

## **Analisis Performa Berbagai Arsitektur *Convolutional Neural Network* (CNN) terhadap Ketepatan Deteksi Cacat pada Kemasan *Snack Box***

**Richo**

Teknik Otomasi, Fakultas Teknik Kelistrikan Kapal, Politeknik Perkapalan Negeri Surabaya,  
Jl. Teknik Kimia Kampus ITS Sukolilo, Surabaya, 60222, Indonesia  
[richo@student.ppns.ac.id](mailto:richo@student.ppns.ac.id)

### **Abstrak**

*Kemasan produk bertujuan untuk menjaga integritas dan kualitas produk, memperpanjang masa simpan produk, memberikan indikasi bahwa produk masih dalam kondisi baik, serta memberikan informasi yang diperlukan. Kerusakan pada kemasan memiliki dampak serius terhadap kualitas produk yang terdapat di dalamnya. Penelitian ini mengaplikasikan metode Convolutional Neural Network (CNN) untuk klasifikasi dan pengenalan kemasan snack box, dengan 2 kategori output yakni “defect” atau “perfect”. Dalam penelitian ini, dilakukan perancangan serta perbandingan berbagai arsitektur CNN yang meliputi Inception-V4, VGGNet-19, EfficientNet B7, ResNet-101, dan NASNet. Penelitian ini bertujuan untuk mengidentifikasi arsitektur yang paling optimal dalam mengenali jenis cacat produk, dan pada akhirnya dapat menghasilkan model terbaik yang mampu mendeteksi cacat kemasan dengan tingkat akurasi yang optimal. Berdasarkan hasil parameter training dan confusion matriks, arsitektur paling optimal dalam pembelajaran model yakni arsitektur VGGNet-19, EfficientNet-B7, ResNet-101, NASNet dengan akurasi pada setiap parameter mencapai 100%. Namun jika ditinjau berdasarkan testing, model CNN yang paling optimal dalam klasifikasi produk snack box adalah model arsitektur EfficientNet-B7. Dari total 60 data uji, model EfficientNet-B7 mampu mendeteksi dengan benar 59 data, hanya mengalami 1 kesalahan prediksi. Arsitektur EfficientNet-B7 menghasilkan akurasi yang lebih tinggi dibandingkan dengan arsitektur lain, dengan tingkat akurasi rata-rata mencapai 98,34%.*

**Kata kunci:** Arsitektur, CNN, Snack Box

### **1. Pendahuluan**

Kemasan suatu produk merupakan suatu aspek penting yang umumnya berfungsi untuk melindungi produk dari kerusakan dan kontaminasi (Kusumawati, Kusumah and Wibisono T, 2022). Menurut Kementerian Perdagangan Republik Indonesia, kemasan produk ditujukan untuk menjaga integritas dan kualitas produk, memperpanjang masa simpan produk, memberikan indikasi bahwa produk masih dalam kondisi baik, serta memberikan informasi yang diperlukan. Selain itu, kemasan juga berfungsi untuk mengidentifikasi asal-usul produk dan potensi risiko, serta menyajikan informasi rinci pada label seperti instruksi penggunaan, informasi gizi, dan aspek keamanan lainnya (Kementerian Perdagangan Republik Indonesia, 2017). Kemasan yang menarik sejatinya dirancang untuk menarik perhatian konsumen serta membangun kesan profesionalisme, kualitas suatu produk, serta sebagai indikator dalam membentuk hubungan kepercayaan antara produsen dan konsumen (Ardhianto, Peter and Kurniawan, 2016). Saat ini, salah satu jenis kemasan yang tengah menjadi tren di pasar adalah kemasan kotak berbahan dasar kertas. Kemasan ini populer karena memiliki konstruksi yang ringan, efisien dalam memberikan dukungan visual terkait keterangan penggunaan produk, serta melindungi produk.

Kerusakan pada kemasan memiliki dampak serius terhadap kualitas produk yang terdapat di dalamnya dan juga berpotensi turunnya daya tarik konsumen untuk melakukan pembelian produk (Chen dkk, 2021). Hal tersebut menimbulkan dampak penurunan penjualan dan risiko terhadap citra perusahaan yang semakin memburuk. Pengamatan inspeksi dan pemilahan produk saat ini masih dilakukan secara manual, yang rentan terhadap kemungkinan kemasan cacat terlewatkan dalam

pengamatan manusia (Prastiwinarti dkk, 2023). Akibatnya, kemasan cacat tersebut kemungkinan akan ikut terkirimkan kepada konsumen.

Dalam ilmu pengolahan citra, sejatinya kecacatan kemasan *snack box* dapat dideteksi dengan basis *image processing* untuk mendeteksi dan mengklasifikasikan berdasarkan unsur kecacatan kemasan seperti lekukan, cekungan, dan kerusakan struktur kemasan (Kusumawardani and Karningsih, 2021). Unsur kerusakan tersebut umumnya diakibatkan karena proses pengiriman dan penanganan yang kurang hati-hati sehingga menyebabkan produk mengalami deformasi pada permukaannya. Ilmu pengolahan citra yang sangat ramai yakni metode *Convolutional Neural Network* (CNN) yang merupakan suatu strategi dalam pengolahan informasi yang berfokus pada penggunaan jaringan saraf tiruan (Shidiq, Suhartono and Saidah, 2022). Pendekatan ini secara khusus dikembangkan untuk mengidentifikasi dan menganalisis pola dalam data visual seperti citra gambar. Penerapan metode CNN pada penelitian ini bertujuan untuk mengklasifikasikan hasil pengenalan citra *snack box* yang tergolong atas 2 kategori yakni “*defect*” dan “*perfect*” berdasarkan unsur kerusakan yang dideteksi.

Penelitian terkait pengembangan metode CNN telah beberapa kali diteliti, diantaranya yang dilakukan oleh Alan, dkk yang mengimplementasikan metode CNN untuk mengklasifikasikan citra kemasan kardus dengan 2 *output* yakni *defect* dan *no defect*, total *dataset* sebanyak 1300 data. Penelitian tersebut melakukan eksperimen terhadap beberapa *hyperparameter* serta penambahan ekstraksi citra dengan *Local Binary Pattern* (LBP) dan diperoleh akurasi keberhasilan mencapai 95,77% yang hanya berlangsung pada proses *training* (Antoni, Rohana and Pratama, 2023). Penelitian sejenis yang dilakukan oleh Richo, dkk mengimplementasikan metode CNN dan membandingkan berbagai optimizer untuk klasifikasi pengenalan produk cacat kemasan berdasarkan adanya bercak tepung pada area kemasan, didapatkan akurasi keberhasilan mencapai 90% (Richo dkk, 2023). Penelitian selanjutnya yang dilakukan oleh Oky, menerapkan metode CNN sebagai *feature learning* serta klasifikasi isyarat tangan pada obyek. Penelitian tersebut menggunakan 29 citra isyarat, 26 diantaranya yakni huruf alphabet dan 3 lainnya merupakan isyarat tambahan, didapatkan hasil akurasi *training* model mencapai 97,2% (Nurhayati, Eridani and Tsalavin, 2022). Penelitian yang dilakukan oleh Atam, dkk mengidentifikasi performa arsitektur CNN untuk klasifikasi kesehatan daun tomat, arsitektur yang dibandingkan dalam penelitian tersebut yakni VGG, Inception, dan MobileNet. Didapatkan hasil bahwa arsitektur Inception memiliki akurasi dan efisiensi pembelajaran model yang lebih optimal dibandingkan MobileNet dan VGG dalam klasifikasi kesehatan daun tomat dengan akurasi 98% (Ronggo, Pratomo and Palupingsih, 2023). Penelitian selanjutnya dilakukan oleh Bima, dkk menambahkan arsitektur Efficient-Net B4 pada model CNN untuk optimalisasi pengenalan bahasa isyarat dari gestur tangan, penambahan arsitektur Efficient-Net B4 berhasil meningkatkan akurasi pembelajaran model dengan akurasi keberhasilan sebesar 99,81% (Wardana, Rachmawati and Wirayuda, 2021).

Berdasarkan beberapa kajian literatur yang telah dikumpulkan oleh peneliti, penelitian ini mengaplikasikan metode *Convolutional Neural Network* (CNN) untuk klasifikasi dan pengenalan *defect* pada kemasan *snack box*, dengan 2 kategori *output* pengenalan yakni *defect* atau *perfect*. Dalam kerangka penelitian ini, dilakukan perancangan serta perbandingan berbagai arsitektur CNN yang meliputi Inception-V4, VGGNet-19, EfficientNet B7, ResNet-101, dan NASNet. Penelitian ini bertujuan untuk mengidentifikasi arsitektur yang paling optimal dalam mengenali jenis cacat produk pada kemasan *snack box*, dan pada akhirnya dapat menghasilkan model terbaik yang mampu mendeteksi cacat kemasan dengan tingkat akurasi yang optimal. Tujuan penelitian ini adalah untuk mengidentifikasi arsitektur yang paling optimal dalam mengenali jenis cacat pada kemasan *snack box*. Selain itu, penelitian ini bertujuan untuk menghasilkan model terbaik yang mampu mendeteksi cacat kemasan dengan tingkat akurasi yang optimal. Pada pemodelan penelitian, kategori produk *defect* ditentukan berdasarkan adanya unsur kecacatan seperti lekukan, cekungan, dan kerusakan struktur kemasan pada area depan dan belakang kemasan.

## 2. Metode Penelitian

Bagian tahapan metode penelitian merincikan bagaimana penelitian dilaksanakan. Penjelasan yang terperinci mengenai langkah-langkah penelitian dan metodenya disajikan untuk mempermudah pemahaman pembaca terhadap proses penelitian yang dilakukan.

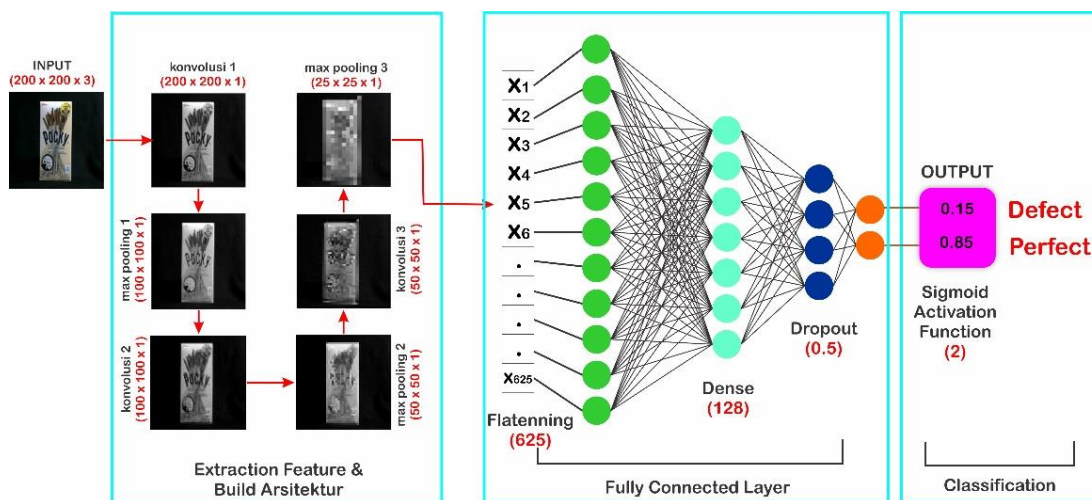
### 2.1 Produk Penelitian

Produk penelitian yang digunakan yakni kemasan *snack box* dengan brand “Pocky” varian vanila susu dengan berat bersih 40 gram. Data gambar diambil dengan menggunakan *webcam* sejumlah 200 data yang meliputi data produk *defect* dan *perfect*. Data yang dikumpulkan terdiri atas *capture* bagian depan dan belakang produk. Tabel 1 merupakan contoh beberapa *dataset* produk yang dikumpulkan.

**Tabel 1.** Tampilan *Dataset* Produk

Perfect Depan	Perfect Belakang	Defect Depan	Defect Belakang
			

### 2.2 Convolutional Neural Network (CNN)



Gambar 1. Ilustrasi Pemrosesan Citra dengan CNN

*Convolutional Neural Network* (CNN) merupakan gabungan dari proses konvolusi citra untuk pemrosesan ekstraksi citra data, dan jaringan neuron untuk klasifikasi. Sejatinnya, sebuah arsitektur CNN secara umum terdiri dari empat jenis lapisan utama, yaitu lapisan konvolusi, ReLU *activation*, lapisan *subsampling*, dan lapisan *fully connecting*. Fungsi dari masing-masing lapisan ini adalah untuk melakukan transformasi data secara bertahap dalam rangka ekstraksi fitur dan klasifikasi data.

1. Lapisan Konvolusi: Lapisan ini bertanggung jawab atas proses konvolusi pada keluaran dari lapisan sebelumnya. Ini merupakan tahap inti yang membentuk dasar operasi dalam CNN. Konvolusi mengacu pada penggunaan fungsi matematis untuk memproses *output* fungsi lainnya.
2. Lapisan ReLU (*Rectified Linear Unit*): Lapisan ini dapat dianggap sebagai suatu bentuk *thresholding* atau fungsi *activation* dalam implementasi jaringan saraf tiruan. Lapisan ini membantu dalam menambahkan unsur non-linearitas dalam jaringan.

3. Lapisan *Subsampling*: *Subsampling* adalah langkah pengurangan ukuran data gambar. Dalam konteks pemrosesan citra, lapisan *subsampling* juga berperan untuk meningkatkan invariansi terhadap posisi suatu fitur yang diekstraksi.
4. Lapisan *Fully Connected*: Lapisan ini sering kali diterapkan dalam model MLP (*Multi-layer Perceptron*) bertujuan untuk mentransformasikan dimensi suatu data sehingga kumpulan data dapat dianalisis dan diklasifikasikan dengan linear atau non-linear.

Ilustrasi pemrosesan citra menggunakan metode CNN dapat dilihat pada Gambar 1.

### 2.3 Arsitektur *Convolutional Neural Network* (CNN)

Arsitektur model CNN merupakan desain struktural yang memberikan pedoman tentang bagaimana lapisan-lapisan akan diatur, jumlah *neuron* pada masing-masing lapisan, dan cara interaksi antara lapisan-lapisan tersebut. Pemilihan arsitektur yang sesuai sangat penting dalam memastikan bahwa model CNN dapat mengekstraksi fitur-fitur yang relevan dan menghasilkan hasil prediksi yang akurat sesuai dengan tujuan yang diinginkan. Berikut merupakan penjelasan beberapa arsitektur yang digunakan pada penelitian ini.

#### a. Arsitektur Inception-V4

Arsitektur Inception-V4, juga dikenal sebagai Inception *version 4*, adalah pengembangan dari arsitektur *Inception (GoogLeNet)* yang dirancang untuk meningkatkan kinerja dalam tugas pengenalan gambar. Inception-V4 memiliki tujuan utama untuk mencapai keseimbangan antara kedalaman (*depth*) dan lebar (*width*) sehingga dapat menghasilkan model yang lebih efisien dan akurat. Inception-V4 menerapkan teknik seperti *factorization* dan *separable convolution* untuk mengurangi kegagalan gradien selama proses pelatihan. Hal ini membantu mempercepat konvergensi dan membuat jaringan lebih mudah dilatih (Wahid, Mustamin and Lawi, 2021).

#### b. Arsitektur VGGNet-19

Arsitektur VGGNet-19 adalah sebuah model arsitektur *Convolutional Neural Network* (CNN) yang dikembangkan oleh tim *Visual Geometry Group* (VGG) di University of Oxford. Arsitektur ini dikenal karena kedalaman (*depth*) yang relatif besar, dengan total 19 lapisan yang terdiri dari *layer* konvolusi, lapisan aktivasi, dan konstruksi *layer fully connected*. Arsitektur VGG-19 memiliki keunggulan dalam ekstraksi fitur yang lebih baik, tetapi juga memiliki komputasi yang lebih berat dibandingkan dengan arsitektur yang lebih ringan. Meskipun demikian, VGG-19 telah menjadi salah satu arsitektur yang sangat berpengaruh dalam pengembangan CNN dan telah memberikan kontribusi signifikan dalam bidang pengenalan gambar (Rizki and Marina, 2019).

#### c. Arsitektur EfficientNet-B7

Arsitektur EfficientNet-B7 merupakan bagian dari model *Convolutional Neural Network* (CNN) yang dikenal sebagai EfficientNet. Arsitektur ini memiliki keunikan dalam mengoptimalkan ukuran, kedalaman, dan resolusi jaringan secara bersamaan untuk mencapai performa yang optimal dalam memproses gambar dengan sumber daya komputasi yang terbatas. Arsitektur EfficientNet menggunakan faktor skalabilitas (*scaling coefficient*) untuk mengontrol ukuran dan kedalaman model. Pada EfficientNet-B7, faktor skalabilitas adalah 7, yang menghasilkan model dengan dimensional citra yang lebih besar dan kompleks (Perdani, Magdalena and Caecar, 2022).

#### d. Arsitektur ResNet-101

ResNet-101 merupakan arsitektur yang menggunakan "*bottleneck blocks*" pada beberapa *layer* untuk mengurangi kompleksitas komputasi. *Bottleneck blocks* terdiri dari tiga lapisan konvolusi, konvolusi 1x1 untuk mengurangi dimensi fitur, konvolusi 3x3 untuk mengekstraksi fitur, dan konvolusi 1x1 lagi untuk memulihkan dimensi awal. Kedalaman Jaringan ResNet-101 memiliki total 101 lapisan, termasuk *layer* konvolusi, *batch normalization*, *activation*, dan *pooling*. Kedalaman ini memungkinkan model untuk mengambil representasi hierarkis yang lebih kompleks dari data gambar (Satria and Pardede, 2022).

**e. Arsitektur NASNet**

Arsitektur NASNet merupakan satu-satunya arsitektur yang memiliki struktur *multi-tiered*, yang berarti memiliki beberapa level atau lapisan yang saling terhubung. Setiap tier terdiri dari sejumlah blok yang dapat diatur, dan algoritma pencarian akan menentukan bagaimana menggabungkan dan mengatur blok-blok tersebut (Hardi, 2022).

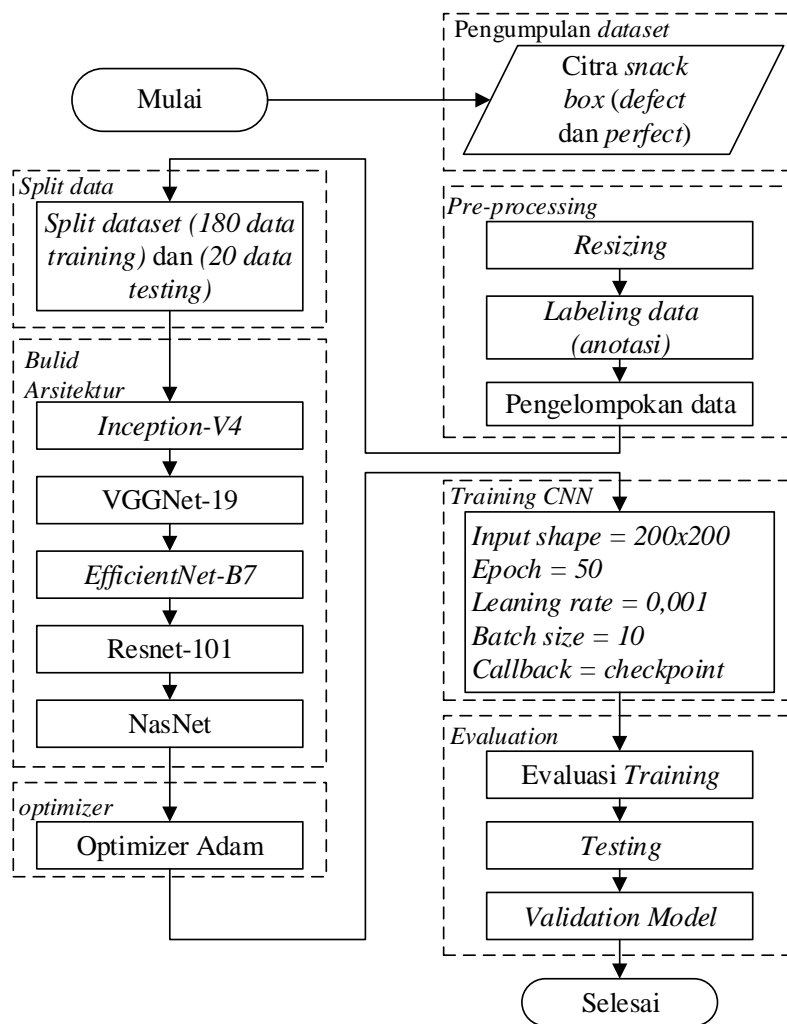
Berdasarkan beberapa kajian literatur yang telah dilakukan peneliti, penelitian ini menetapkan *layer* konstruksi dan penambahan elemen khas pada setiap arsitektur yang disajikan pada Tabel 2 berikut.

**Tabel 2.** Perancangan Arsitektur Model CNN

Arsitektur	Parameter Perancangan Arsitektur Oleh Peneliti					
	Konvolusi	Sub Sampling	Concatenate	Dense	Dropout	Layer Unggulan
Inception-V4	9 layer	3 max pooling dan 1 average pooling	3 tower	3 lapis	3 lapis	skema kontrol gradien, label smoothing
VGGNet-19	16 layer	5 max pooling	-	3 lapis, 4096 neuron	-	Include heatmaps dan grad-CAM
EfficientNet-B7	12 layer	5 max pooling	12 layer batch normalizati	2 layer, 2048 neuron	0,5 hidden	Mobile Inverted Residual Bottleneck (MBConv), exploding gradient
ResNet-101	4 layer	-	4 layer batch normalizati	1 layer, 1000 neuron	Identity block, global average	shortcut connections
NASNet	3 layer konvolusi, 9 layer normal cell	2 layer reduction cell	5 layer sepConv	-	-	reduction cell dan sepConv

**2.4 Alur Penelitian**

Alur penelitian digunakan untuk membahas tahapan penelitian yang akan dijalankan, Gambar 2 berikut merupakan *flowchart* alur penelitian. Berdasarkan Gambar 2, tahapan alur penelitian dimulai dengan mengumpulkan *dataset* produk yang terdiri atas produk *defect* dan *perfect*. Proses selanjutnya dilakukan tahapan *preprocessing* yang meliputi *resizing* untuk menyamakan ukuran *dataset*, dan *labeling* data untuk memberikan identitas pada setiap data, dan pengelompokan data berdasarkan kategori. Langkah selanjutnya yakni *split data* dengan *persentase* 90% *training data* dan 10% merupakan *testing data*. Tahapan yang tak kalah penting yakni membangun setiap arsitektur pada setiap pola model, dalam penelitian ini *build* arsitektur terbagi atas 5 model dengan berbagai variasi arsitektur yakni Inception-V4, VGGNet-19, EfficientNet-B7, ResNet-101, dan NasNet. Optimizer yang digunakan yakni optimizer Adam untuk meminimalisir terjadinya *overfitting* selama proses pembelajaran. Selanjutnya dilakukan proses *training* menggunakan *google colab* untuk mendapatkan konstruksi model yang optimal. Proses evaluasi *training* untuk membandingkan arsitektur apa yang paling optimal ketika proses pembelajaran telah dilakukan. Tahapan *testing* untuk pengujian prediksi dan akurasi pengenalan data, serta *validation model* untuk memastikan model yang dihasilkan memiliki klasifikasi produk yang handal. Pada akhirnya, proses pengujian dan validasi ini akan memberikan informasi tentang ketepatan model dalam mengenali produk. Semua langkah ini dirancang untuk memperoleh model yang memiliki kualitas dan performa terbaik dalam klasifikasi produk, serta memiliki prediksi yang akurat.



Gambar 2. Alur Penelitian

## 2.5 Confusion matriks

Confusion matriks merupakan metode yang digunakan untuk mengevaluasi model CNN terhadap ketepatan klasifikasi pengenalan. Tabel 3 berikut merupakan ilustrasi pola confusion matriks.

Tabel 3. Ilustrasi Confusion matriks

	Predict Label (Yes) → Defect	Predict Label (No) → Perfect
True Label (Yes) → Defect	TP	FN
True Label (No) → Perfect	FP	TF

True Negative (TN) mengindikasikan data yang akurat dalam klasifikasi negatif atau yang benar-benar dianggap sebagai data negatif. True Positive (TP) menggambarkan data yang terklasifikasikan secara benar. False Positive (FP) mewakili data yang salah diklasifikasikan sebagai positif. Sementara itu, False Negative (FN) menunjukkan kesalahan data yang dikategorikan sebagai negatif. Dalam rangkaian penelitian ini, menggunakan matriks nilai rata-rata (macro avg) untuk dua kategori, yakni "defect" dan "perfect". Penentuan besarnya akurasi, presisi, recall, dan f1-score melibatkan perhitungan menggunakan formula 1, 2, 3, dan 4 seperti berikut (Antoni, Rohana and Pratama, 2023).

$$Accuracy (\%) = \frac{(TPositif + TNegatif)}{(TPositif + FPositif + TNegatif + FNegatif)} \quad (1)$$

$$Precision (\%) = \frac{TPositif}{(TPositif + FPositif)} \quad (2)$$

$$\text{Recall} (\%) = \frac{T\text{Positif}}{(T\text{Positif} + F\text{Negatif})} \quad (3)$$

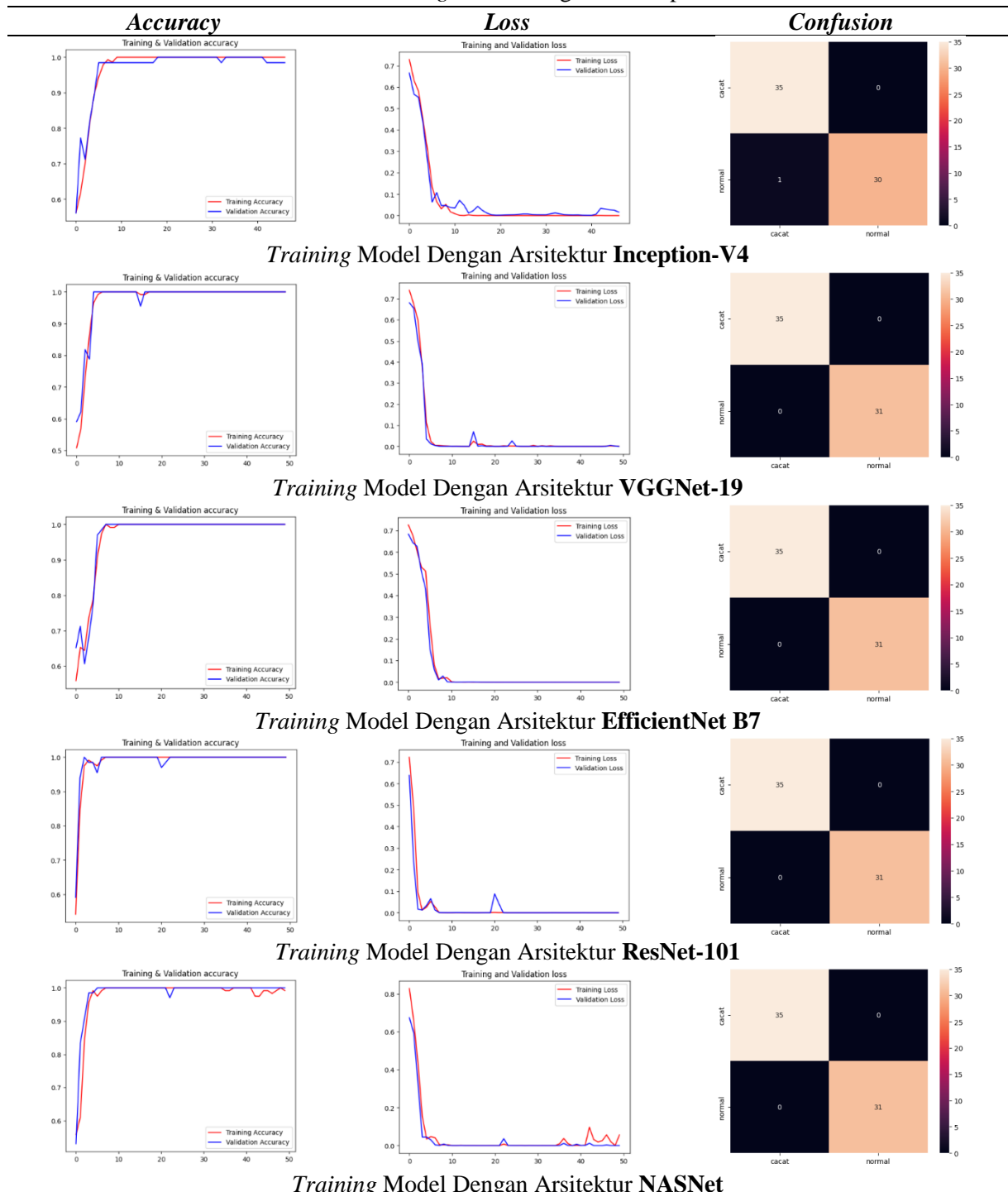
$$F1 - \text{Score} (\%) = \frac{(2 \times \text{recall} \times \text{precision})}{(\text{recall} + \text{precision})} \quad (4)$$

### 3. Hasil dan Pembahasan

#### 3.1. Hasil Training dan Evaluasi Model

Hasil *training* pada setiap model arsitektur ditetapkan dengan *epochs* sebanyak 50 *step*, *learning rate* sebagai optimalisasi untuk memperbarui bobot selama pembelajaran model sebesar 0,001, *batch size* sebanyak 10 *step*, dan *callback* yang digunakan yakni *checkpoint callbacks*.

**Tabel 4.** Hasil Training Model dengan Beberapa Arsitektur



**a. Hasil Training Terhadap Perbandingan Beberapa Arsitektur CNN**

Pada tahapan ini dilakukan perbandingan nilai *training* terhadap beberapa arsitektur yang telah dirancang pada tahap sebelumnya. Arsitektur yang dibandingkan yakni Inception-V4, VGGNet-19, EfficientNet B7, ResNet-101, dan NASNet. Hasil *training* dapat dilihat pada Tabel 4. Berdasarkan Tabel 4 diatas, didapatkan bahwa arsitektur inception-v4 mengalami *underfitting* pada titik iterasi ke 10 hingga 15. Sementara itu pada arsitektur ResNet-101 mengalami *overfitting* pada titik iterasi 10 hingga 20. Arsitektur NasNet mengalami *underfitting* pada parameter *training accuracy* Arsitektur VGGNet-19 dan EfficientNetB7 telah terhindar dari *underfitting* dan *overfitting* yang kemudian akan dilakukan analisis kembali pada tahap *testing*. Oleh karena itu model dengan performa baik sementara yakni Arsitektur VGGNet-19 dan EfficientNetB7. Selanjutnya akan dilakukan perbandingan dengan parameter lainnya untuk memutuskan arsitektur terbaik dalam pengenalan produk.

**b. Akumulasi Training dengan Beberapa Arsitektur**

Berdasarkan hasil *training* yang diperoleh setelah proses pembelajaran model pada Tabel 4, peneliti merincikan hasil akumulasi *training* pada Tabel 5 berikut sebagai langkah untuk menganalisis pengaruh penggunaan arsitektur selama proses pembelajaran.

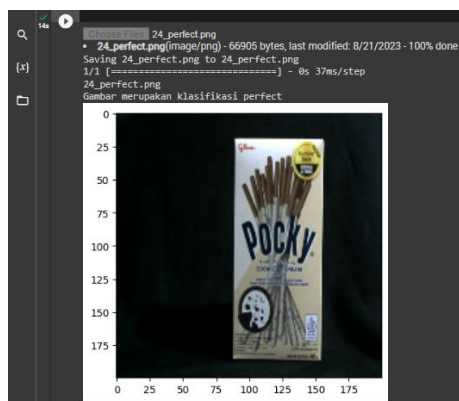
**Tabel 5.** Hasil Akumulasi *Training Model*

	Parameter Training				Confusion Matriks (Weight)			
	Training Loss	Training Acc.	Validation Loss	Validation Acc.	Acc.	Precision	Recall	F1-Score
<b>Inception-V4</b>	9.996e-06	1.00	0.0165	0.9848	0.98	0.99	0.98	0.98
<b>VGGNet-19</b>	9.2117e-04	1.00	6.836e-05	1.00	1.00	1.00	1.00	1.00
<b>EfficientNet B7</b>	3.363e-06	1.00	2.125e-05	1.00	1.00	1.00	1.00	1.00
<b>ResNet-101</b>	1.341e-06	1.00	2.302e-05	1.00	1.00	1.00	1.00	1.00
<b>NASNet</b>	7.833e-04	1.00	7.5376e-05	1.00	1.00	1.00	1.00	1.00

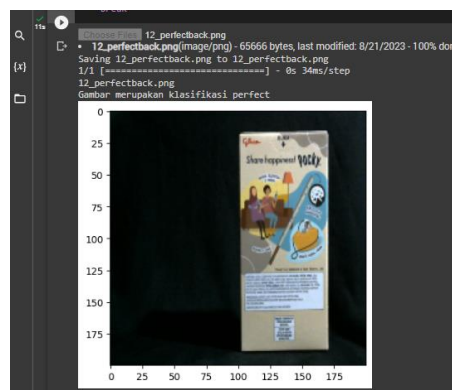
Berdasarkan Tabel 5 jika ditinjau berdasarkan parameter *training*, hasil *validation* dan *training* terbaik yakni arsitektur EfficientNet-B7 dengan akurasi *training* dan *validation* mencapai 1.00 atau 100%, sedangkan pada nilai *loss* menghasilkan nilai yang paling minimum dengan nilai 2.125e-05. Jika ditinjau berdasarkan parameter *confusion*, nilai terbaik adalah arsitektur VGGNet-19, EfficientNet-B7, ResNet-101, NASNet dengan akurasi pada setiap parameter mencapai 1.00 atau 100%. Berdasarkan hal tersebut, peneliti menganalisis lebih lanjut pada tahapan *testing* untuk mendapatkan model pembelajaran arsitektur terbaik terhadap pengenalan citra.

**c. Hasil Testing Terhadap Perbandingan Beberapa Arsitektur CNN**

Pada tahapan ini dilakukan untuk menganalisis proses *testing* terhadap beberapa arsitektur yang telah dibentuk pada proses *training*. Tahapan ini dilakukan dengan meng-*upload* data *testing* pada Google Colab untuk mengetahui keberhasilan pengenalan model terhadap ketepatan deteksi berdasarkan kategori yang telah ditetapkan. Hasil *testing* ditampilkan pada Gambar 3 dan Gambar 4 berikut.



(a) Testing Sample "Perfect" Depan



(b) Testing Sample "Perfect" Belakang

Gambar 3. Keberhasilan Testing Sample Data Perfect



(a) Testing Sample “Defect” Depan



(b) Testing Sample “Defect” Belakang

Gambar 4. Keberhasilan Testing Sample Data Defect

Tahapan pengujian dilakukan dengan jumlah data sebanyak 60 data yang terdiri atas 30 data “perfect” dan 30 data lainnya merupakan kategori “defect”. Berdasarkan keberhasilan pengujian *testing* pada Gambar 3 dan Gambar 4, didapatkan hasil *testing* sebagaimana yang telah tertera pada Tabel 6 sebagai berikut.

Tabel 6. Hasil Akumulasi Training Model

	Deteksi Bagian Depan		Deteksi Bagian Belakang		Total Benar	Akurasi rata-rata
	Benar deteksi “Perfect”	Benar deteksi “Defect”	Benar deteksi “Perfect”	Benar deteksi “Defect”		
Inception-V4	14	15	13	13	55	91,67%
VGGNet-19	14	15	14	14	57	95,00%
EfficientNet B7	15	15	14	15	59	98,34%
ResNet-101	14	14	14	15	57	95,00%
NASNet	13	14	12	13	52	86,67%

Berdasarkan pengujian pada Tabel 6, nilai akurasi keberhasilan didapatkan berdasarkan perbandingan antara total deteksi benar dengan total data yang diujikan yang kemudian dikalikan dengan 100%. Berdasarkan hasil *testing* akurasi, ditemukan bahwa model *Convolutional Neural Network* (CNN) yang paling optimal dalam klasifikasi produk *snack box* adalah model yang menggunakan arsitektur EfficientNet-B7. Dari total 60 data uji, model EfficientNet-B7 mampu mendeteksi dengan benar 59 data, hanya mengalami 1 kesalahan prediksi. Arsitektur EfficientNet-B7 menghasilkan akurasi yang lebih optimal dibandingkan arsitektur lain, dengan akurasi rata-rata mencapai 98,34%.

Arsitektur Efficient Net merupakan model terbaik dalam klasifikasi produk *snack box*, hal ini dikarenakan Efficient Net dilengkapi fitur *blok Mobile Convolution* (MBConv) yang dirancang untuk meningkatkan representasi fitur dengan memanfaatkan mekanisme *gating research citra*. Struktur ini memungkinkan model untuk mempelajari fitur-fitur yang lebih kompleks dari data. Selain itu, Efficient Net memanfaatkan teknik NAS (*Automated Neural Architecture Search*) dalam perancangannya. Proses ini secara otomatis melakukan kompleksitas struktur arsitektur terbaik, sehingga model dapat mendeteksi cacat pada kemasan *snack box*.

#### 4. Kesimpulan

Berdasarkan berbagai aspek pengujian yang telah dilaksanakan, penelitian ini fokus pada pengenalan produk *snack box* dengan 2 kategori yakni “defect” dan “perfect” untuk identifikasi kecacatan kemasan *snack box*. Pada penelitian ini dilakukan analisa perbandingan model CNN dengan beberapa penambahan arsitektur untuk mendapatkan model terbaik. Arsitektur yang diujikan meliputi Arsitektur Inception-V4, VGGNet-19, EfficientNet B7, ResNet-101, dan NASNet. Berdasarkan hasil

parameter *training* dan *confusion matriks*, arsitektur paling optimal dalam pembelajaran model yakni arsitektur VGGNet-19, EfficientNet-B7, ResNet-101, NASNet dengan akurasi pada setiap parameter mencapai 100%. Namun jika ditinjau berdasarkan *testing*, model *Convolutional Neural Network* (CNN) yang paling optimal dalam klasifikasi produk *snack box* adalah model yang menggunakan arsitektur EfficientNet-B7. Dari total 60 data uji, model EfficientNet-B7 mampu mendeteksi dengan benar 59 data, hanya mengalami 1 kesalahan prediksi. Arsitektur EfficientNet-B7 menghasilkan akurasi yang lebih tinggi dibandingkan dengan arsitektur lain, dengan tingkat akurasi rata-rata mencapai 98,34%. Meskipun demikian perlu adanya penambahan *dataset* dan variasi parameter yang beragam sehingga klasifikasi lebih optimal. Dimasa yang akan datang penelitian berharap adanya inovasi penunjang untuk menciptakan sistem sortir sebagai pemilahan produk “*defect*” dan “*perfect*” pada *snack box* yang terintegrasi dengan sistem mikrokontroler sehingga *output* yang dihasilkan dapat berpengaruh terhadap problematika klasifikasi produk dengan skala yang lebih luas.

### Daftar Pustaka

- Antoni, A., Rohana, T. and Pratama, A. R. (2023) ‘Implementasi Algoritma Convolutional Neural Network Untuk Klasifikasi Citra Kemasan Kardus Defect dan No Defect’, *Building of Informatics, Technology and Science (BITS)*, 4(4), pp. 1941–1950. doi: 10.47065/bits.v4i4.3270.
- Ardhianto, Peter and Kurniawan, M. (2016) ‘Strategi Iklan Produk Minuman Coca-Cola di Indonesia Dalam Mempertahankan Dan Menguatkan Citra’, *Jurnal Widya Sosiopolitika*, 7(2), pp. 91–106.
- Chen, Y. et al. (2021) ‘Surface Defect Detection Methods for Industrial Products: A Review’, *Applied Sciences (Switzerland)*, 11(16).
- Hardi, N. (2022) ‘Komparasi Algoritma MobileNet Dan Nasnet Mobile Pada Klasifikasi Penyakit Daun Teh’, *Reputasi: Jurnal Rekayasa Perangkat Lunak*, 3(1), pp. 50–55. doi: 10.31294/reputasi.v3i1.1313.
- Kementerian Perdagangan Republik Indonesia (2017) *Desain Kemasan Produk Makanan Olahan*. Available at: [djpen.kemendag.go.id/](http://djpen.kemendag.go.id/).
- Kusumawardani, R. and Karningsih, P. D. (2021) ‘Deteksi Dan Klasifikasi Cacat Kemasan Kaleng Menggunakan Convolutional Neural Network’, *PROZIMA (Productivity, Optimization and Manufacturing System Engineering)*, 4(1), pp. 1–11.
- Kusumawati, D. N. I., Kusumah, W. I. and Wibisono T, R. W. (2022) ‘Analisis Desain Kemasan Produk Umkm Makanan Tradisional Lempur Berbahan Alami Memiliki Daya Tarik Dan Ketahanan Mutu Produk’, *Jurnal Nawala Visual*, 4(1), pp. 41–49. doi: 10.35886/nawalavisual.v4i1.340.
- Nurhayati, O. D., Eridani, D. and Tsalavin, M. H. (2022) ‘Sistem Isyarat Bahasa Indonesia ( SIBI ) Metode Convolutional Neural Network Sequential Secara Real Time A Real-Time Indonesian Language Sign System’, *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, 9(4), pp. 819–828. doi: 10.25126/jtiik.202294787.
- Perdani, W. R., Magdalena, R. and Caecar, N. K. (2022) ‘Deep Learning untuk Klasifikasi Glaukoma dengan menggunakan Arsitektur EfficientNet’, *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 10(2), p. 322. doi: 10.26760/elkomika.v10i2.322.
- Prastiwinarti, W. et al. (2023) ‘Perancangan Pemanfaatan Machine Learning untuk Deteksi Cacat Kemasan Produk’, *SNIV: Seminar Nasional Inovasi Vokasi*, 2(1), pp. 97–102.
- Richo et al. (2023) ‘Analisis Pengaruh Optimizer pada Model CNN untuk Identifikasi Cacat pada Perekat Kemasan’, *Jurnal Sisfotenika*, 13(2), pp. 217–229.
- Rizki, A. M. and Marina, N. (2019) ‘Klasifikasi Kerusakan Bangunan Sekolah Menggunakan Metode Convolutional Neural Network Dengan Pre-Trained Model Vgg-16’, *Jurnal Ilmiah Teknologi dan Rekayasa*, 24(3), pp. 197–206. doi: 10.35760/tr.2019.v24i3.2396.

Ronggo, R., Pratomo, B. and Palupingsih, P. (2023) 'Analisis Perbandingan Performa Model Klasifikasi Kesehatan Daun Tomat menggunakan Arsitektur VGG, MobileNet, dan Inception V3', *Jurnal Ilmu Komputer Agri-Informatika*, 10, pp. 98–110.

Satria, M. R. and Pardede, J. (2022) 'Image Captioning Menggunakan Metode Inception-V3 dan Transformer', *Jurnal Tera*, 2(2), pp. 1–14.

Shidiq, A. L. A., Suhartono, E. and Saidah, S. (2022) 'Klasifikasi Kecacatan Ban Untuk Mengendalikan Kualitas Produk Menggunakan Model CNN Dengan Arsitektur VGG-16', *e-Proceeding of Engineering*, 8(6), pp. 3216–3225.

Wahid, M. I., Mustamin, S. A. and Lawi, D. A. (2021) 'Identifikasi Dan Klasifikasi Citra Penyakit Daun Tomat Menggunakan Arsitektur Inception V4', *Konferensi Nasional Ilmu Komputer (KONIK)*, 05(2019), pp. 257–264. Available at: <https://prosiding.konik.id/index.php/konik/article/view/61>.

Wardana, B. K., Rachmawati, E. and Wirayuda, T. A. B. (2021) 'Pengenalan Gestur Tangan Statis Menggunakan CNN Dengan Arsitektur Efficient-Net B4', *e-Proceeding of Engineering Informatics Technology*, 8(2), pp. 3446–3463.

Halaman ini sengaja dikosongkan