

# **Evaluasi Kinerja Model *Inception Resnet-V2* dan *Inception-V4* dalam Mengklasifikasi Kualitas Biji Kakao**

**MUHAMMAD ICHWAN, HANIFAH SUMANTRI**

Program Studi Informatika, Institut Teknologi Nasional Bandung  
Email : [ichwan@itenas.ac.id](mailto:ichwan@itenas.ac.id)

*Received 15 November 2023 | Revised 8 Maret 2024 | Accepted 13 Mei 2024*

## **ABSTRAK**

*Kakao adalah komoditas ekspor penting bagi Indonesia, dan untuk memenuhi standar mutu, diperlukan sistem klasifikasi biji kakao. Penelitian menggunakan metode CNN dengan fokus pada arsitektur Inception Resnet-V2 dan Inception-V4 untuk mengatasi permasalahan klasifikasi citra kualitas biji kakao. Keduanya merupakan inovasi dari keluarga Inception, dengan Inception Resnet-V2 menggabungkan elemen dari ResNet dan Inception, sementara Inception-V4 memperbaiki performa dengan mengurangi kompleksitas arsitektur. Tujuan penelitian adalah untuk mengevaluasi performa model dari kedua arsitektur tersebut dalam konteks klasifikasi biji kakao. Hasil penelitian menunjukkan bahwa arsitektur Inception Resnet-V2 dengan pengaturan optimizer RMSprop, batch size 16, dan learning rate 0,0001 memiliki performa terbaik, dengan nilai akurasi 91,87%, presisi 92,00%, recall 92,00%, f1-score 92,00%, dan ROC AUC Score 0,950.*

**Kata kunci:** *Deep Learning, CNN, Inception Resnet-V2, Inception-V4, Biji Kakao*

## **ABSTRACT**

*Cocoa is an important export commodity for Indonesia, and to meet quality standards, a cocoa bean classification system is needed. The research utilizes CNN method, focusing on Inception Resnet-V2 and Inception-V4 architectures to address cocoa bean quality image classification issues. Both are innovations from the Inception family, with Inception Resnet-V2 combining elements of ResNet and Inception, while Inception-V4 improves performance by reducing architectural complexity. The research objective is to evaluate the performance of models from both architectures in the context of cocoa bean classification. Research results show that the Inception Resnet-V2 architecture with RMSprop optimizer settings, batch size 16, and learning rate 0.0001 performs the best, with an accuracy of 91.87%, precision of 92.00%, recall of 92.00%, f1-score of 92.00%, and ROC AUC Score of 0.950.*

**Keywords:** *Deep Learning, CNN, Inception Resnet-V2, Inception-V4, Cocoa Beans*

## 1. PENDAHULUAN

Kakao merupakan komoditas ekspor penting bagi Indonesia di pasar internasional. Meskipun ekspor biji kakao Indonesia cukup besar, namun keunggulan komparatifnya masih kalah dibandingkan dengan negara-negara produsen utama lainnya. Salah satu faktor penyebabnya adalah kualitas biji kakao yang belum sebaik negara pesaing. Oleh karena itu, pemerintah Indonesia harus terus memperhatikan dan meningkatkan kualitas biji kakao. Untuk itu, pemerintah telah menetapkan standar mutu, seperti yang diatur dalam standar SNI 01-2323-2008, untuk memastikan bahwa biji kakao yang diekspor memenuhi persyaratan yang ditetapkan. Standar ini mencakup definisi, klasifikasi, persyaratan mutu, prosedur pengambilan sampel, metode pengujian, persyaratan label, metode kemasan, dan rekomendasi lainnya.

Kriteria untuk menilai kualitas biji kakao yang baik termasuk kematangan, fermentasi yang optimal (ditandai dengan kekeringan, warna coklat, aroma asam, serta tanpa keberadaan warna ungu dan abu-abu ketika dibelah), berat kering tidak melebihi 1 gram, ukuran seragam, cangkang utuh, dan warna yang merah kecokelatan. Sebelum biji kakao dapat diekspor atau diperdagangkan, para eksportir harus menyediakan sampel biji kepada petugas dari Balai Pengujian dan Sertifikasi Mutu Barang (BPSMB) untuk diperiksa secara berkala sesuai dengan Standar Nasional Indonesia (SNI) demi mendapatkan sertifikasi. Proses pengujian dilakukan secara manual, yang dapat mengakibatkan kesalahan dan memakan waktu cukup lama untuk menilai kualitasnya **(Winarto, dkk., 2021)**.

Metode pengklasifikasian gambar bisa dimanfaatkan untuk menilai kualitas biji kakao. Pendekatan ini melibatkan teknik pembelajaran mesin dan visi komputer yang bertujuan untuk mengekstraksi fitur serta informasi dari setiap lapisan gambar berwarna, sehingga memungkinkan untuk mengelompokkan gambar-gambar tersebut berdasarkan fitur yang ditemukan **(Abed, dkk., 2019)**. *Deep learning* telah menjadi metode yang umum digunakan dalam kasus klasifikasi gambar. Dengan kemampuannya, *deep learning* telah berhasil menunjukkan kinerja klasifikasi yang mengesankan dengan mampu mengidentifikasi fitur-fitur yang relevan untuk memecahkan masalah klasifikasi gambar **(Kadhim & Abed, 2020)**. Ada berbagai metode yang tersedia dalam ranah *deep learning*, di antaranya adalah *convolutional neural network* (CNN). CNN telah menjadi pilihan yang populer dan banyak digunakan dalam beberapa tahun terakhir untuk menyelesaikan berbagai masalah yang kompleks dan bervariasi, termasuk pengenalan gambar dan klasifikasi, dengan memanfaatkan serangkaian lapisan respons **(Kadhim & Abed, 2020)**. Metode CNN dapat menjadi solusi permasalahan pengklasifikasian kualitas biji kakao.

Pada 2016, para peneliti Google mengemukakan struktur arsitektur *Inception Resnet-V2* dan *Inception-V4* **(Szegedy, dkk., 2016)**. *Inception Resnet-V2* merupakan sebuah struktur *Convolutional Neural Network* (CNN) yang dikembangkan dari arsitektur *Inception*, namun dengan penambahan koneksi *residual*. *Resnet* memperkenalkan *residual connection* untuk pelatihan jaringan yang lebih dalam dan akurat, sementara blok *Inception* mengumpulkan informasi dari berbagai skala *input* citra. Kombinasi ini meningkatkan kinerja arsitektur. Berdasarkan hasil pengujian yang dilakukan, arsitektur ini menghasilkan tingkat *error* yang lebih rendah dibandingkan *Inception-V3* **(Rochmawanti, dkk., 2021)**. Arsitektur *Inception-V4* adalah versi terbaru dari *Inception*. *Inception-V4* adalah kerangka jaringan neural konvolusional mendalam yang dibangun berdasarkan versi seri *Inception* sebelumnya dengan menyederhanakan arsitektur, menambahkan lapisan induk, dan menggunakan lebih banyak modul *Inception* daripada *Inception-V3* dengan lebih sederhana dan lebih seragam **(Al Husaini, dkk., 2022)**.

*Inception-ResNet-V2* dan *Inception-V4* juga memiliki performa yang baik, masing-masing mengungguli performa pada kumpulan data validasi *ImageNet* (Szegedy, dkk., 2016). Hal ini terlihat pada penelitian (Nugraha, dkk., 2022) dengan judul "Deteksi Objek Dan Jenis Burung Menggunakan *Convolutional Neural Network* Dengan Arsitektur *Inception Resnet-V2*" dengan tingkat akurasi yang mencapai 93,33%, 98,28%, dan 97,89% secara berturut-turut pada setiap percobaan, penelitian ini menunjukkan bahwa penerapan arsitektur *Inception Resnet-V2* untuk menguji akurasi menggunakan tiga saluran warna (RGB) lebih efektif daripada menggunakan saluran warna tunggal atau skala abu-abu. Hal ini terbukti melalui hasil pengujian yang dilakukan dalam penelitian ini. Penelitian lain yang dilakukan oleh (Wahid, dkk., 2021) berjudul "Identifikasi Dan Klasifikasi Citra Penyakit Daun Tomat Menggunakan Arsitektur *Inception-V4*" dimana penelitian tersebut menggunakan model pembelajaran *Inception-V4* untuk mengidentifikasi dan mengklasifikasikan citra penyakit daun tomat dan mencapai akurasi 90,00% dalam menentukan jenis penyakit tomat. Pada penelitian (Szegedy, dkk., 2016) dengan judul "*Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning*" konfirmasi keakuratan model belum dapat dipastikan. Dalam penelitian membandingkan *Inception-V3*, *Inception-V4* dan *Inception Resnet*, hasilnya diperoleh dengan terlebih dahulu mengevaluasi 1 varian. *Error Inception-V4* memiliki nilai sebesar 20.0% dan *error Inception Resnet-V2* memiliki nilai sebesar 19.9% yang merupakan nilai terendah dibandingkan varian *Inception* lainnya. Hal ini menegaskan bahwa *Inception Resnet-V2* dan *Inception-V4*, varian terbaru dari arsitektur *Inception*, menunjukkan kinerja yang unggul dibandingkan dengan variasi *Inception* lainnya. Penelitian ini merupakan langkah awal untuk membandingkan kinerja kedua arsitektur ini menggunakan dataset *cocoa beans* yang belum pernah digunakan sebelumnya. Studi ini bertujuan untuk menemukan model terbaik diantara kedua arsitektur tersebut dalam mengklasifikasikan citra dan menganalisis hasil eksperimen yang telah dilakukan.

Penelitian ini akan menggunakan Dataset *Cocoa Beans* yang diperoleh dari *Kaggle*, yang dikembangkan oleh Khawaritzmi Abdallah Ahmad. Dataset ini akan digunakan untuk melatih arsitektur *Inception Resnet-V2* dan *Inception-V4* dalam melakukan klasifikasi kualitas biji kakao. Penelitian ini akan mencoba variasi *hyperparameter* pada kedua arsitektur tersebut untuk mengevaluasi performanya. Diharapkan penelitian ini dapat menentukan arsitektur terbaik untuk melakukan klasifikasi kualitas biji kakao.

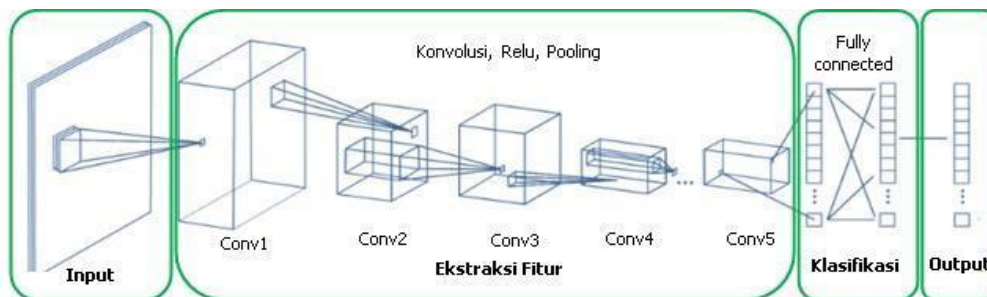
## 2. METODE PENELITIAN

### 2.1 Deep Learning

*Deep learning* merupakan suatu pendekatan dalam bidang *machine learning* yang memiliki kemampuan kuat untuk secara otomatis mempelajari representasi fitur dari data. Khususnya, teknik ini telah menghasilkan kemajuan signifikan dalam deteksi objek. *Deep learning* telah terbukti menjadi teknologi yang sangat canggih dalam analisis data berupa gambar, suara, dan deteksi objek. Selain itu, *deep learning* juga memungkinkan pembelajaran model dari beberapa lapisan pemrosesan yang menghasilkan representasi berbagai tingkat abstraksi (Windiawan, dkk., 2021). *Deep Learning* memungkinkan model komputasi untuk mempelajari representasi yang sangat kompleks, halus, dan abstrak, mendorong kemajuan yang signifikan dalam berbagai masalah seperti pengenalan visual, deteksi objek, pengenalan ucapan, pemrosesan bahasa alami, analisis citra medis, penemuan obat, dan genomic (Liu, dkk., 2020).

## 2.2 Convolutional Neural Network (CNN)

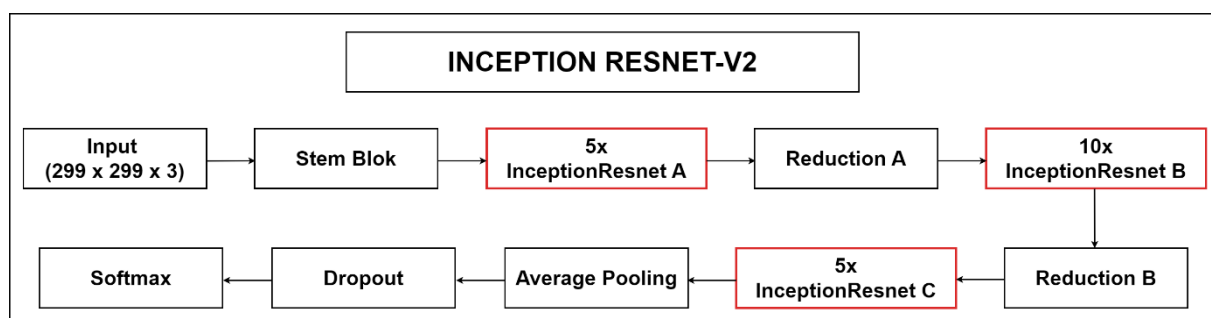
*Convolutional Neural Network (CNN) adalah bentuk jaringan saraf tiruan feed-forward yang spesifik. Arsitekturnya terdiri dari beberapa lapisan, termasuk lapisan konvolusional, lapisan pooling, dan lapisan terhubung sepenuhnya (Abed, dkk., 2019). CNN telah digunakan secara meluas dalam berbagai bidang, termasuk visi komputer, pemrosesan bahasa alami, identifikasi wajah, dan lain-lain. Struktur CNN terinspirasi dari neuron yang ada pada otak manusia dan hewan, dengan kesamaan dengan jaringan saraf konvensional (Alzubaidi, dkk., 2021). Struktur CNN melibatkan input data, proses ekstraksi fitur, proses klasifikasi, dan output. Proses ekstraksi fitur dalam CNN terdiri dari beberapa lapisan tersembunyi, termasuk lapisan konvolusi, fungsi aktivasi (ReLU), dan pooling. CNN beroperasi secara hierarkis, dimana hasil output dari lapisan konvolusi pertama dijadikan sebagai input untuk lapisan konvolusi selanjutnya. Proses klasifikasi melibatkan penggunaan fungsi terhubung penuh dan fungsi aktivasi (softmax) untuk menghasilkan output berupa hasil klasifikasi (Winarto, dkk., 2021). Gambar 1 memperlihatkan ilustrasi dari struktur CNN.*



**Gambar 1. Arsitektur Convolutional Neural Network (Arrofiqoh & Harintaka, 2018)**

## 2.3 Inception Resnet-V2

*Inception-ResNet-V2 merupakan sebuah struktur Convolutional Neural Network (CNN) yang dikembangkan dari keluarga arsitektur Inception, namun dengan penambahan koneksi residual (Winarto, dkk., 2021). Inception-ResNet-V2 memanfaatkan semua keunggulan metode lainnya sambil mempertahankan efisiensi komputasinya (Nugraha, dkk., 2022). Gambar 2 menampilkan ilustrasi dari struktur Inception Resnet-V2.*

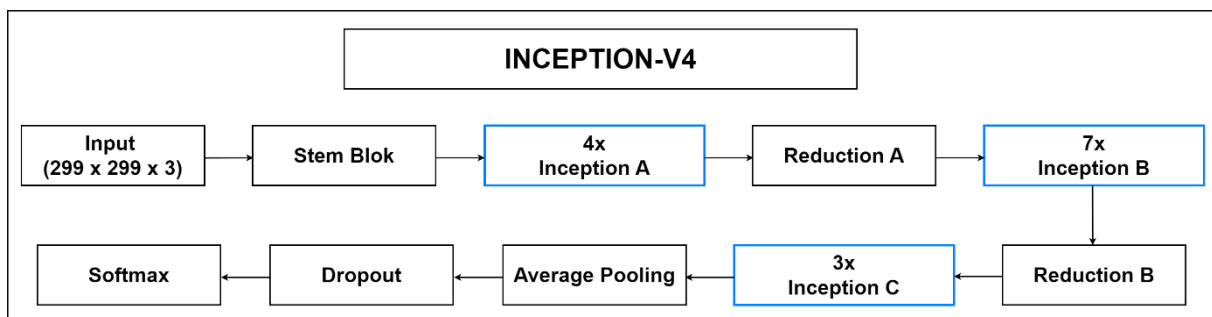


**Gambar 2. Blok Proses pada Arsitektur Inception Resnet-V2**

Gambar 2 menampilkan setiap bagian blok dari *Inception Resnet-V2*, termasuk *Stem Block*, *Inception Resnet-A Block*, *Reduction A*, *Inception Resnet-B Block*, *Reduction B*, dan *Inception Resnet-C Block*.

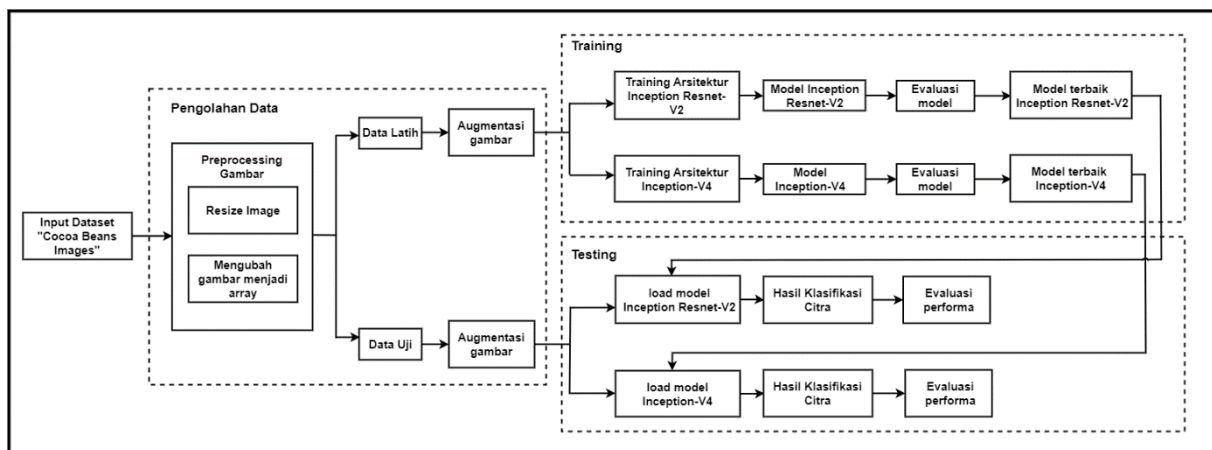
## 2.4 Inception-V4

*Inception-V4* merupakan varian murni dari *Inception* tanpa menggunakan koneksi *residual*, namun memiliki kinerja pengenalan gambar yang hampir sebanding dengan *Inception ResNet-V2*. Arsitektur *Inception-V4* lebih sederhana dibandingkan dengan *Inception-V3*, dengan modul yang lebih seragam. *Inception-V4* terdiri dari tiga modul yang memiliki kemiripan dengan *Inception ResNet-V2* (Akbar, dkk., 2020). *Inception-V4* adalah perkembangan dari versi sebelumnya, terutama *Inception-V3*. *Inception-V4* terdiri dari beberapa blok, termasuk *stem*, blok *inception A*, blok *inception B*, dan *inception C*. Terdapat juga *reduction block* dalam *Inception-V4* yang berfungsi untuk mengubah ukuran lebar dan tinggi grid. Perbedaan utama dengan versi sebelumnya terletak pada struktur blok asli dan beberapa penyesuaian kecil pada blok C. Dimensi *input* citra yang diterima oleh *Inception-V4* adalah 299x299x3. Ilustrasi arsitektur *Inception-V4* ditunjukkan pada Gambar 3.



Gambar 3. Blok Proses pada Arsitektur *Inception-V4*

## 2.5 Blok Diagram



Gambar 4. Blok Diagram


Gambar 4 menampilkan diagram blok terdapat 3 tahapan yaitu pengolahan data, proses *training* dan *testing*. Pada tahap pengolahan data akan dilakukan *input* citra dari *datasets*, lalu dilakukan proses *preprocessing* untuk perbaikan citra, selanjutnya dataset akan dibagi menjadi data pelatihan dan data pengujian. Selama proses *training*, citra dari data pelatihan dimasukkan, kemudian dilakukan dua kali pelatihan dengan menggunakan dua arsitektur yang berbeda, yaitu arsitektur *Inception Resnet-V2* dan *Inception-V4*, yang menghasilkan model-model dari kedua arsitektur tersebut, kemudian dilakukan evaluasi model untuk mengetahui masing-masing kinerja dari beberapa model yang telah didapatkan, lalu setelah-nya hasil akhir dari proses *training* adalah mendapatkan model terbaik dari kedua arsitektur tersebut. Pada tahap *testing*, dilakukan *input* citra dari data uji kemudian akan dilakukan pengujian dengan


model terbaik *Inception Resnet-V2* dan *Inception-V4* yang telah melalui proses pelatihan sebelumnya. Hasil dari proses pengujian adalah klasifikasi kualitas biji kakao, apakah masuk ke dalam salah satu dari enam kategori kelas dataset biji kakao. Kemudian proses terakhir akan dilakukan evaluasi kinerja dari kedua model terbaik tersebut.

## 2.6 Deskripsi *Datasets*

Dataset yang digunakan dalam penelitian diunduh dari platform *Kaggle* dengan judul "*Cocoa Beans Images Dataset*" yang dikembangkan oleh Khawarizmi Abdallah Ahmad. Dataset ini terdiri dari 614 gambar biji kakao dalam format JPG yang terbagi ke dalam enam kelompok kualitas biji kakao. Jumlah gambar dalam setiap kelas bervariasi antara 100 hingga 105. Informasi detail mengenai dataset ini dapat ditemukan dalam Tabel 1.

**Tabel 1. Dataset *Cocoa Beans***

No	Kelas	Gambar	Jumlah Data	Deskripsi
1	<i>Bean Fraction Cocoa</i>		100	Kategori untuk biji kakao yang pecah
2	<i>Broken Beans Cocoa</i>		100	Kategori untuk biji kakao yang mengalami kerusakan
3	<i>Fermented Cocoa</i>		105	kategori untuk biji kakao yang telah difermentasi
4	<i>Moldy Cocoa</i>		105	kategori untuk biji kakao yang terinfeksi jamur
5	<i>Unfermented Cocoa</i>		100	kategori untuk biji kakao yang belum mengalami proses fermentasi

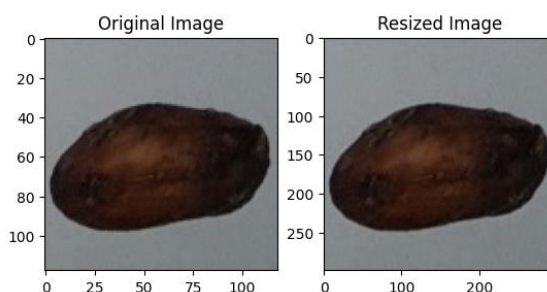
No	Kelas	Gambar	Jumlah Data	Deskripsi
6	<i>Whole Beans Cocoa</i>		104	kategori untuk biji kakao yang masih utuh atau tidak rusak
<b>Total Dataset</b>			<b>614</b>	

## 2.7 Preprocessing Data

Sebelum masuk ke dalam proses selanjutnya yaitu proses pelatihan, disini akan dilakukan proses *preprocessing* terlebih dahulu pada citra. Pemrosesan awal data terdiri dari mengubah ukuran data, pembagian data menjadi data latih dan data uji, dan augmentasi.

### A. *Resize Image*

*Resize* adalah langkah untuk menyesuaikan ukuran gambar. Dalam konteks ini, gambar dengan piksel yang bervariasi akan diubah menjadi 299x299 piksel dengan 3 *channel* warna RGB. Ukuran yang disesuaikan ini akan digunakan sebagai citra *input* untuk tahap berikutnya, yang melibatkan proses pelabelan di mana jenis *cocoa* diberi label 0 hingga 5 tergantung pada kondisinya. Hasil *resize image* pada sebuah sample citra ditunjukkan pada Gambar 5.



**Gambar 5. Hasil *Resize* pada *Sample* Citra Biji Kakao**

### B. Pembagian Dataset

Dataset akan dibagi menjadi proses pelatihan dan pengujian, dengan masing-masing pemisahan sebesar 80% dan 20%, penyebab pemilihan rasio 80:20 sebagai titik awal yang optimal dalam pemisahan dataset adalah karena kemampuannya untuk secara substansial meningkatkan akurasi. Pada intinya, data pelatihan mencerminkan mayoritas variasi dalam dataset (Prashanth, dkk., 2020). Berikut rincian sebaran dataset yang tercantum pada Tabel 2.

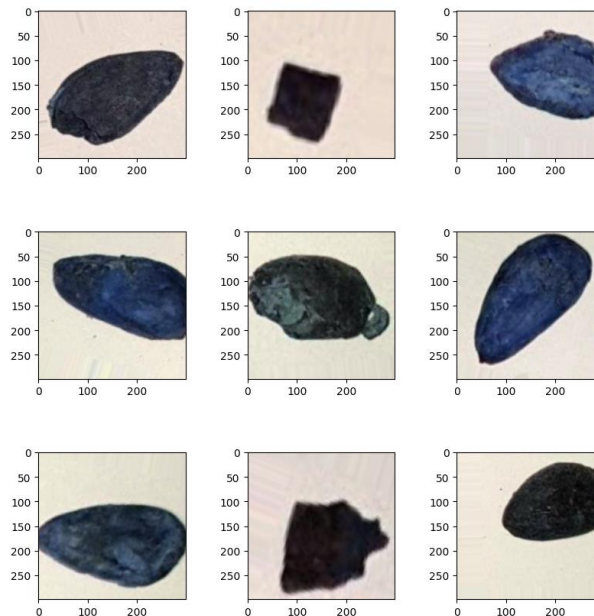
**Tabel 2. Pembagian Dataset**

No	Kelas	<i>Training</i> (80%)	<i>Testing</i> (20%)	Jumlah
1	<i>Bean Fraction Cocoa</i>	80	20	100
2	<i>Broken Beans Cocoa</i>	80	20	100

No	Kelas	Training (80%)	Testing (20%)	Jumlah
3	<i>Fermented Cocoa</i>	84	21	105
4	<i>Moldy Cocoa</i>	84	21	105
5	<i>Unfermented Cocoa</i>	80	20	100
6	<i>Whole Beans Cocoa</i>	83	21	104
<b>Total</b>		<b>491</b>	<b>123</b>	<b>614</b>

### C. *Augmentasi*

Pada proses *augmentasi* disini akan dilakukan proses *rotation range*, *zoom range*, *width and height shift*, dan *horizontal flip*. Proses *augmentasi* dilakukan menggunakan modul keras *ImageDataGenerator*. Hasil *augmentasi* data pada sebuah *sample* citra ditunjukkan pada Gambar 6.



**Gambar 6. Hasil Augmentasi pada *Sample* Citra Biji Kakao**

### 2.8 Pelatihan Model (*Training Model*)

Pada penelitian dilakukan dua kali pelatihan menggunakan dua arsitektur yang berbeda yaitu, *Inception Resnet-V2* dan *Inception-V4*. Pada proses pelatihan *Inception Resnet-V2*, lapisan dibagi menjadi beberapa blok, termasuk blok *Steam*, blok *Inception Resnet-A*, blok *Inception Resnet-B*, blok *Inception Resnet-C*, *Reduction A*, *Reduction B*, serta lapisan *softmax*. Proses pelatihan dimulai dengan citra yang masuk ke dalam *Steam* blok, yang terdiri dari operasi konvolusi, penggabungan filter, dan *max pooling*. Selanjutnya, citra melewati berbagai blok seperti *Inception Resnet-A*, *Reduction-A*, *Inception Resnet-B*, *Reduction-B*, dan *Inception Resnet-C*. Setelah fitur diekstraksi, fitur tersebut digabungkan di lapisan *fully connected* sebelum masuk ke lapisan *softmax* untuk menentukan bobot kelas. Proses pelatihan *Inception-V4* mirip dengan *Inception Resnet-V2*, tetapi perbedaannya terletak pada blok-blok *Inception* yang digunakannya. *Inception-V4* menggunakan blok *Inception-A*, blok *Inception-B*, dan blok *Inception-C*.



## 2.9 Evaluasi Kinerja Model

Pada tahap ini kinerja sistem yang dibuat dihitung dengan mengukur kinerja klasifikasi menggunakan empat metrik, yaitu TP (*true positive*), TN (*true negative*), FP (*false positive*) dan FN (*false negative*) (Masruroh, dkk., 2023). Performa diukur dalam hal akurasi, presisi, *recall*, dan skor F-1.

*Accuracy*: Tingkat akurasi model ditetapkan berdasarkan parameter-parameter model dan diungkapkan dalam bentuk persentase. Perhitungan akurasi menggunakan Persamaan (1)

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

*Precision*: Presisi adalah perbandingan antara prediksi yang benar positif dan total hasil yang diprediksi positif. Perhitungan presisi menggunakan Persamaan (2)

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

*Recall*: *Recall* merupakan proporsi prediksi positif yang sebenarnya terhadap total data yang sebenarnya positif. Perhitungan *recall* menggunakan Persamaan (3)

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

*F-1 Score*: Skor F1 adalah gabungan tertimbang antara presisi dan *recall*. Perhitungan *F1 Score* menggunakan persamaan (4)

$$F1\ Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

Model klasifikasi dapat juga dibandingkan secara visual menggunakan grafik kurva ROC (*Receiver Operating Characteristic*). Grafik ini memiliki garis diagonal yang menggambarkan klasifikasi acak. Performa model dianggap baik jika garis ROC mendekati titik *True Positive Rate* (TPR) daripada titik *False Positive Rate* (FPR) (Hadiwandra, 2019). Kurva ROC dibuat dari perhitungan *Confusion Matrix*, yang menghubungkan *False Positive Rate* dan *True Positive Rate*. Untuk membandingkan kinerja algoritma, kita dapat membandingkan luas area di bawah kurva (*Area Under Curve* atau AUC) (Kristiawan & Widjaja, 2021). Rumus dari grafik ROC dapat menghitung *True Positive Rate* dan *False True Rate* pada Persamaan (5) dan (6).

$$TPR = \frac{TP}{TP+FN} \quad (5)$$

$$FPR = \frac{FP}{FP+TN} \quad (6)$$

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Pelatihan Model (*Training Model*)

Penelitian dilakukan dua kali pelatihan menggunakan dua arsitektur yang berbeda yaitu, *Inception Resnet-V2* dan *Inception-V4*. Model akan dibangun dengan menyesuaikan *hyperparameter* yang berbeda sebagaimana dijelaskan dalam Tabel 3.

**Tabel 3. Skenario *Hyperparameter***

No	Optimizer	Batch size	Learning rate
1	Adam	8	0,001
2		8	0,0001
3		8	0,00001
4		16	0,001
5		16	0,0001
6		16	0,00001
7	RMSprop	8	0,001
8		8	0,0001
9		8	0,00001
10		16	0,001
11		16	0,0001
12		16	0,00001

Proses *training* menggunakan *Inception Resnet-V2* dilakukan sesuai dengan *hyperparameter* yang sudah ditentukan sebelumnya pada Tabel 3. Pada setiap proses *training* pada model akan menggunakan *epoch* yang sama yaitu sebesar 100 *epoch*. Setelah dilakukan proses *training* dari setiap model, seluruh detail hasil dari proses pelatihan untuk setiap model dengan menggunakan arsitektur *Inception Resnet-V2* dapat ditemukan dalam Tabel 4.

**Tabel 4. Hasil *Training Model Inception Resnet-V2***

No	Optimizer	Batch size	Learning rate	Accuracy	Validation accuracy	Loss	Validation loss
A1	Adam	8	0,001	89,41%	75,60%	0,224912	0,578557
A2		8	0,0001	99,59%	87,00%	<b>0,017894</b>	0,649678
A3		8	0,00001	98,17%	84,55%	0,050648	0,744839
A4		16	0,001	50,92%	52,84%	0,356156	7,465338
A5		16	0,0001	99,19%	88,61%	0,037898	0,453132
A6		16	0,00001	99,19%	87,80%	0,047098	0,541869
A7	RMSprop	8	0,001	81,26%	72,36%	0,443875	0,860627
A8		8	0,0001	96,54%	90,24%	0,024268	4,574804
A9		8	0,00001	99,39%	86,99%	0,045833	0,607494
A10		16	0,001	82,69%	82,11%	0,149740	0,615930
A11		16	0,0001	<b>99,59%</b>	<b>91,87%</b>	0,019418	0,568102
A12		16	0,00001	99,39%	86,18%	0,027013	0,729571

Untuk proses *training* menggunakan *Inception-V4* juga dilakukan sesuai dengan *hyperparameter* yang sudah ditentukan sebelumnya pada Tabel 3. Pada setiap proses *training* pada model akan menggunakan *epoch* yang sama yaitu sebesar 100 *epoch*. Setelah dilakukan proses *training* dari setiap model, seluruh detail hasil dari proses pelatihan untuk setiap model yang menggunakan arsitektur *Inception-V4* tersedia dalam Tabel 5.

**Tabel 5. Hasil *Training Model Inception-V4***

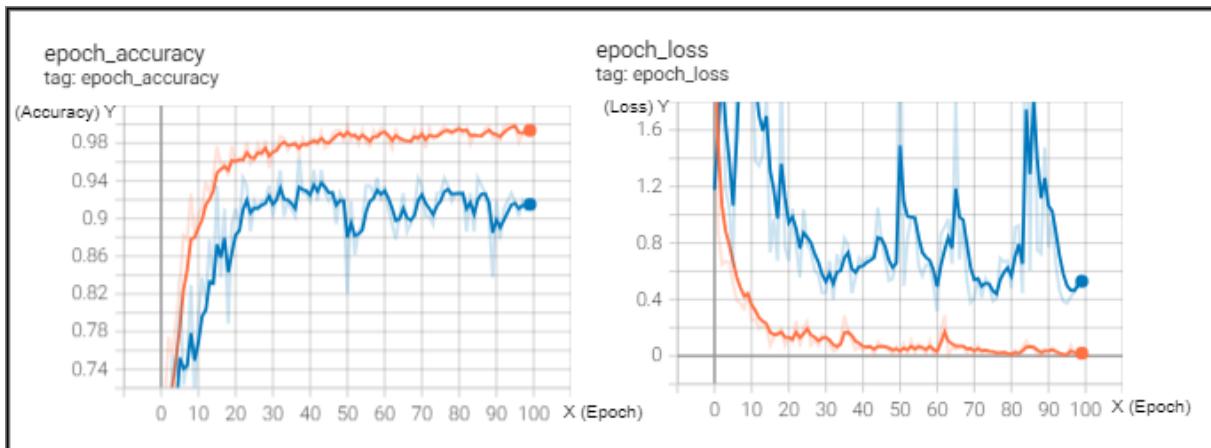
No	Optimizer	Batch size	Learning rate	Accuracy	Validation accuracy	Loss	Validation loss
B1	Adam	8	0,001	80,04%	78,86%	0,725785	0,589238
B2		8	0,0001	94,09%	86,99%	0,638714	<b>0,385920</b>
B3		8	0,00001	83,50%	78,05%	0,886655	0,546659
B4		16	0,001	73,12%	68,29%	0,499938	0,966833
B5		16	0,0001	94,30%	83,74%	0,492135	0,636562
B6		16	0,00001	84,52%	75,61%	0,754157	0,497491
B7	RMSprop	8	0,001	69,45%	61,79%	0,878345	1,139260
B8		8	0,0001	94,30%	82,11%	0,419358	0,491964
B9		8	0,00001	81,06%	74,80%	0,923248	0,724642
B10		16	0,001	49,49%	44,71%	0,752299	2,471623
B11		16	0,0001	71,69%	69,10%	0,323374	1,417822
B12		16	0,00001	81,06%	69,10%	0,855798	1,109122

Dari hasil pelatihan model yang telah dilakukan dapat dilihat pada Tabel 4 dan Tabel 5, Hasil pelatihan model menunjukkan bahwa model-model yang berbeda memberikan hasil terbaik pada parameter evaluasi pelatihan yang berbeda pula, yang mungkin disebabkan oleh penggunaan *hyperparameter* yang berbeda. *Learning rate* memengaruhi akurasi, di mana *learning rate* 0,0001 memberikan performa terbaik. *Batch size* juga berpengaruh, dengan *batch size* yang lebih kecil memberikan konvergensi yang lebih stabil tetapi konvergensi yang lebih lambat. Penggunaan *optimizer* juga mempengaruhi hasil, dengan *RMSprop* lebih cocok untuk *Inception Resnet-V2* dan *Adam* lebih cocok untuk *Inception-V4*. Hal ini disebabkan oleh mekanisme perhitungan yang berbeda antara kedua *optimizer* dan kompleksitas arsitektur masing-masing model.

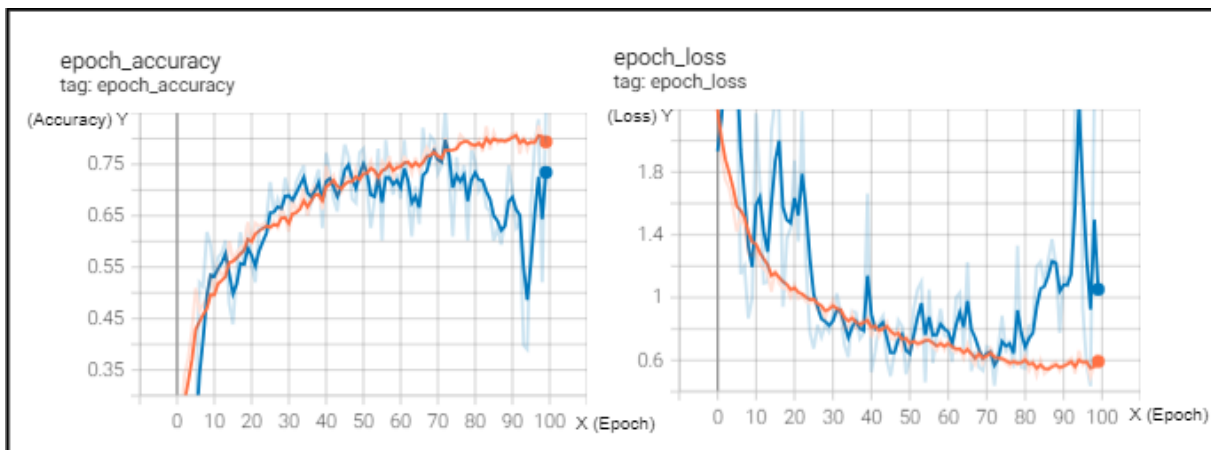
Dari 24 model tersebut, dapat dilihat bahwa hampir keseluruhan model memiliki nilai-nilai evaluasi yang baik yang membuktikan bahwa model-model tersebut dapat bekerja dengan baik dalam mengklasifikasi kualitas biji kakao. Model yang memiliki akurasi dan validasi akurasi tertinggi yaitu *Inception Resnet-V2* model A11 dengan nilai akurasi 99,59% dan validasi akurasi 91,87%. Model yang memiliki *loss* terkecil yaitu *Inception Resnet-V2* model A2 dengan nilai 0,017894 dan *loss validation* terkecil yaitu *Inception-V4* model B2 dengan nilai 0,385920. Jika dilihat semua hasil nilai terbaik berada pada *learning rate* 0,0001 namun dengan *optimizer* dan *batch size* nya berbeda, yang dimana untuk nilai *loss* dan *validation loss* terkecil berada pada *batch size* 8, hal ini mungkin terjadi dikarenakan semakin kecil *batch size* yang digunakan, maka model akan semakin baik dalam mengetahui kesalahan model dalam memprediksi klasifikasi. Namun jika dilihat secara nilai keseluruhan yaitu nilai *accuracy*, *validation accuracy*, *loss* dan *validation loss*, *Inception Resnet-V2* dengan kombinasi *optimizer RMSprop*, *batch size* 16 dan *learning rate* 0,0001 memiliki nilai performa yang paling seimbang dibandingkan dengan model yang lainnya.

Hasil evaluasi pelatihan menunjukkan bahwa model arsitektur *Inception Resnet-V2* mengungguli model *Inception-V4*. Ini disebabkan oleh kombinasi pendekatan dari *Inception* dan *Resnet* yang dimiliki oleh *Inception Resnet-V2*, yang memberikan manfaat dari kedua metode tersebut. Penggunaan *Inception* membantu mengurangi beban komputasi jaringan saraf, sementara penggunaan *Resnet* meningkatkan akurasi. Meskipun dataset yang digunakan dalam penelitian ini tergolong kecil untuk proses *deep learning*, hasil dari *training* menunjukkan bahwa *Inception Resnet-V2* dapat menghasilkan ekstraksi fitur yang lebih baik dan representasi yang lebih kuat, sehingga meningkatkan kinerja pada dataset yang kecil.

Pada Gambar 7 dan Gambar 8 dapat dilihat grafik hasil pelatihan model *Inception Resnet-V2* A11 dan model *Inception-V4* B2.



**Gambar 7. Grafik Akurasi dan Loss Model *Inception Resnet-V2* A11**



**Gambar 8. Grafik Akurasi dan Loss Model *Inception-V4* B2**

### 3.2 Hasil Pengujian Model

Setelah dilakukan proses *training* pada setiap model, selanjutnya masuk pada proses *testing* model menggunakan *datasets testing*. Berikut merupakan hasil *testing* model *Inception Resnet-V2* tercantum pada Tabel 6.

**Tabel 6 Hasil *Testing* Model *Inception Resnet-V2***

No	Optimizer	Batch size	Learning rate	Accuracy	Precision	Recall	F-1 Score
A1	Adam	8	0,001	75,61%	81,00%	76,00%	75,00%
A2		8	0,0001	86,99%	88,00%	87,00%	87,00%
A3		8	0,00001	84,55%	86,00%	85,00%	84,00%
A4		16	0,001	52,85%	59,00%	53,00%	53,00%
A5		16	0,0001	88,62%	90,00%	90,00%	88,00%
A6		16	0,00001	87,80%	89,00%	89,00%	88,00%
A7	RMSprop	8	0,001	72,36%	76,00%	75,00%	71,00%
A8		8	0,0001	90,24%	90,00%	90,00%	90,00%
A9		8	0,00001	86,99%	87,00%	87,00%	87,00%

Evaluasi Kinerja Model Inception Resnet-V2 dan Inception-V4 Dalam Mengklasifikasi Kualitas Biji Kakao

No	Optimizer	Batch size	Learning rate	Accuracy	Precision	Recall	F-1 Score
A10		16	0,001	82,11%	86,00%	80,00%	81,00%
A11		16	0,0001	<b>91,87%</b>	<b>92,00%</b>	<b>92,00%</b>	<b>92,00%</b>
A12		16	0,00001	86,18%	87,00%	87,00%	86,00%

Berikut merupakan hasil *testing* model *Inception-V4* tercantum pada Tabel 7.

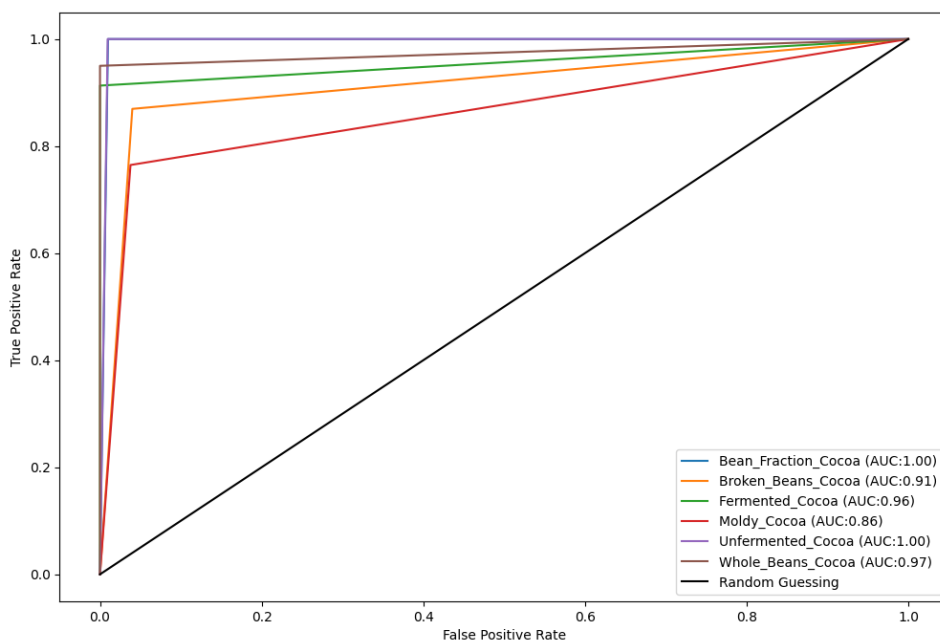
**Tabel 7. Hasil *Testing* Model *Inception-V4***

No	Optimizer	Batch size	Learning rate	Accuracy	Precision	Recall	F-1 Score
B1	<i>Adam</i>	8	0,001	78,86%	79,00%	80,00%	78,00%
B2		8	0,0001	<b>86,99%</b>	<b>87,00%</b>	<b>87,00%</b>	<b>87,00%</b>
B3		8	0,00001	78,05%	79,00%	79,00%	78,00%
B4		16	0,001	68,29%	73,00%	67,00%	66,00%
B5		16	0,0001	83,74%	84,00%	84,00%	83,00%
B6		16	0,00001	75,61%	77,00%	76,00%	76,00%
B7	<i>RMSprop</i>	8	0,001	61,79%	65,00%	62,00%	59,00%
B8		8	0,0001	82,11%	83,00%	83,00%	82,00%
B9		8	0,00001	74,80%	80,00%	74,00%	75,00%
B10		16	0,001	44,72%	60,00%	46,00%	41,00%
B11		16	0,0001	69,11%	80,00%	68,00%	66,00%
B12		16	0,00001	69,11%	72,00%	69,00%	69,00%

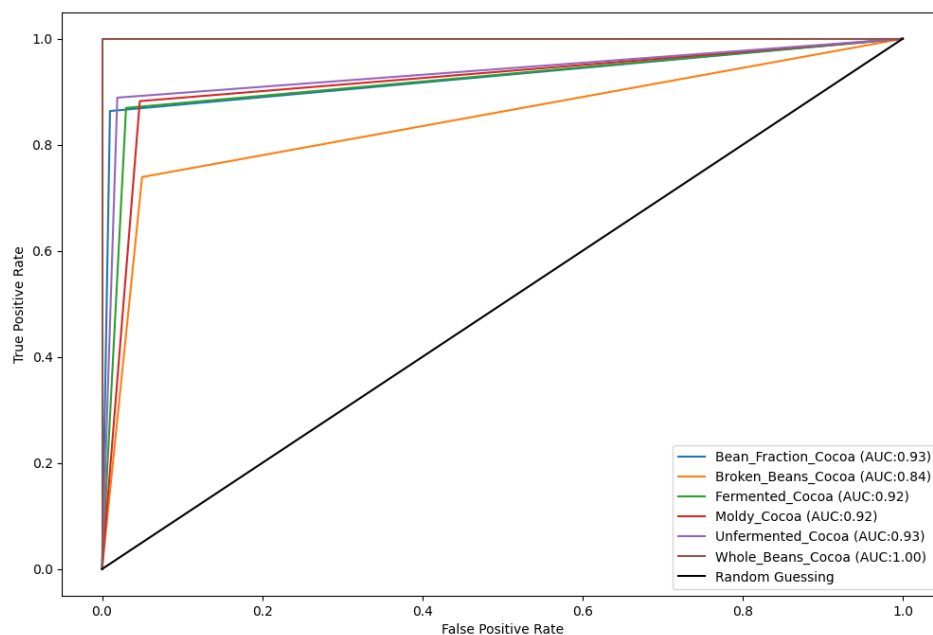
Terlihat dari Tabel 6 dan Tabel 7 bahwa *learning rate* berpengaruh terhadap keakuratan hasil, dibuktikan dengan setiap variasi model, pada *learning rate* sebesar 0,001 yang selalu disertai peningkatan akurasi pada *learning rate* 0,0001, tetapi selalu kembali menurun pada *learning rate* 0,00001, hal ini mungkin dikarenakan pemakaian *learning rate* yang semakin besar berdampak pada proses *training* yang semakin cepat tetapi nilai tingkat ketelitian jaringan yang semakin berkurang, sedangkan semakin kecil *learning rate* akan membuat model terjebak dalam minimum lokal yang dapat mencegah model mencapai hasil lebih baik, sehingga disini *learning rate* yang memiliki performa terbaik pada percobaan adalah 0,0001. *Batch size* yang digunakan dalam pelatihan model juga akan berpengaruh pada akurasi, semakin kecil *batch size* yang digunakan dapat memberikan konvergensi yang lebih stabil karena model melihat variasi data yang lebih besar dalam setiap iterasi, namun semakin kecil *batch size* dapat membuat konvergensi yang lebih lambat, dikarenakan variasi data dalam setiap iterasi lebih besar. Dapat dilihat juga pengaruh dari kedua *optimizer* yang digunakan, terlihat beberapa model *Inception Resnet-V2* menunjukkan akurasi yang lebih tinggi saat menggunakan *RMSprop optimizer*, sementara *Inception-V4* mencapai akurasi yang lebih tinggi ketika menggunakan *Adam optimizer*. Hal ini bisa dipengaruhi karena *RMSprop* memiliki mekanisme perhitungan yang dapat membantu mengatasi masalah dengan *sparse gradient* yang berguna pada arsitektur yang dalam dan kompleks seperti *Inception Resnet-V2*. Sedangkan *Adam optimizer* menggabungkan keuntungan dari *RMSprop* dan Momentum yang dapat membantu konvergensi lebih cepat dan lebih stabil pada arsitektur model seperti *Inception-V4* yang besar dan kompleks.

Dapat dilihat juga pada Tabel 6 dan Tabel 7 didapatkan hasil terbaik dari masing – masing arsitektur dengan kombinasi *hyperparameter* yang berbeda. Untuk arsitektur *Inception Resnet-V2* berada pada model A11 dengan *optimizer RMSprop*, *batch size* 16 dan *learning rate*

0,0001. Sedangkan pada arsitektur *Inception-V4* berada pada model B2 dengan *optimizer adam*, *batch size* 8 dan *learning rate* 0,0001. Jika dilihat dari hasil evaluasi pengujian model dari kedua model tersebut, didapatkan bahwa kombinasi *hyperparameter* pada model A11 *Inception Resnet-V2* memiliki nilai *accuracy*, *precision*, *recall*, serta *F1-score* yang lebih tinggi dibandingkan dengan varian model lainnya. Performa model juga dapat dinilai secara visual melalui grafik ROC. Grafik ini menampilkan TPR (*True Positive Rate*) dan FPR (*False Positive Rate*), di mana semakin dekat nilai TPR ke 1, semakin baik kinerja modelnya. Gambar 9 menunjukkan grafik ROC untuk kedua model terbaik dari setiap arsitektur. Dari grafik tersebut, terdapat 6 garis yang menunjukkan hasil performa untuk setiap kelas.



**Gambar 9. Grafik ROC *Inception Resnet-V2* Model 11**



**Gambar 10. Grafik ROC *Inception-V4* Model B2**

Berikut adalah skor ROC AUC untuk kedua model terbaik dari setiap arsitektur, terdapat dalam Tabel 8. Di sini, *Inception Resnet-V2* Model A11 menunjukkan nilai ROC AUC yang lebih tinggi dibandingkan *Inception-V4* model B2. Tabel 8 juga menunjukkan bahwa *Inception Resnet-V2* Model A11 memiliki nilai kinerja yang lebih tinggi untuk keenam kelas dibandingkan dengan *Inception-V4* model B2. Kemungkinan hal ini bisa juga disebabkan oleh hasil pelatihan *Inception Resnet-V2* Model A11 yang menunjukkan akurasi dan validasi tertinggi, yang menghasilkan kinerja yang tinggi dalam semua metrik evaluasi yang digunakan dalam penelitian ini. Oleh karena itu, dapat disimpulkan bahwa *Inception Resnet-V2* Model A11 cukup valid untuk dianggap sebagai model terbaik dalam penelitian ini.

**Tabel 8. Nilai ROC AUC**

No	Nama kelas	Model	
		<i>Inception Resnet-V2</i> model A11	<i>Inception-V4</i> model B2
1	<i>Bean Fraction Cocoa</i>	1,00	0,93
2	<i>Broken Beans Cocoa</i>	0,91	0,84
3	<i>Fermented Cocoa</i>	0,96	0,92
4	<i>Moldy Cocoa</i>	0,86	0,92
5	<i>Unfermented Cocoa</i>	1,00	0,93
6	<i>Whole Beans Cocoa</i>	0,97	1,00
<b>ROC AUC Score</b>		<b>0,950</b>	<b>0,924</b>

#### 4. KESIMPULAN

Pada penelitian ini sistem dapat mengimplementasikan arsitektur *Inception Resnet-V2* dan *Inception-V4* dalam mengklasifikasi kualitas biji kakao yang dibagi menjadi 6 kelas kualitas biji kakao. Penelitian ini dilakukan pembuatan 24 variasi model berbeda berdasarkan *hyperparameter* yang telah ditentukan sebelumnya yaitu *optimizer*, *batch size*, dan *learning rate*. Dari hasil penelitian, disimpulkan bahwa *Inception Resnet-V2* Model A11 memberikan performa terbaik dengan menggunakan *optimizer RMSprop*, *batch size* 16 dan *learning rate* 0,0001 yang menghasilkan tingkat *accuracy* sebesar 91,87%, *precision* 92,00%, *recall* 92,00%, *f1-score* 92,00% dan ROC AUC Score 0,950.

Hasil evaluasi menunjukkan bahwa model *Inception Resnet-V2* lebih unggul dibandingkan *Inception-V4* karena kombinasi pendekatan *Inception* dan *Resnet* pada *Inception Resnet-V2* memberikan manfaat dari kedua metode tersebut: *Inception* mengurangi beban komputasi, sementara *Resnet* meningkatkan akurasi. Meskipun dataset kecil, *Inception Resnet-V2* mampu menghasilkan ekstraksi fitur dan representasi yang lebih baik, meningkatkan kinerja pada dataset tersebut.

Dari seluruh penelitian yang telah dilakukan, dapat ditarik kesimpulan bahwa kualitas sebuah model dapat dipengaruhi oleh beberapa faktor, seperti penggunaan *hyperparameter* yang sesuai, kondisi data, dan jumlah dataset yang digunakan. Pengaturan *hyperparameter* yang tepat akan menjadi kunci untuk memperoleh kinerja dan keberhasilan yang optimal dalam melakukan klasifikasi prediksi. Lalu kondisi data yang baik dari segi kejernihan gambar juga dapat memberikan pengaruh, karena semakin baik kualitas data maka semakin mudah model dalam mengklasifikasikan data, kemudian semakin besar jumlah data yang digunakan juga dapat mempengaruhi akurasi model, sehingga disarankan untuk penelitian selanjutnya dapat menambahkan jumlah *datasets* dan meningkatkan kualitas *datasets* yang digunakan.

## DAFTAR RUJUKAN

- Abed, M. H., Al-Rammahi, A. H. I., & Radif, M. J. (2019). Real-Time Color Image Classification Based On Deep Learning Network. *Journal of Southwest Jiaotong University*, 54(5). <https://doi.org/10.35741/issn.0258-2724.54.5.23>
- Akbar, S. N. A. F., Hendra, & Supri Bin Hj. Amir. (2020). *Perbandingan Kinerja Arsitektur Inception-V4 Dan Resnet-50 Dalam Mengklasifikasikan Citra Paru-Paru Terinfeksi Covid-19*. 2.
- Al Husaini, M. A. S., Habaebi, M. H., Gunawan, T. S., Islam, M. R., Elsheikh, E. A. A., & Suliman, F. M. (2022). Thermal-based early breast cancer detection using inception V3, inception V4 and modified inception MV4. *Neural Computing and Applications*, 34(1), 333–348. <https://doi.org/10.1007/s00521-021-06372-1>
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. In *Journal of Big Data* (Vol. 8, Issue 1). Springer International Publishing. <https://doi.org/10.1186/s40537-021-00444-8>
- Arrofiqoh, E. N., & Harintaka, H. (2018). Implementasi Metode Convolutional Neural Network Untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi. *Geomatika*, 24(2), 61. <https://doi.org/10.24895/jig.2018.24-2.810>
- Hadiwandura, T. Y. (2019). Perbandingan Kinerja Model Klasifikasi Decision Tree , Bayesian Classifier, Instance Base, Linear Function Base, Rule Base pada 4 Dataset Berbeda. *Sains Dan Teknologi InformasI*, 5(1).
- Kadhim, M. A., & Abed, M. H. (2020). Convolutional neural network for satellite image classification. In *Studies in Computational Intelligence* (Vol. 830, Issue January). Springer International Publishing. [https://doi.org/10.1007/978-3-030-14132-5\\_13](https://doi.org/10.1007/978-3-030-14132-5_13)
- Kristiawan, K., & Widjaja, A. (2021). Perbandingan Algoritma Machine Learning dalam Menilai Sebuah Lokasi Toko Ritel. *Jurnal Teknik Informatika Dan Sistem Informasi*, 7(1), 35–46. <https://doi.org/10.28932/jutisi.v7i1.3182>
- Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., & Pietikäinen, M. (2020). Deep Learning for Generic Object Detection: A Survey. *International Journal of Computer Vision*, 128(2), 261–318. <https://doi.org/10.1007/s11263-019-01247-4>
- Masruroh, F., Surarso, B., Warsito, B., & Korespondensi, P. (2023). *Perbandingan Kinerja Inception-Resnetv2, Xception, Inception-V3, Dan Resnet50 Pada Gambar Bentuk Wajah Performance Comparison of Inception-Resnetv2, Xception, Inception-V3, and Resnet50*



- on Face Shape Images. 10(1), 11–20. <https://doi.org/10.25126/jtiik.2023104941>*
- Nugraha, P., Komarudin, A., Ramadhan, E., & Learning, D. (2022). *Deteksi Objek Dan Jenis Burung Menggunakan Convolutional Neural Network Dengan Arsitektur Inception Resnet-V2. 8, 47–55.*
- Prashanth, D. S., Mehta, R. V. K., & Sharma, N. (2020). Classification of Handwritten Devanagari Number - An analysis of Pattern Recognition Tool using Neural Network and CNN. *Procedia Computer Science, 167(2019), 2445–2457.* <https://doi.org/10.1016/j.procs.2020.03.297>
- Rochmawanti, O., Utamingrum, F., & Bachtiar, F. A. (2021). Analisis Performa Pre-Trained Model Convolutional Neural Network dalam Mendeteksi Penyakit Tuberkulosis. *Jurnal Teknologi Informasi Dan Ilmu Komputer, 8(4), 805.* <https://doi.org/10.25126/jtiik.2021844441>
- Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2016). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *Pattern Recognition Letters, 42(1), 11–24.* <http://arxiv.org/abs/1512.00567>
- Wahid, M. I., Mustamin, S. A., & Lawi, A. (2021). Identifikasi Dan Klasifikasi Citra Penyakit Daun Tomat Menggunakan Arsitektur Inception V4. *Konferensi Nasional Ilmu Komputer (KONIK), 2019, 257–264.*
- Winarto, E. G., Lawi, A., Studi, P., Informasi, S., Hasanuddin, U., Studi, P., & Informasi, S. (2021). *Implementasi Arsitektur Inception Resnet-V2 untuk Klasifikasi Kualitas Biji Kakao. 132–137.*
- Windiawan, R., Suharso, A., & Artikel, S. (2021). Identifikasi Penyakit pada Daun Kopi Menggunakan Metode Deep Learning VGG16. *Jurnal Keilmuan Dan Aplikasi Teknik Informatika, 13(2), 9–16.* <https://doi.org/10.35891/explorit>