

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Tinjauan Pustaka memuat uraian sistematis tentang hasil – hasil penelitian yang didapat oleh peneliti yang di dapat dari penelitian terdahulu yang dijadikan acuan dalam penelitian. Berikut beberapa acuan dari penelitian terdahulu:

1. Pada penelitian sebelumnya yang berjudul “Aplikasi Pendeteksi Penyakit tanaman Apel Dengan Metode *Convolutional Neural Network*” dalam penelitian ini membangun aplikasi untuk mengidentifikasi penyakit tanaman apel dengan teknik *Deep Learning* dengan metode *Convolutional Neural Network*”. Pada penelitian ini menggunakan dataset daun tanaman apel sebanyak 3151 citra daun yang telah diklasifikasikan berdasarkan kelasnya masing-masing. Penelitian ini menghasilkan akurasi model sebesar 99,4% dan akurasi validasi sebesar 97,8% (Wicaksono, Andryana, & Benrahman, 2020).
2. Pada penelitian sebelumnya yang berjudul “ *Classification of Rice Plant Diseases Using the Convolutional Neural Network Method*”, Pada penelitian ini membangun sistem klasifikasi penyakit tanaman padi menggunakan metode *Convolutional Neural Network* dengan arsitektur VGG19, yang memiliki fitur untuk meningkatkan hasil. Citra yang digunakan sebagai data latih dan uji terdiri dari 105 citra yang terbagi menjadi citra latih dan citra uji. Pengujian parameter menggunakan variasi

epoch dan augmentasi data. Hasil penelitian diperoleh akurasi uji sebesar 95,24% (Priyanka, 2021).

3. Muhammad Hammad Saleem pada penelitiannya yang berjudul “*Plant Disease Detection and Classification by Deep Learning*”, Pada penelitian ini melakukan pendeteksian penyakit tanaman dengan *Deep Learning*. Dalam penelitian ini menggunakan dataset daun tanaman tomat dan menghasilkan akurasi sebesar 97,62%. (Saleem, Potgieter, & Arif, 2019).
4. pada penelitian sebelumnya yang berjudul “Deteksi Keaslian Mata Uang Rupiah Berbasis Android Menggunakan Algoritma *Convolutional Neural Network* Dengan *Tensorflow*”. Pada penelitian ini peneliti menggunakan algoritma *Convolutional Neural Network* (CNN). Proses *training* yang dilakukan menggunakan 2000 step dengan jumlah 24 *batch* pada saat *training* dengan nilai rata-rata *Loss function* dengan rata-rata 1.1124 diselesaikan dengan rata-rata waktu 1.6 detik perlangkah. Proses percobaan output dilakukan sebanyak 3 kali dengan kategori uang asli dan uang palsu dengan pencahayaan terang dan redup, dengan nilai rata-rata dengan uang asli terang ialah 0.9967, uang asli redup ialah 0.7833, uang palsu terang ialah 0.9933 dan uang palsu redup ialah 0.82 (Pamungkas, 2020).
5. pada penelitian sebelumnya yang berjudul “Deteksi dan Pengenalan Ikan Menggunakan Algoritma *Convolutional Neural Network*”. Pada penelitian ini dilakukan 6 kali percobaan training untuk ditemukn nilai palin baik, dan mendapatkan nilai *test score* 2.475, *test accuracy* 0.4237. data yang yang digunakan pada penelitian ini berupa tangkapan gambar dari hasil video

secara langsung / *realtime* menggunakan *webcam*. Penelitian ini menghasilkan tingkat akurasi sebesar 85,18% dengan pengujian 27 kali yang diman 4 kali tidak dapat mengidentifikasi foto dan 23 kali berhasil dalam mengidentifikasi foto ikan. (Prasmatio, 2020)

Tabel 2. 1 : Tabel Perbandingan

No	Penulis	Objek	Metode	Hasil / Keluaran Penelitian
1.	Guntur Wicaksono (2020)	Pendeteksi penyakit pada tanaman apel.	CNN (<i>Convolutional Neural Network</i>)	Prototype aplikasi android pendeteksi penyakit tanaman apel dengan akurasi model sebesar 99,4% dan akurasi validasi sebesar 97,8%.
2.	A A JE Veggy Priyanka (2021)	Klasifikasi penyakit pada tanama padi	CNN (<i>Convolutional Neural Network</i>)	Pengklasifikasian jenis – jenis penyakit pada tanman padi dengan tingkat akurasi 95,24%.
3.	Muhammad Hammad Saleem (2019)	Deteksi dan klasifikasi penyakit	<i>Deep Learning</i> (DL)	Pendekatan <i>Deep Learning</i> untuk pendeteksian penyakit tanaman dengan akurasi 97,62%
4.	Nur hidayat Pamungkas (2020)	Deteksi Keaslian mata uang rupiah	CNN (<i>Convolutional Neural Network</i>)	Sistem pendeteksi keaslian mata uang rupiah menggunakan Algoritma CNN (<i>Convolutional Neural Network</i>) dengan <i>Tensorflow</i> .
5.	R. Mehindra Prasmatio (2020)	Deteksi dan Pengenalan Ikan	Algoritma <i>Convolutional Neural Network</i>	Pendeteksian foto ikan dengan tingkat akurasi sebesar 85,18%.

2.2 Landasan Teori

2.2.1 Citra Digital

Citra merupakan representasi objek dua dimensi dari dunia visual, menyangkut berbagai macam disiplin ilmu yang mencakup seni, human vision, astronomi, Teknik dan sebagainya merupakan suatu Kumpulan piksel-piksel atau titik-titik yang berwarna yang berbentuk dua dimensi (Hutahaean, Waluyo, & Rais, 2019).

Citra Digital merupakan citra $f(x,y)$ dimana dilakukan diskritisasi koordinat spasial/ sampling dan diskritisasi tingkat kwantisasi (kecemerlangan/ keabuannya). Citra digital adalah suatu fungsi intensitas Cahaya $f(x,y)$, dimana nilai x dan nilai y adalah koordinat spasial. Nilai fungsi tersebut di setiap titik (x,y) adalah tingkat kecemerlangan citra di titik tersebut.

Citra digital adalah sebuah matriks dimana indeks baris maupun kolomnya menyatakan sebuah titik pada citra tersebut dan elemen matriksnya (yang dikenal sebagai elemen gambar/ *picture element*/ piksel/ *pixel*/ *pels*) menyatakan tingkat keabuan di titik tersebut. Sehingga, informasi yang terkandung didalamnya bersifat diskret. Citra digital tidaklah selalu merupakan hasil rekaman data yang sifatnya kontinyu seperti gambar pada monitor tv, foto pada sinar-x dan lain-lain. Dengan begitu untuk memperoleh suatu citra digital dibutuhkan sebuah proses konversi, sehingga selanjutnya citra tersebut bis diproses (Hutahaean, Waluyo, & Rais, 2019).

2.2.2 Penyakit Tanaman Tomat

Tanaman tomat merupakan salah satu tanaman hortikultura yang penting di Indonesia. Namun, budidaya tanaman tomat banyak mengalami masalah yang dapat menyebabkan produksi tanaman menjadi rendah baik secara kuantitas maupun kualitas.

Tanaman pada umumnya dapat dikatakan terserang penyakit apabila pertumbuhannya menyimpang dari keadaan normal. Penyebabnya bisa dari beberapa macam, diantaranya dapat disebabkan oleh jamur, bakteri, dan virus. Beberapa jenis penyakit pada tanaman diantaranya penyakit yellow leaf curl, potato virus y, tomato mozaik virus, dan bercak daun (Sigitta, 2023).

a. Bacterial Spot

Bacterial spot disebabkan oleh bakteri *Xanthomonas* yang tersebar luas dan merupakan penyakit bakteri yang merusak pada tanaman tomat. Infeksi biasanya menghasilkan lesi dan defoliasi pada daun dan buah, ini dapat menurunkan kualitas buah. Inveksi *bacterial spot* banyak terjadi di daerah yang hangat dan lembab dalam satu musim. Sayangnya, ketika suatu lahan pertanian terkena penyakit ini ini, akan sulit sekali untuk dilakukan pengendalian penyaki (Astianingrum, Arhandi, & Ariditya, 2020).



Gambar 2. 1 Tomato Bacterial Spot

(Sumber Gambar :

<https://www.kaggle.com/code/ammarnassanalhajali/tomato-leaf-disease-detection-0-998-inference>)

b. Early Blight (Pembusukan Dini)

Early blight (EB) atau pembusukan daun dini disebabkan oleh jamur *Alternaria Solani*. EB ditandai dengan cincin konsentris berwarna coklat hingga hitam seperti bintik-bintik daun, yang menyatu secara lateral yang mengakibatkan busuk daun, defoliiasi dan membuat gugur buah-buahan yang belum matang. Penyakit ini juga ditandai dengan adanya lesi yang cekung dan bintik bintik hitam di dekat pangkal batang yang mengakibatkan pengerdilan pada batang (Shoaib, Awan, & Khan, 2019).



Gambar 2. 2 Tomato Early Blight

(Sumber Gambar :

<https://www.kaggle.com/code/ammarnassanahajali/tomato-leaf-disease-detection-0-998-inference>)

c. Late Blight (Busuk Daun)

Penyakit busuk daun pada tanaman tomat disebabkan oleh ‘jamur’ *Phytophthora Infestans* yang mana penyakit ini akan sangat merusak bagi tanaman tomat yang ditanam di lingkungan lembab dan dapat menyebabkan gagal panen total hanya dalam beberapa hari (Lage, Marouelli, & Cafe-Filho, 2019). Semua bagian tanaman bisa terpengaruh oleh penyakit ini dan juga bisa mengakibatkan pembusukan pada buah. Penyakit busuk daun dapat menginfeksi daun muda (atas) atau tua (bawah). Penyakit ini ditandai dengan munculnya sebagai bercak air hijau pucat mulai dari ujung daun yang membesar dengan cepat, membentuk bercak hitam kehijauan yang tidak beraturan. Ini dapat berkembang dengan cepat ketika daun basah atau kelembaban udara tinggi (Astianingrum, Arhandi, & Ariditya, 2020).



Gambar 2. 3 Tomato Late Blight

(Sumber Gambar :

<https://www.kaggle.com/code/ammarnassanahajali/tomato-leaf-disease-detection-0-998-inference>)

d. Leaf Mold (Bercak Daun)

Bercak daun disebabkan oleh jamur *Cladosporium fulfum*, merupakan salah satu penyakit daun tomat yang paling merusak tanaman dan tumbuh dalam kondisi lingkungan yang lembab. Penyakit ini merupakan masalah yang umum terjadi pada tomat yang dibudidayakan di rumah kaca, namun kadang dapat muncul pada tomat yang ditanam di ladang dengan lingkungan lembab (Astianingrum, Arhandi, & Ariditya, 2020).



Gambar 2. 4 *Tomato Leaf Mold*

(Sumber Gambar :

<https://www.kaggle.com/code/ammarnassanalhajali/tomato-leaf-disease-detection-0-998-inference>)

e. *Septoria Leaf Spot*

Bercak daun *Septoria* disebabkan oleh jamur *Septoria lycopersici*. Ini adalah salah satu penyakit paling merusak pada dedaunan tomat dan sangat parah di daerah yang cuacanya basah dan lembab dalam jangka waktu lama. Bercak daun *Septoria* biasanya muncul di daun bawah setelah buah pertama terbentuk. Bintik-bintiknya berbentuk lingkaran, diameternya sekitar 1/16 hingga ¼ inci dengan pinggiran berwarna coklat keabu-abuan dengan struktur buah kecil berwarna hitam.

Ciri khas dari penyakit ini yaitu terdapat banyak bintik pada banyak bintik pada setiap daun. Penyakit ini menyebar dari pertumbuhan tertua hingga tertua. Jika lesi daun banyak, daun menjadi agak kuning, kemudian coklat, dan kemudian layu. Infeksi buah jarang terjadi (Astianingrum, Arhandi, & Ariditya, 2020).



Gambar 2. 5 Tomato Septoria Leaf Spot

(Sumber Gambar :

<https://www.kaggle.com/code/ammarnassanalhajali/tomato-leaf-disease-detection-0-998-inference>)

f. Spider mites two Spotted Spider mete

Spider Mites Two Spider Mite atau tungau laba laba berbintik dua adalah hama pertanian yang menyerang lebih dari 140 famili tanaman yang berbeda dan 1.100 spesies tanaman termasuk tomat. Tungau laba-laba ini berkembang dengan baik ketika kondisi hangat dan kering. Tungau laba-laba menyerang tanaman tomat dengan memasukkan *stylet* mereka ke dalam jaringan daun dan menghilangkan isi sel. Hilangnya klorofil daun berpengaruh terhadap fotosintesis yang dilakukan daun, ini menyebabkan perubahan warna daun, sering disebut bronzing juga mengakibatkan

penurunan kesehatan secara keseluruhan atau bahkan mengakibatkan kematian (Rakha, 2017).



Gambar 2. 6 Tomato Spider Mites Two Spider Mite

(Sumber Gambar :

<https://www.kaggle.com/code/ammarnassanalhajali/tomato-leaf-disease-detection-0-998-inference>)

g. Yellow Leaf Curl Virus

Tomato yellow leaf curl virus merupakan penyakit tomat yang disebabkan sebuah virus. Tanaman tomat yang terinfeksi akan sulit untuk tumbuh atau mengerdil. Daun dari tanaman yang terinfeksi berukuran kecil dan melengkung keatas serta menunjukkan warna kekuningan. Bahkan dalam beberapa kasus, tanaman terinfeksi yang sejak dini pertumbuhan tanaman menyerupai bonsai karena tidak tumbuh namun berdaun agak lebat. Meskipun berdaun lebat virus ini mengakibatkan gugurnya setiap bunga yang tumbuh. Dampak ini berakibat turunnya hasil panen (Astianingrum, Arhandi, & Ariditya, 2020).



Gambar 2. 7 Tomato Yellow Leaf Curl Virus
(Sumber Gambar :
<https://www.kaggle.com/code/ammarnassanalhajali/tomato-leaf-disease-detection-0-998-inference>)

h. Mosaic Virus

Mosaic virus merupakan salah satu virus yang menyerang tomat, bahkan bukan hanya tomat tapi juga lada, kentang, apel, pir, ceri, dan lainnya. virus ini memiliki gejala berupa bercak hijau muda dan tua pada daun, daun mungkin melengkung, berbentuk aneh dan ukurannya mengecil. Jika tanaman terinfeksi lebih awal, tanaman akan nampak lebih kecil dan menguning. akibat kondisi yang memburuk juga berdampak pada buah yang dihasilkan yakni biasanya terjadi matang yang tidak rata, bahkan jumlah buah akan berkurang begitu juga dengan ukurannya (Astianingrum, Arhandi, & Ariditya, 2020).



Gambar 2. 8 Tomato Mosaic Virus

(Sumber Gambar :
<https://www.kaggle.com/code/ammarnassanalhajali/tomato-leaf-disease-detection-0-998-inference>)

2.2.3 Deep Learning

Deep Learning merupakan sebuah model yang dapat mempelajari metode komputasinya sendiri menggunakan "otaknya" sendiri. Algoritma *Deep Learning* berusaha belajar dalam berbagai level sesuai dengan tingkat abstraksinya.

Algoritma *Deep Learning* menggunakan metadata sebagai input dan mengolahnya menggunakan sejumlah lapisan tersembunyi (*hidden layer*) transformasi *non linier* dari data masukan untuk menghitung nilai output. Algoritma pada *Deep Learning* memiliki fitur yang unik yaitu sebuah fitur yang mampu mengekstraksi secara otomatis. Hal ini berarti algoritma yang dimilikinya secara otomatis dapat menangkap fitur yang relevan sebagai keperluan dalam pemecahan suatu masalah (Miftah, 2021).

Deep learning memiliki kemampuan yang sangat baik dalam visi komputer dengan kapabilitasnya yang signifikan dalam memodelkan berbagai data kompleks seperti data gambar. Salah satu metode *Deep Learning* yang memiliki hasil paling bagus dalam pengenalan citra adalah *Convolutional Neural Network* (CNN). Hal ini disebabkan karena metode CNN berusaha meniru sistem pengenalan citra pada *visual cortex* manusia sehingga memiliki kemampuan mengolah informasi citra. CNN dirancang khusus untuk *Deep Learning* yaitu pada bidang *image processing* (Hasanah, 2019).

2.2.4 Convolutional Neural Network (CNN)

Convolutional Neural Network merupakan salah satu jenis algoritma *Deep Learning* yang dapat menerima input berupa gambar, menentukan aspek atau obyek apa saja dalam sebuah gambar yang bisa digunakan mesin untuk mempelajari dan mengenali gambar, dan membedakan antara satu gambar dengan yang lainnya (Nafi, Hakim, Nugroho, & Minarno, 2022).

CNN adalah versi resmi dari *multilayer perceptrons* biasanya berarti jaringan yang terhubung penuh, yaitu setiap neuron dalam satu lapisan terhubung ke semua neuron di lapisan berikutnya. “Koneksi Penuh” dari jaringan – jaringan ini membuat mereka rentan terhadap *overfitting* data (Azmi, 2019).

CNN menggunakan pra-pemrosesan yang relative sedikit dibandingkan dengan algoritma klasifikasi gambar lainnya. Arsitektur CNN terbilang mirip dengan pola koneksi neuron atau sel saraf dalam otak manusia. CNN tercipta karena terinspirasi dengan *Visual Cortex*, yaitu bagian pada otak yang bertugas untuk memproses informasi dalam bentuk *visual*. Dengan arsitektur seperti itu, CNN dapat dilatih untuk memahami detail sebuah gambar dengan lebih baik. Dengan begitu, CNN dapat menangkap dependensi Spasial dan Temporal dalam sebuah gambar setelah diberi filter yang relevan (Hasanah, 2019).

Convolutional Neural Network adalah jenis jaringan syaraf yang digunakan dalam pengolahan citra dan tugas pengenalan pola visual. CNN terdiri dari beberapa lapisan yang berbeda, yang bekerja bersama untuk mengekstraksi fitur dari gambar. Berikut adalah konsep dari semua lapisan CNN:

1. *Input layer* (Lapisan Input)

Lapisan ini merupakan lapisan pertama di CNN yang menerima citra sebagai input. Setiap piksel gambar direpresentasikan sebagai node di lapisan ini.

2. *Convolution layer* (Lapisan Konvolusi)

Lapisan ini terdiri dari beberapa filter konvolusi yang bergerak melintasi gambar input untuk mengekstraksi fitur-fitur seperti tepi, sudut, dan tekstur.

3. *Activation layer* (Lapisan Aktivasi)

Setelah konvolusi, fungsi aktivasi seperti ReLU (*Rectified Linear Unit*) diterapkan untuk memperkenalkan non linearitas ke dalam jaringan. Ini membantu jaringan CNN untuk mempelajari representasi yang lebih kompleks.

4. *Pooling layer* (Lapisan Pooling)

Lapisan ini digunakan untuk mereduksi resolusi citra dan mengurangi jumlah parameter dalam jaringan. Max-pooling adalah salah satu teknik umum yang digunakan di sana.

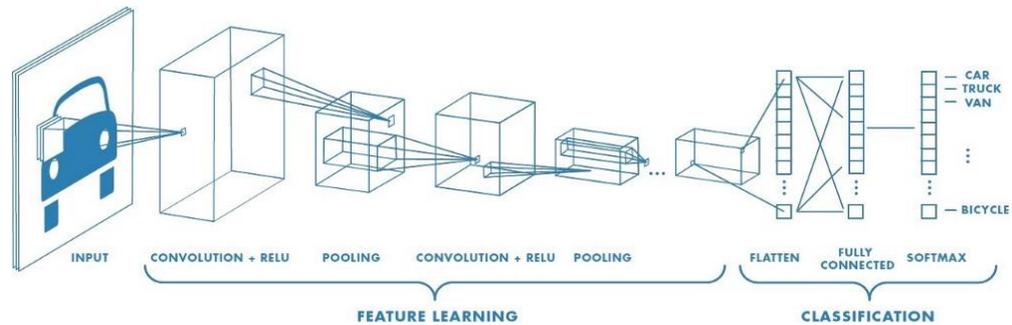
5. *Fully connected layer* (Lapisan Terhubung Penuh)

Fully connected layer adalah lapisan yang mirip dengan lapisan dalam jaringan saraf biasa (*feedforward neural network*). Ini digunakan untuk menghubungkan hasil dari lapisan sebelumnya ke unit-unit output yang sesuai dengan kelas-kelas yang diidentifikasi.

6. *Output layer* (lapisan Output)

Lapisan ini menghasilkan hasil klasifikasi atau regresi, tergantung pada jenis tugas yang dijalankan oleh CNN. Misalnya, pada tugas klasifikasi gambar, ini dapat menghasilkan probabilitas kelas yang berbeda

Pada gambar di bawah ini bisa dilihat alur dari proses CNN dalam mengolah citra masukan sampai mengklasifikasikan citra tersebut ke kategori tertentu berdasarkan nilai keluarannya.



Gambar 2. 9 Arsitektur CNN

(Sumber : <https://old.mikroskil.ac.id/ejurnal/index.php/jsm/article/view/670/330>)

Arsitektur dari *Convolutional Neural Network* dibagi menjadi 2 bagian yaitu *Feature Extraction Layers (Feature Learning)* dan *Fully Connected Layer (MLP)* atau *Classification*. Dikatakan *Feature Extraction Layers*, karena proses yang terjadi pada bagian ini adalah melakukan *encoding* dari sebuah gambar menjadi features yang berupa angka – angka yang meresentasikan gambar tersebut. *Fully Connected Layer* atau *Classification* yang di mana “*fully connected*” artinya setiap *neuron* pada lapisan sebelumnya terhubung dengan setiap *neuron* pada lapisan berikutnya. Tujuan dari lapisan *Fully Connected* adalah menggunakan fitur-fitur ini untuk mengklasifikasikan gambar *input* ke dalam berbagai kelas berdasarkan *dataset* pelatihan. Setiap *neuron* akan diberikan fungsi dan bobot tertentu sehingga dapat memberikan angka probabilitas bahwa suatu gambar mengandung objek tertentu. (Nur Arkhamia Batubara, 2020)

2.2.5 Python

Python merupakan Bahasa pemrograman yang dapat melakukan eksekusi sejumlah instruksi multiguna secara langsung (interpretative) dengan berorientasi pada objek serta menggunakan semantic dinamis untuk memberikan tingkat keterbacaan kode atau *syntax*. Sebagian besar mengartikan *python* sebagai bahasa dengan tingkat kemampuan tinggi, menggabungkan kapabilitas, dan sintaks kode yang sangat jelas dan dilengkapi oleh fungsionalitas dari Pustaka dasar yang sangat besar dan komperehensif. Walaupun *python* ini digolongkan sebagai Bahasa pemrograman tingkat tinggi, namun tetap python dirancang sedemikian rupa supaya mudah dipahami dan dipelajari.

Python juga dapat berjalan dibanyaak platform seperti *Mac*, *Linux*, dan *Windows* dll. *Python* bersifat *open source*, sehingga masih banyak orang yang berkontribusi untuk mengembangkan dimana yang hak dan kekayaan intelektual dipegang oleh PSF. Bahasa *Python* didukung oleh *library* yang didalamnya menyediakan fungsi analisis data dan dan fungsi *machine learning*, *data processing tools*, serta visualisasi data. Hal ini membuat python menjadi Bahasa pemrograman yang populer pada bidang *data science* dan analisis (Saputra, 2020).

2.2.6 Flask

Flask adalah sebuah *web framework* yang ditulis dengan bahasa *python* dan tergolong sebagai jenis *microframework*. Flask berfungsi sebagai kerangka kerja aplikasi dan tampilan suatu web. Dengan menggunakan Flask dan bahasa Python, pengembang dapat membuat sebuah web yang terstruktur dan dapat mengatur *behaviour* suatu web dengan lebih mudah. Flask termasuk pada jenis

microframework karena tidak memerlukan suatu alat atau pustaka tertentu dalam penggunaannya.

Sebagian besar fungsi dan komponen umum seperti *validasi form*, *database*, dan sebagainya tidak terpasang secara *default* di Flask. Hal ini dikarenakan fungsi dan komponen – komponen tersebut sudah disiapkan oleh pihak ketiga dan Flask dapat menggunakan ekstensi yang membuat fitur dan komponen – komponen tersebut seakan diimplementasikan oleh Flask sendiri. Selain itu, meskipun Flask disebut sebagai *microframework*, bukan berarti Flask mempunyai kekurangan dalam hal fungsionalitas. *Microframework* disini berarti bahwa Flask bermaksud untuk membuat *core* dari aplikasi ini sesederhan mungkin tapi tetap dapat dengan mudah ditambahkan. Dengan begitu, fleksibilitas serta skalabilitas dari Flask dapat dikatakan cukup tinggi dibandingkan dengan *framework* lainnya (Maulid, 2021).

2.2.7 Google Colaboratory

Colaboratory, atau “*collab*” merupakan produk dari *Google Research*. *Collab* memungkinkan siapa saja menulis dan mengeksekusi kode python arbitrer melalui *browser*, dan sangat cocok untuk *machine learning*, analisis data, serta pendidikan. Secara lebih teknis, *Collab* merupakan layanan notebook *Jupyter* yang dihosting dan dapat digunakan tanpa penyimpanan, serta menyediakan akses gratis ke *resource* komputasi termasuk GPU. *Resource Collab* tidak dijamin dan sifatnya terbatas, serta batas penggunaannya terkadang berfluktuasi. Hal ini diperlukan agar *Collab* dapat menyediakan *resource* secara gratis. Pengguna yang ingin memiliki akses lebih andal ke *resource* yang lebih baik dapat menggunakan *Collab Pro*. Memperkenalkan *Collab Pro* merupakan langkah

pertama yang Google ambil untuk melayani pengguna yang ingin melakukan lebih banyak hal di Collab. Tujuan jangka panjang pihak Google adalah untuk terus menyediakan versi gratis Collab, dan di saat yang bersamaan berkembang secara berkelanjutan untuk memenuhi kebutuhan pengguna Google (Soen, Marlina, & Renny, 2022).

2.2.8 Matriks Konfusi

Matriks konfusi (*confusion Matrix*) adalah alat evaluasi yang digunakan dalam statistika dan pembelajaran mesin untuk mengukur kinerja suatu model klasifikasi. Matriks ini menyajikan ringkasan hasil prediksi model pada suatu dataset, membaginya menjadi empat kategori berdasarkan kombinasi prediksi yang benar atau salah terkait dengan kelas sebenarnya. Matriks konfusi terdiri dari empat sel utama, masing-masing mewakili satu dari empat kondisi berikut:

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 2.10 Matriks Konfusi

(Sumber : <https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62>)

- True Positive (TP);** Jumlah sampel yang benar-benar positif dan juga diprediksi sebagai positif oleh model.
- True Negative (TN);** Jumlah sampel yang benar-benar negatif, dan juga diprediksi sebagai negatif oleh model.

- c. **False Positive (FP)**; Jumlah sampel yang benar-benar negatif, tetapi diprediksi sebagai positif oleh model.
- d. **False Negative (FN)**; Jumlah sampel yang benar-benar positif, tetapi diprediksi sebagai negatif oleh model.

Matriks konfusi memungkinkan kita untuk mengevaluasi berbagai aspek performa model, seperti akurasi, presisi, recall, dan F1-score, dengan menggunakan rumus-rumus berikut:

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

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

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

$$F1-score = \frac{2(Presisi)(Recall)}{presisi+Recall}$$

Penggunaan matriks kofusi dan matrik evaluasi ini membantu dalam memahami seberapa baik model CNN dapat mengklasifikasikan data dan memberikan wawasan yang berguna untuk perbaikan dan peningkatan model (KantinIT, 2023).