. *Pixel* pada citra *grayscale* membutuhkan masing-masing 8 bit memori [9].

Citra Biner

Citra biner adalah bentuk gambar digital yang hanya memiliki warna hitam dan putih sebagai warna utamanya. Setiap bit piksel gambar biner hanya membutuhkan 1 bit memori dan dapat memiliki 2 intensitas berbeda. Hanya membutuhkan 1 bit memori pada setiap bit pikselnya dan memiliki 2 kemungkinan nilai intensitas. Nilai intensitas dari citra biner secara numerik terdiri dari angka 0 yang menghasilkan warna hitam, dan 1 menghasilkan warna putih [15].

Convolutional Neural Networks (CNN)

Convolutional Neural Network (CNN) 1 adalah salah satu metode *supervised deep learning* karena memiliki kompleksitas yang tinggi dengan struktur yang dirancang mengikuti sistem kinerja otak manusia seperti *Artificial Neural Networks* (ANN). Sejarah CNN diawali oleh Hubel dan Wiesel pada tahun 1959, tahun di mana studi neurologis pertama dilakukan. Eksperimen neurobiologis kemudian dikembangkan oleh peneliti dari NHK *Broadcasting Science Research Laboratories*, Tokyo, Jepang bernama Kunihiko Fukushima dan diberi nama *neocognitron* pada tahun 1979, kemudian menjadi cikal bakal dari perkembangan CNN [11].

Convolutional Layer

Convolutional layer berisi filter (*kernel*) dengan ukuran tinggi dan lebar yang lebih kecil daripada model *input*. *Kernel* kemudian dilakukan pergeseran (*stride*) melintasi tinggi dan lebar input citra kemudian melakukan operasi perkalian "*dot*" antara *input* dan nilai dari *filter* tersebut [12]. *Convolutional layer* diaplikasikan untuk mendapatkan *feature map*.

Stride

Stride merupakan parameter jumlah pergeseran filter yang dilakukan selama proses konvolusi pada matriks citra. Jika *stride* bernilai 1 maka

dilakukan pergeseran 1 piksel pada proses konvolusi. Jika nilai *stride* 2 maka pergeseran dilakukan 2 piksel pada proses konvolusi dan seterusnya. *Stride* dapat dilakukan baik secara vertikal atau *horizontal* [14].

Fungsi Aktivasi *Rectified Linear Units* (ReLU)

Rectified Linear Units (ReLU) adalah salah satu dari fungsi aktivasi, yaitu untuk menghapus nilai negatif pada citra. Aktivasi ReLU bekerja dengan mengganti nilai negatif pada citra dengan nilai 0 [14]. ReLU digunakan untuk meningkatkan *fase* pelatihan jaringan saraf tiruan dan meminimalisir *error* dalam melakukan *feature extraction*. Persamaan (1) fungsi ReLU.

$$f(x) = \max(0, x) \quad (1)$$

Dengan, $f(x)$ adalah nilai dari ReLU, x : nilai matriks

Pooling Layer

Pooling layer mengurangi jumlah parameter dan mempercepat proses komputasi dengan cara memperkecil *feature map* (*downsampling*) [12]. *Pooling Layer* memiliki keuntungan yaitu secara bertahap mengurangi *feature map* dan ukuran *volume* keluaran, yang dapat memungkinkan pengendalian *overfitting*. *Pooling* berfungsi untuk memperkecil ukuran data, sehingga jumlah parameter untuk dipelajari dapat berkurang. Metode *pooling* terdapat 2 jenis yaitu menggunakan *average pooling* dan *max pooling*.

Fully Connected Layer

Feature map hasil dari *feature extraction* dilakukan "*flatten*" untuk menjadi masukan pada *fully connected layer*. Saat proses *flatten*, nilai *layer* sebelumnya diubah menjadi urutan numerik (vektor) atau dimensi untuk mendukung proses program klasifikasi. Setelah fitur diekstraksi oleh lapisan konvolusi dan di-*downsampling* oleh *pooling layer*, fitur tersebut dipetakan oleh *fully connected layer* yang menghasilkan *output* jaringan seperti probabilitas untuk setiap kelas untuk tugas melakukan klasifikasi melalui fungsi aktivasi yaitu aktivasi *softmax* [13]. Lapisan terakhir *fully connected layer* memiliki jumlah simpul *output* yang sama dengan jumlah kelas.

Fungsi Aktivasi *Softmax*

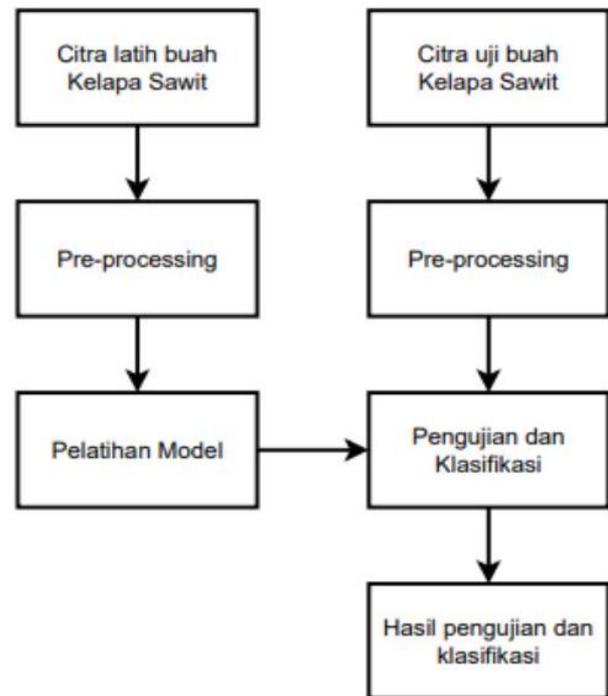
Fungsi aktivasi *softmax* terletak pada akhir *fully connected layer*. *Softmax* bekerja sebagai *layer* aktivasi dan digunakan untuk klasifikasi kelas. *Softmax* menghasilkan nilai probabilitas untuk setiap kelas [14]. Ketika sudah menerapkan *softmax* komponen akan bernilai (0-1) dan akan bertambah 1. *Softmax* digunakan untuk menentukan *output* yang sesuai. Persamaan (2) fungsi aktivasi *softmax*

$$\text{Softmax}(x)_i = \frac{\exp(x_i)}{\sum_{j=1}^m \exp(x_j)} \quad (2)$$

MobileNet

Arsitektur dari CNN yang digunakan untuk mengatasi masalah dari *computing resource* adalah *MobileNet* yang diciptakan oleh peneliti dari *Google* untuk memenuhi kebutuhan dari *mobile*. *MobileNet* memiliki ketepatan yang hampir sama dengan model CNN umum, namun *MobileNet* memberikan efisiensi yang tinggi, *stable identification*, dan dapat memenuhi masalah biaya karena bersifat *Low-cost*. *MobileNet* memiliki konvolusi yang terbagi menjadi 2 (dua), *depthwise convolution* dan *pointwise convolution*.

Lapisan konvolusi standar pada umumnya hanya menggunakan lapisan konvolusi sebesar 3×3 . Namun, pada lapisan *depthwise separable convolution* dipisahkan menjadi dua konvolusi, yaitu 1×1 *pointwise convolution* dan 3×3 *depthwise convolution* serta *batchnorm* dan ReLU pada setiap lapisan konvolusi.



Gambar 2. Blok Diagram Sistem.

Metodologi Penelitian

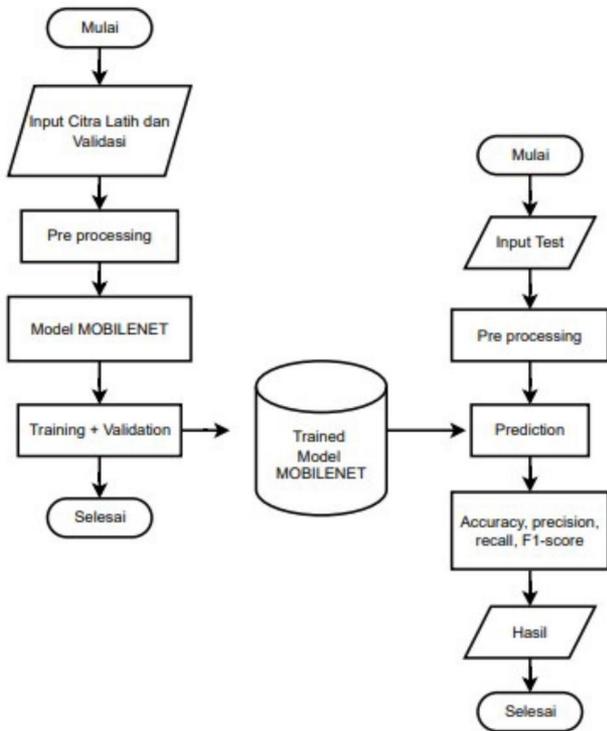
Desain Sistem

Sistem *Deep Learning* diimplementasikan dalam penelitian tugas akhir ini untuk mengklasifikasikan tingkat kematangan dari buah kelapa sawit menggunakan metode *Convolutional Neural Network* (CNN) dengan arsitektur model *MOBILENET*.

Pada gambar 2 menunjukkan tahapan dari rancangan sistem. Pertama, melakukan *input* citra menjadi dataset citra buah kelapa sawit berupa citra RGB dengan format .JPG. Kemudian, citra akan dilakukan *preprocessing* melalui proses *resize* agar citra yang di-*input* lebih mudah dipahami oleh sistem. Selanjutnya melakukan pelatihan model menggunakan CNN untuk disimpan dan kemudian dilakukan pengujian. Keluaran sistem akan dilakukan pengukuran akurasi dengan parameter *accuracy*, *precision*, *recall*, *F1-score*.

Model Penelitian

Tahap pertama pada diagram alir 3 yaitu melakukan *input* citra dataset. Selanjutnya citra yang telah di-*input* masuk ke tahap *preprocessing* yaitu melakukan *resize* citra dengan mengubah ukuran piksel citra sebelum masuk ke proses CNN. Setelah proses *preprocessing*, masuk ke model pelatihan CNN yang memiliki dua *fase* yaitu *training* dan *testing* dengan arsitektur *MOBILENET*. Proses pada tahap *training* dilakukan untuk melatih model yang akan digunakan untuk tahap *testing*. Data *training* memiliki hasil yang akan mengalami proses performansi sistem arsitektur *MOBILENET* dan disimpan pada *database*. Sebelum masuk ke *database* data *training* tersebut dilakukan validasi menggunakan data *validation*. Setelah itu memasukkan data yang tersimpan pada data citra uji yang telah melalui proses yang sama dengan data *training*. Terakhir sistem akan memberikan hasil sesuai dengan model yang dihasilkan.



Gambar 3. Diagram Alir Penelitian.



Gambar 4. Diagram Alur Preprocessing.

Dataset

Dataset yang digunakan pada Tugas Akhir ini merupakan dataset primer berupa citra RGB yang diambil menggunakan kamera smart-phone. Dataset yang diperoleh berupa data citra kelapa sawit dengan format .JPG yang terdiri kelas yaitu mentah, matang, dan busuk. Jumlah dataset yang digunakan adalah sebanyak 300 data yang terbagi menjadi 100 dataset mentah, 100 dataset matang, dan 100 dataset busuk. Pada penelitian ini dilakukan pembagian dengan rasio 65% data latih, 20% data validasi, dan 15% data uji.

Pre-processing

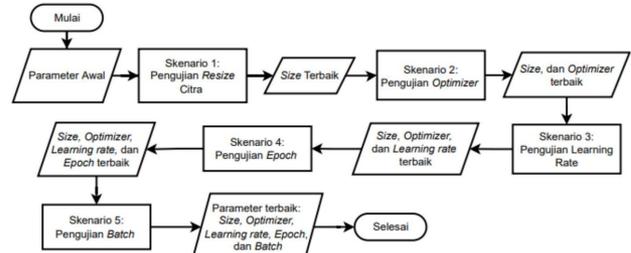
Preprocessing memiliki tujuan agar citra yang di-input layak untuk masuk ke tahap selanjutnya dan mudah dipahami oleh sistem. Pada Gambar 4 terlihat alur proses dari preprocessing. Proses preprocessing dilakukan dengan cara menghilangkan bagian dari citra input yang tidak dibutuhkan pada sistem. Pada sistem ini menggunakan proses resize.

Performansi Sistem

Performansi sistem digunakan untuk mengukur kinerja klasifikasi model pelatihan CNN.

Confusion Matrix

Confusion Matrix berfungsi sebagai alat pengukuran dan perhitungan kinerja model atau seberapa benar pengklasifikasian yang telah dibuat.



Gambar 5. Diagram Alir Skenario Pengujian.

Pada confusion matrix memiliki baris dan kolom, baris sebagai kelas prediksi dan kolom sebagai kelas sebenarnya. Confusion matrix digunakan untuk menghitung akurasi, presisi, loss, recalls, dan fl score yang berfungsi sebagai parameter untuk menentukan performansi dan kinerja dari sistem yang telah dikerjakan.

Akurasi

Akurasi adalah metrik uji seberapa akurat model yang digunakan dapat mengklasifikasikan dengan benar. Nilai akurasi didapat dari perbandingan data prediksi benar dengan keseluruhan data.

Presisi

Presisi adalah indikator uji yang menggambarkan seberapa akurat model memprediksi peristiwa positif melalui berbagai operasi prediktif. Nilai presisi didapat dari perbandingan data True Positive (TP) dengan keseluruhan data yang bernilai positive.

Loss

Loss adalah indikator uji yang menjelaskan perhitungan yang menyebabkan ketidakakuratan sistem dalam pengenalan objek.

Recall

Recall adalah indikator uji yang menentukan keberhasilan jenis citra yang sudah diidentifikasi. Nilai recall didapat dari membandingkan nilai True Positive (TP) dengan banyaknya data yang sebenarnya positive.

F1 Score

F1 Score merupakan perhitungan evaluasi dalam informasi pencarian menggabungkan hasil recall dan presisi. Nilai F1 Score merupakan nilai mean dari nilai presisi dan recall.

Hasil dan Pembahasan

Dengan parameter awal rasio dataset 65:20:15, learning rate 0.0001, batch 16, epoch 50, optimizer adam 5.

Skenario Pengujian

Pada tabel 1, Size citra 224×224 menghasilkan nilai akurasi tertinggi dengan akurasi training sebesar 99,49%, akurasi testing sebesar 97,78%. Nilai precision, recall dan f1-score yang didapatkan menunjukkan nilai terbaik dengan hasil pada masing-masing parameter evaluasi citra memiliki nilai average sebesar 98%.

Skenario Kedua (Optimizer)

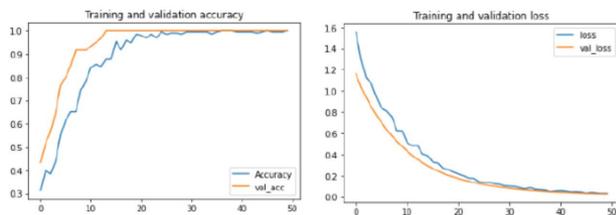
Pada tabel 2, Optimizer RMSprop menghasilkan nilai akurasi tertinggi dengan akurasi training dan akurasi testing sebesar 100%. Nilai precision, recall dan fl-score yang didapatkan menunjukkan nilai terbaik

Table 1. Hasil Pengujian *Resize*

Size Citra	Train			Test	Performance		
	Training(%)	Validation(%)	Loss	Testing(%)	Avg Precision(%)	Avg Recall(%)	Avg F1-Score(%)
128 × 128	98,97	98,33	0,0909	97,78	98	98	98
160 × 160	97,95	100	0,1283	97,78	98	98	98
192 × 192	98,46	98,33	0,0987	95,56	96	96	96
224 × 224	99,49	100	0,1032	97,78	98	98	98

Table 2. Hasil Pengujian *Optimizer*

Optimizer	Train			Test	Performance		
	Training(%)	Validation(%)	Loss	Testing(%)	Avg Precision(%)	Avg Recall(%)	Avg F1-Score(%)
Adam	99,49	100	0,1032	97,78	98	98	98
RMSprop	100	100	0,0349	100	100	100	100
Nadam	99,49	100	0,0983	100	100	100	100

**Gambar 6.** Grafik Hasil *Training* Pengujian Skenario Terbaik.

dengan hasil pada masing-masing parameter evaluasi citra memiliki nilai *average* sebesar 100%.

Skenario Ketiga (*Learning Rate*)

Pada tabel 3, LR 0,01 dan LR 0,0001 menghasilkan nilai akurasi tertinggi dengan akurasi *training* dan akurasi *testing* sebesar 100%. Nilai *precision*, *recall* dan *f1-score* yang didapatkan juga menunjukkan nilai terbaik dengan hasil pada masing-masing parameter evaluasi citra memiliki nilai *average* sebesar 100%. Namun LR 0,01 menghasilkan grafik yang tidak stabil sehingga parameter LR 0,0001 dianggap sebagai nilai *learning rate* terbaik.

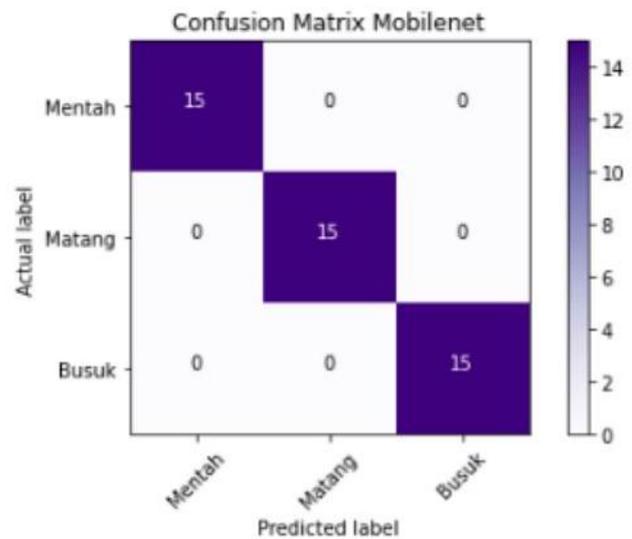
Skenario Keempat (*Epoch*)

Pada tabel 4, *Epoch* 50 menghasilkan nilai akurasi tertinggi dengan akurasi *training* dan akurasi *testing* sebesar 100%. Nilai *precision*, *recall* dan *f1-score* yang didapatkan menunjukkan nilai terbaik dengan hasil pada masing-masing parameter evaluasi citra memiliki nilai *average* sebesar 100%.

Skenario Kelima (*Batch*)

Parameter terbaik mendapatkan hasil pengujian yang optimal dengan nilai akurasi *training* sebesar 100% dan akurasi *testing* sebesar 100% serta nilai *loss training* 0,0349 dan *loss testing* 0,0569. Parameter terbaik ini juga menghasilkan kinerja yang baik dilihat dari grafik pengujian pada Gambar 6 dan *confusion matrix* pada Gambar 7.

Grafik tidak mengalami *overfitting* maupun *underfitting* karena selisih antara nilai akurasi dan validasi akurasi serta nilai *loss* dan validasi *loss* yang kecil. Berdasarkan hasil *confusion matrix* menunjukkan

**Gambar 7.** Confusion Matrix Skenario Terbaik.

bahwa pengujian sistem menggunakan 45 data uji ini berhasil melakukan klasifikasi pada kelas mentah, matang, dan busuk tanpa terjadi kesalahan.

Kesimpulan

Sistem yang dirancang berhasil melakukan klasifikasi dari tingkat kematangan buah kelapa sawit menggunakan CNN dengan arsitektur *MOBILENET* menjadi 3 kelas, yaitu mentah, matang, dan busuk dengan nilai akurasi *training* dan akurasi *testing* 100%. Akurasi tertinggi dan performansi yang optimal diperoleh saat sistem menggunakan kombinasi parameter sebagai berikut, rasio dataset dengan perbandingan 65:20:15, *resize* citra berukuran 224 × 224, *optimizer* RMSprop, *learning rate* 0,0001; *epoch* 50, dan *batch* 16. Nilai akurasi *training* dihasilkan sebesar 100% dan akurasi *testing* sebesar 100% serta nilai *loss training* 0,0349 dan *loss testing* 0,0569. Performansi dari sistem

Table 3. Hasil Pengujian *Learning Rate*

LR	Train			Test		Performance	
	Training(%)	Val(%)	Loss	Testing(%)	Avg Precision(%)	Avg Recall(%)	Avg F1-Score(%)
0,01	100	100	1,0932e - 05	100	100	100	100
0,001	99,49	100	0,0909	100	100	100	100
0,0001	100	100	0,0349	100	100	100	100
0,00001	68,21	85	0,7344	80	85	80	78

Table 4. Hasil Pengujian *Epoch*

Epoch	Train			Test		Performance	
	Training(%)	Validation(%)	Loss	Testing(%)	Avg Precision(%)	Avg Recall(%)	Avg F1-Score(%)
10	73,85	91,67	0,6483	93,33	93	93	93
20	94,87	100	0,2331	97,78	98	98	98
30	98,97	100	0,1111	100	100	100	100
40	99,49	100	0,0592	100	100	100	100
50	100	100	0,0349	100	100	100	100

Table 5. Hasil Pengujian *Batch*

Rasio Data Latih : Data Validasi : Data Uji	65 : 20 : 15
Size Citra	224 × 224
Optimizer	RMSprop
Learning Rate	50
Epoch	16
Batch	0001

mendapatkan nilai yang baik dari parameter *Precision*, *Recall*, dan *F1-Score* dengan nilai *average* 100%. Sistem rancangan yang telah diuji masuk dalam kategori *good fitting* tidak *underfitting* atau *overfitting*. Berdasarkan grafik pengujian terlihat selisih antara data *training* dan data validasi *training* kecil dan cenderung berhimpitan. Pada pengujian dilihat dari *confusion matrix* tidak terdapat kesalahan pada proses klasifikasi dari data uji.

Daftar Pustaka

- Yuliani A, Labellapansa A, Yulianti A. Klasifikasi Citra Daun Kelapa Sawit Yang Terkena Dampak Hama Menggunakan Metode K-Nearest Neighbor. In: Seminar Nasional Informatika Medis (SNIMed); 2019. p. 73-8.
- Industri Kelapa Sawit Indonesia: Menjaga Keseimbangan Aspek Sosial, Ekonomi, dan Lingkungan;. Available from: <https://www.ekon.go.id/publikasi/detail/2921/industri-kelapa-sawit-indonesia>.
- Ibrahim N, et al. Klasifikasi Tingkat Kematangan Pucuk Daun Teh menggunakan Metode Convolutional Neural Network. Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika. 2022;10(1):162.
- Agustini SLD. Implementasi CNN dalam mendeteksi kematangan mentimun melalui rekognisi citra digital. UIN Sunan Gunung Djati Bandung; 2022.
- Oktafanda E. Klasifikasi Citra Kualitas Bibit dalam Meningkatkan Produksi Kelapa Sawit Menggunakan Metode Convolutional Neural Network (CNN). Jurnal Informatika Ekonomi Bisnis. 2022;4(3):72-7.
- Fauzi IS Y E Widyastuti, Paeru RH. Kelapa sawit; 2012.
- Masykur M. Pengembangan Industri Kelapa Sawit Sebagai Penghasil Energi Bahan Bakar Alternatif Dan Mengurangi Pemanasan Global. Reformasi. 2013;3(2):319-428.
- Fitrya N, Wirman SP, Fitri W. Identifikasi Karakteristik Buah Kelapa Sawit Siap Panen Dengan Metode Laser Spekel Imaging (Lsi). Photon: Jurnal Sains dan Kesehatan. 2018;9(1):139-42.
- Andono PN, Sutojo T. Pengolahan Citra Digital. 2017:1-2.
- Effendi M, Fitriyah F, Effendi U. Identifikasi Jenis dan Mutu Teh Menggunakan Pengolahan Citra Digital dengan Metode Jaringan Syaraf Tiruan. Teknotan: Jurnal Industri Teknologi Pertanian. 2017;11(2):67.
- Rawat W, Wang Z. Deep convolutional neural networks for image classification: A comprehensive review. Neural Computation. 2017;29(9):2352-449.
- Ke Q, Liu J, Bennamoun M, et al. Computer vision for human-machine interaction. In: Computer Vision for Assistive Healthcare. Elsevier; 2018. p. 127-45.
- Wang W, Yang Y. Development of convolutional neural network and its application in image classification: a survey. Optical Engineering. 2019;58(4):040901.
- Setiawan W, Utoyo MI, Rulaningtyas R. Classification of neo-vascularization using convolutional neural network model. Telekomnika (Telecommunication Computing Electronics and Control). 2019;17(1):463-72.
- Vlerick A. A visual introduction to binary image processing (part 1) — by alejandra vlerick — towards data science; 2020. Available from: <https://towardsdatascience.com/a-visual-introduction-to-binary-image-processing-part-1-d2fba9f102a4>.