**PENERAPAN METODE *CONVOLUTIONAL NEURAL NETWORK* (CNN) DALAM APLIKASI PENDETEKSI PENYAKIT DAUN TANAMAN KENTANG**

**BERBASIS ANDROID**

**PROPOSAL SKRIPSI**

**Diajukan Sebagai Salah Satu Syarat Penyusunan Skripsi**

**Pada Program Studi Teknik Informatika**

**Oleh:**

**SONNYA GHANDI**

**NIM: 201351131**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI TEKNOLOGI WASTUKANCANA**

**PURWAKARTA**

**2024**

DAFTAR ISI

[DAFTAR ISI i](#_Toc164884966)

[BAB I PENDAHULUAN 1](#_Toc164884967)

[1.1 Latar Belakang Masalah 1](#_Toc164884968)

[1.2 Rumusan Masalah 3](#_Toc164884969)

[1.3 Batasan Masalah 3](#_Toc164884970)

[1.4 Tujuan Penelitian 3](#_Toc164884971)

[1.5 Manfaat Penelitian 4](#_Toc164884972)

[1.5.1 Bagi Pengguna 4](#_Toc164884973)

[1.5.2 Bagi Peneliti 4](#_Toc164884974)

[1.6 Sistematika Penulisan 4](#_Toc164884975)

[BAB II TINJAUAN PUSTAKA 6](#_Toc164884976)

[2.1 Penerapan 6](#_Toc164884977)

[2.2 *Cross Industry Standard Process for Data Mining* (CRISP-DM) 6](#_Toc164884978)

[2.3 Sistem 8](#_Toc164884979)

[2.4 Kentang 8](#_Toc164884980)

[2.4.1 Penyakit Daun Tanaman Kentang 9](#_Toc164884981)

[2.5 *Machine Learning* 11](#_Toc164884982)

[2.6 *Deep Learning* 11](#_Toc164884983)

[2.7 *Convolutional Neural Network* 12](#_Toc164884984)

[2.7.1 *Convolution Layer* 12](#_Toc164884985)

[2.7.2 *Pooling Layer* 13](#_Toc164884986)

[2.7.3 *Fully Connected Layer* 13](#_Toc164884987)

[2.8 TensorFlow 14](#_Toc164884988)

[2.9 *Keras* 14](#_Toc164884989)

[2.10 Android 15](#_Toc164884990)

[2.11 React Native 15](#_Toc164884991)

[2.12 Penelitian Terdahulu 16](#_Toc164884992)

[BAB III METODOLOGI PENELITIAN 21](#_Toc164884993)

[3.1 Kerangka Berpikir 21](#_Toc164884994)

[3.2 *Business Understanding* 22](#_Toc164884995)

[3.3 *Data Understanding* 22](#_Toc164884996)

[3.4 *Data Preparation* 22](#_Toc164884997)

[3.5 *Modeling* 23](#_Toc164884998)

[3.6 *Evaluation* 23](#_Toc164884999)

[3.7 *Deployment* 24](#_Toc164885000)

[DAFTAR PUSTAKA 25](#_Toc164885001)

# PENDAHULUAN

## Latar Belakang Masalah

Penerapan ilmu komputer semakin meluas ke berbagai sektor, termasuk bidang geografis, pertanian, pariwisata, dan kedokteran. Perkembangan teknologi informasi juga memberikan dampak positif terhadap bidang pertanian, khususnya dalam manajemen dan identifikasi penyakit tanaman. Salah satu contohnya adalah dalam klasifikasi penyakit pada tanaman kentang. Tanaman kentang (*Solanum Tuberosum*) merupakan jenis umbi-umbian yang penting dalam pangan global, di samping gandum, jagung, dan beras (Auliatuzahra, 2024).

Kentang, sebagai salah satu tumbuhan dikotil dari keluarga *Solanaceae*, memiliki ciri khas yang mempengaruhi cara reproduksinya. Tanaman kentang umumnya berkembang biak melalui umbinya, yang memungkinkan penyebaran dan pertumbuhan yang efisien di lingkungan yang cocok. Faktor iklim yang lebih dingin cenderung mendukung pertumbuhan optimal tanaman kentang (Fuadi & Suharso, 2022). Selain itu, kentang memiliki peran vital sebagai sumber karbohidrat dalam memenuhi kebutuhan pangan manusia(Auliatuzahra, 2024). Dalam industri pangan, kentang menjadi bahan pokok dalam berbagai produk, menegaskan kepentingannya dalam menjaga ketahanan pangan. Namun, meskipun memiliki peran yang signifikan, pertanian kentang sering dihadapkan pada berbagai tantangan dan masalah. Salah satu permasalahan yang sering terjadi adalah munculnya penyakit pada daun tanaman kentang, yang dapat berdampak langsung pada hasil panen.

Penurunan produksi kentang sebesar 80% disebabkan oleh penyakit yang dapat menyebabkan penurunan hasil secara signifikan karena menghambat pertumbuhan (Fuadi & Suharso, 2022). Diduga bahwa jamur penyebab penyakit ini berasal dari umbi-umbi bibit. Pertumbuhan jamur pada umbi kentang dipengaruhi oleh suhu lingkungan. Jamur yang dimaksud adalah jamur *Patogen* dan jamur *Alternaria Solani* (Aumatullah et al., 2021). Secara umum, terdapat dua jenis penyakit yang disebabkan oleh jamur yang menyerang tanaman kentang, yakni *early blight* dan *late blight*(Fuadi & Suharso, 2022).

Gejala awal dari penyakit busuk daun (*late blight*) ini adalah adanya bercak basah yang muncul di tepi daun, yang juga dapat terjadi di bagian tengahnya. Bercak ini kemudian akan meluas dan menyebabkan perubahan warna daun menjadi coklat atau abu-abu. Sementara itu, gejala penyakit bercak kering (*early blight*) ditandai oleh munculnya bercak kering berwarna coklat dalam bentuk lingkaran di bagian bawah daun (Rozaqi et al., 2021). Dengan mengamati gejala yang muncul pada tanaman kentang, petani dapat segera mengatasi masalah penyakit daun yang membusuk. Namun, proses identifikasi gejala penyakit ini seringkali menghadapi kendala karena tidak semua petani memiliki pengetahuan yang memadai tentang cara mengenali gejala penyakit ini secara komprehensif.

Penanganan masalah penyakit pada daun kentang telah mengalami perkembangan yang signifikan, tidak hanya terbatas pada sektor pertanian, melainkan juga melibatkan kontribusi dari bidang teknologi. Salah satu kontribusi tersebut datang dari bidang informatika, yang mengimplementasikan teknik pengolahan citra digital atau yang lebih dikenal sebagai *image processing* dalam mengidentifikasi penyakit yang menginfeksi tanaman kentang. Pemanfaatan *image processing* dalam proses identifikasi ini membantu para pengelola pertanian untuk memberikan penanganan yang lebih efektif dan efisien terhadap tanaman yang terkena penyakit atau menunjukkan gejala tidak normal. Seiring dengan kemajuan teknologi, banyak penelitian yang mengembangkan pengolahan citra digital dalam konteks pertanian, baik untuk identifikasi penyakit maupun untuk menganalisis hasil produksi pertanian. Salah satu tujuan penelitian ini adalah untuk memanfaatkan citra digital dalam mengidentifikasi penyakit pada daun tanaman kentang.

Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi berbasis android yang dapat membantu petani atau pengelola pertanian dalam mengidentifikasi penyakit pada daun kentang dengan memanfaatkan data gambar daun kentang. Identifikasi penyakit pada daun tanaman kentang dibagi menjadi tiga kategori, yaitu daun sehat, *late blight*, dan *early blight*. Penelitian ini akan menggunakan algoritma *Convolutional Neural Network* (CNN), sebuah metode dalam bidang *Deep Learning*, untuk melakukan identifikasi tersebut.

Berdasarkan latar belakang yang telah diuraikan di atas, Peneliti akan mengambil judul "PENERAPAN METODE *CONVOLUTIONAL NEURAL NETWORK* (CNN) DALAM APLIKASI PENDETEKSI PENYAKIT DAUN TANAMAN KENTANG BERBASIS ANDROID” sebagai fokus utama penelitian ini.

## Rumusan Masalah

Berdasarkan latar belakang yang sudah dijelaskan, maka dapat dirumuskan permasalahan yang dapat diambil pada penelitian ini adalah bagaimana mengimplementasikan metode CNN dalam aplikasi pendeteksi penyakit daun tanaman kentang berbasis android?

## Batasan Masalah

Berdasarkan rumusan masalah yang telah diuraikan, penting untuk menetapkan batasan masalah agar penelitian tetap terfokus pada rencana yang telah disusun, dengan demikian tujuan utama penelitian dapat tercapai sesuai yang direncanakan. Oleh karena itu, penelitian ini akan dibatasi dengan beberapa aspek, yakni:

1. Aplikasi yang dibangun hanya untuk *platform* Android.
2. Metode yang digunakan dalam aplikasi ini adalah *Convolutional Neural Network.*
3. Bahasa pemrograman yang digunakan adalah *Javascript.*
4. *Framework* yang digunakan untuk membangun aplikasi berbasis Android adalah *React Native.*
5. Luaran dari aplikasi ini adalah menampilkan hasil deteksi penyakit daun tanaman kentang.

## Tujuan Penelitian

Mengacu pada rumusan masalah, maka tujuan yang ingin dicapai oleh peneliti adalah membangun aplikasi pendeteksi penyakit daun pada tanamann kentang berbasis android dengan mengimplementasikan metode *Convlutional Neural Network.*

## Manfaat Penelitian

Adapun beberapa manfaat pada penelitian ini terbagi menjadi dua yaitu bagi pengguna, dan bagi peneliti.

### Bagi Pengguna

1. Mempermudah pendeteksian penyakit daun pada tanaman kentang.
2. Memudahkan petani atau pengelola pertanian untuk segera melakukan penanganan terhadap penyakit *early blight* dan *late blight.*

### Bagi Peneliti

1. Meningkatkan kompetensi akademik peneliti, yang dapat diterapkan dalam studi lanjutan atau karir di masa depan.
2. Memberikan kesempatan bagi peneliti untuk menerapkan pengetahuan teoritis dalam konteks yang praktis dan relevan

## Sistematika Penulisan

Sistematika penulisan laporan yang akan disusun terbagi dalam beberapa bab, antara lain:

**BAB I PENDAHULUAN**

Bab ini menguraikan berbagai hal mengenai latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian dan sistematika penulisan. Uraian bab ini dimaksudkan untuk menjelaskan latar belakang penelitian sehingga dapat memberi masukan sesuai dengan tujuan penelitian dengan batasan-batasan masalah yang ada.

**BAB II TINJAUAN PUSTAKA**

Pada bab ini peneliti akan memaparkan dasar teori yang digunakan, serta rujukan dari penelitian terdahulu yang akan digunakan dalam perancangan aplikasi, dan membantu penulis untuk memahami algoritma *convolutional neural network* selama proses penelitian.

**BAB III METODOLOGI PENELITIAN**

Bab ini berisi tentang langkah-langkah terstruktur dan sistematis yang dilakukan dalam penelitian, dan mengungkapkan proses dan tahapan penelitian yang dimulai dari mendapatkan dataset hingga proses klasifikasi data uji yang ada.

**BAB IV HASIL DAN PEMBAHASAN**

Bab ini berisi tentang hasil penelitian, dimulai dengan hasil akhir aplikasi, klasifikasi data uji, dan akurasi dari aplikasi.

**BAB V KESIMPULAN DAN SARAN**

Bab ini berisi tentang kesimpulan dari penelitian yang telah dilakukan dan saran perbaikan yang perlu dilakukan pada objek penelitian.

# TINJAUAN PUSTAKA

## Penerapan

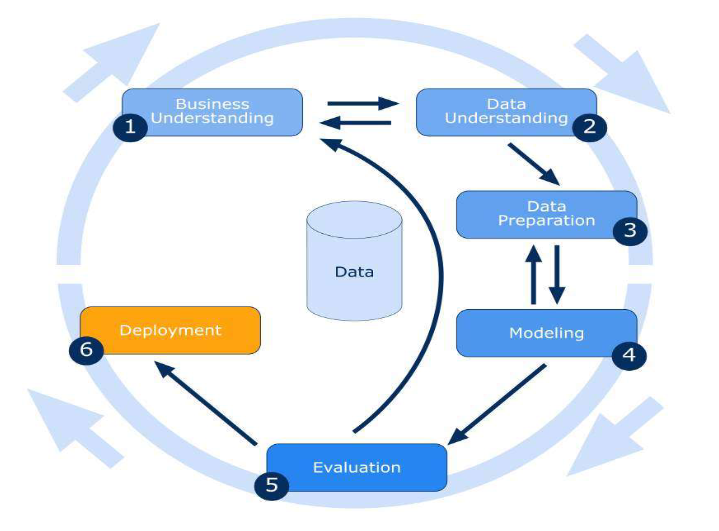
Menurut Kamus Besar Bahasa Indonesia (KBBI), penerapan adalah proses, cara, perbuatan menerapkan.

Penerapan merujuk pada pelaksanaan suatu teori, metode, atau konsep tertentu dengan tujuan mencapai hasil yang diinginkan. Tindakan ini dilakukan atas dasar rencana yang telah disusun sebelumnya untuk mencapai tujuan tertentu, dan dapat melibatkan penggunaan sumber daya yang tersedia secara efektif. Dalam konteks ini, penerapan adalah langkah konkret untuk menerapkan ide atau gagasan dalam praktik, dengan tujuan untuk memenuhi kebutuhan atau kepentingan tertentu(Salam et al., 2021).

## *Cross Industry Standard Process for Data Mining* (CRISP-DM)

CRISP-DM (*Cross Industry Standard Process for Data Mining*) didirikan pada akhir 1990-an oleh empat perusahaan utama: Integral Solutions Ltd. (perusahaan solusi data mining komersial), NCR (perusahaan database), DaimlerChrysler (produsen mobil), dan OHRA (perusahaan asuransi). CRISP-DM diakui sebagai proses standar yang digunakan untuk mengatasi masalah dalam praktik data mining di berbagai industri(Pambudi, 2023).

Menurut F. Krose (dalam Kurniawan & Yasir, 2022) Dalam siklus pengembangannya, CRISP-DM diakui sebagai metodologi *data mining* yang paling komprehensif dalam memenuhi kebutuhan proyek industri. Metodologi ini telah menjadi yang paling umum digunakan dalam proyek analitik, data mining, dan ilmu data. CRISP-DM terdiri dari enam tahapan, dimulai dari *Business Understanding*, kemudian *Data Understanding*, *Data Preparation*, *Modeling*, *Evaluation*, dan *Deployment*.



Gambar 2. 1 Diagram CRISP-DM (Hasanah et al., 2021)

Berikut 6 tahapan CRISP-DM (Hasanah et al., 2021):

1. *Business Understanding*

Pada tahap ini, kegiatan yang dilakukan meliputi memahami kebutuhan dan tujuan dari perspektif bisnis, menguraikan pengetahuan menjadi definisi masalah dalam konteks *data mining*, dan merencanakan serta menetapkan strategi untuk mencapai tujuan *data mining* tersebut.

1. *Data Understanding*

Data understanding merupakan bagian dari tahap persiapan di mana dilakukan pemeriksaan terhadap data yang akan digunakan, pengumpulan data awal, dan identifikasi terhadap kualitas data. Pada tahap ini, data akan dijelaskan secara rinci untuk setiap fitur yang ada.

1. *Data Preparation*

*Data Preparation* adalah langkah yang dilakukan setelah data terkumpul. Pada tahap ini, data akan menjalani proses identifikasi, seleksi, pembersihan, dan transformasi.

1. *Modeling*

*Modeling* adalah langkah implementasi algoritma untuk mencari, mengidentifikasi, dan menghasilkan pola yang akan diterapkan pada data penelitian.

1. *Evaluation*

*Evaluation* adalah proses untuk mengukur hasil evaluasi dari model yang telah diimplementasikan pada tahap sebelumnya, yaitu *modeling*. Hasil evaluasi tersebut mencerminkan proses *data mining* yang telah dilakukan dan menilai model yang paling optimal untuk digunakan.

1. *Deployment*

Menggunakan model yang dihasilkan dapat meliputi pembuatan laporan berdasarkan hasil analisis dari *data mining* atau penerapan proses *data mining* dalam konteks praktis.

## Sistem

Sistem dapat didefinisikan sebagai kumpulan prosedur yang interdependen dan terintegrasi, yang berkolaborasi untuk menjalankan aktivitas tertentu atau untuk mencapai tujuan yang spesifik. Dalam konteks lain, sistem diartikan sebagai agregasi dari dua atau lebih komponen yang berinteraksi dan saling terkait, yang bekerja bersama untuk mencapai sebuah objektif. Lebih lanjut, sistem juga diinterpretasikan sebagai entitas yang terdiri dari elemen-elemen data yang terkoneksi dan terorganisasi secara prosedural, membentuk sebuah kesatuan yang terstruktur(Sallaby & Kanedi, 2020).

## Kentang

Kentang (*Solanum tuberosum L.*) adalah jenis tanaman sayuran yang tumbuh setiap musim, dengan siklus hidup antara 90 hingga 180 hari, dan masuk dalam kategori tanaman perdu. Tanaman ini memerlukan tanah yang telah diolah dengan baik dan memiliki tekstur yang lembut. Idealnya, kentang ditanam di area yang berada di dataran tinggi atau wilayah pegunungan, di tempat yang memiliki ketinggian di atas 700 meter dari permukaan laut(Utami et al., 2015).

Kentang adalah contoh dari jenis sayuran yang masuk dalam kelompok sayuran yang membutuhkan iklim sejuk untuk tumbuh dan berkembang, dengan kebutuhan suhu berkisar antara 10 hingga 18 derajat *Celsius*. Tanaman ini memiliki asal usul dari kawasan Pegunungan Andes, khususnya di Peru dan Bolivia, sebelum akhirnya menyebar ke negara-negara seperti Chili, Kolombia, Ekuador, Spanyol, dan meluas ke seluruh wilayah Eropa(Vincent E. Rubatzky & Mas Yamaguchi, 1998). Kentang yang diperkenalkan ke Indonesia, yaitu varietas *Eigenheimer*, berasal dari Amerika. Varietas ini pertama kali ditemukan di wilayah sekitar Cimahi, Bandung, pada tahun 1794. Setelah itu, penyebarannya meluas ke berbagai daerah seperti Karo, Aceh, Padang, Bengkulu, Minahasa, Bali, Seram, dan Timor, menandakan adaptasi dan penerimaan terhadap tanaman ini di berbagai kondisi geografis Indonesia(Utami et al., 2015).

### Penyakit Daun Tanaman Kentang

Berikut ini beberapa penyakit yang dapat menyerang daun tanaman kentang:

1. *Early Blight*

Gejala dari penyakit bercak kering atau disebut juga *early blight* dapat ditemukan pada daun, batang, dan buah yang berusia lebih tua. Munculnya bercak dengan warna dari abu-abu ke coklat pada daun yang secara bertahap menyebar ke luar secara konsentris dari pusat yang tetap bersih, menghasilkan pola yang mirip dengan mata banteng. Sekitar luka ini, terdapat lingkaran dengan warna kuning terang yang mencolok. Seiring perkembangan penyakit, daun bisa menguning secara keseluruhan (klorosis) dan akhirnya rontok, yang dapat menyebabkan kehilangan daun secara signifikan (Plantix, 2023a).



Gambar 2. 2 Penyakit early blight (Ali, 2019)

*Early blight* disebabkan oleh dua jenis jamur yang memiliki kekerabatan dekat, yaitu *Alternaria tomatophila* dan *Alternaria solani*. Kedua jamur ini bertahan hidup di dalam tanah dan pada reruntuhan tanaman (L. Meno dalam Fuadi & Suharso, 2022). Banyak fungisida tersedia di pasaran untuk mengatasi penyakit bercak kering, termasuk produk yang mengandung *azoxystrobin*, *pyraclostrobin*, *difenoconazole*, *boscalid*, *chlorothalonil*, *fenamidone*, *maneb*, *mancozeb*, *trifloxystrobin*, dan *ziram*, baik secara tunggal maupun dalam kombinasi. Sangat dianjurkan untuk berganti-ganti antar senyawa kimia yang digunakan untuk mencegah resistensi. Aplikasi fungisida harus dilakukan secara tepat waktu, dengan memperhatikan prakiraan cuaca. Penting juga untuk memperhatikan interval waktu yang aman antara aplikasi terakhir fungisida dan panen, agar produk dapat dipanen dengan aman (Plantix, 2023a).

1. *Late Blight*

Penyakit busuk daun, dikenal juga sebagai *late blight*, umumnya muncul pada tanaman selama periode pertumbuhan minggu kelima hingga minggu keenam. Tanda-tanda awal dari penyakit ini mencakup kemunculan noda-noda lembap pada pinggiran daun, yang dapat juga terjadi di bagian tengahnya. Kemudian bercak ini akan melebar dan warna daun berubah menjadi coklat/abu-abu.(Rozaqi et al., 2021).



Gambar 2. 3 Penyakit late blight (Ali, 2019)

*Late blight* disebabkan oleh *Phytophthora infestans* yaitu suatu mikroorganisme yang menyukai lingkungan lembab dan sejuk (A. Majeed dalam Fuadi & Suharso, 2022). Tanda awal penyakit busuk daun yang paling jelas adalah bercak coklat atau bercak penyakit pada batang tanaman. Aplikasi fungisida penting untuk mengendalikan penyakit busuk daun, terutama di daerah lembab. Fungisida kontak yang melapisi daun dapat efektif sebelum infeksi dan tidak memicu kekebalan pada jamur. Fungisida yang mengandung *mandipropamid*, *chlorothalonil*, *fluazinam*, atau *mancozeb* juga dapat digunakan sebagai perlakuan pencegahan. Perlakuan benih sebelum disemai dengan fungisida seperti *mancozeb* juga berhasil(Plantix, 2023b).

## *Machine Learning*

*Machine learning* adalah salah satu cabang ilmu komputer yang bertujuan untuk mengembangkan algoritma yang memungkinkan komputer untuk belajar dari data yang ada. Dengan kata lain, komputer diajarkan untuk memahami pola dan informasi dari data historis sehingga dapat meningkatkan kinerja dalam menganalisis dan mengolah informasi yang diberikan. Hal ini membuat machine learning sering disebut sebagai proses "belajar dari data", di mana komputer memperoleh pengetahuan atau keterampilan baru melalui pengalaman yang diperoleh dari data yang ada. Tujuan utamanya adalah untuk menciptakan model atau program yang dapat mengoptimalkan performa dalam memahami dan menginterpretasikan data yang diberikan (Alpaydin dalam Sidik & Ansawarman, 2022).

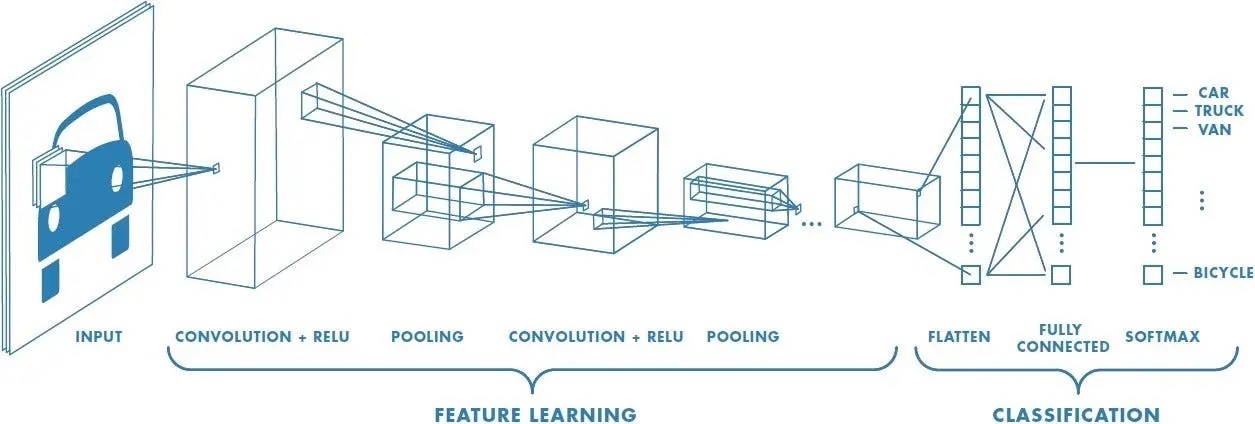
Menurut Tom M. Mitchell dalam (Sidik & Ansawarman, 2022) *Machine Learning* merupakan program komputer yang meningkatkan kemampuannya melalui pengalaman yang didapat dari melaksanakan tugas-tugas tertentu, di mana efektivitasnya dapat diukur secara objektif.

## *Deep Learning*

*Deep Learning* merupakan metode dalam *machine learning* yang memproses informasi kompleks secara nonlinier dengan memanfaatkan berbagai lapisan. Teknik ini bertujuan untuk mengidentifikasi pola, mengekstrak ciri, dan melakukan klasifikasi melalui model komputasi yang dibangun dari berlapis-lapis struktur pemrosesan. Dengan memanfaatkan data pada berbagai tingkat abstraksi, *deep learning* memungkinkan pembelajaran yang mendalam dari informasi yang diberikan (Radikto et al., 2022).

## *Convolutional Neural Network*

*Convolutional Neural Network* (CNN) adalah serangkaian lapisan pemrosesan yang menjalankan operasi konvolusi, bekerja secara paralel, dan terdiri dari beberapa elemen. Model ini terinspirasi oleh sistem saraf biologis(Romario et al., 2020). Pengolahan citra merupakan salah satu domain dalam deep learning yang dapat memecahkan masalahnya melalui metode *Convolutional Neural Network* (CNN). CNN efektif dalam menangani masalah yang kompleks, menawarkan kinerja yang unggul, dan mampu melakukan klasifikasi pada data citra yang diwakili dalam bentuk matriks. CNN juga termasuk dalam kategori *supervised learning*, di mana modelnya dilatih terlebih dahulu untuk mengidentifikasi pola antara input data dan label output(Harani et al., 2019).



Gambar 2. 4 Convlutional Neural Network Layers(Saha, 2018)

Gambar 2.4 menunjukan bahwa inputan akan mengalami proses konvolusi pada segmen pembelajaran fitur (*feature learning*). Fitur, yang merupakan elemen-elemen dalam citra yang diekstraksi oleh filter, kemudian dijadikan peta fitur (*feature map*). Dalam fase pembelajaran fitur, terdapat lapisan tersembunyi (*hidden layer*) yang mencakup lapisan-lapisan konvolusi dan pooling maksimum (*max pooling*). Menyusul segmen pembelajaran fitur, terdapat fase klasifikasi, di mana transformasi fitur menjadi bentuk satu dimensi dijalankan menggunakan teknik pelandaian (*flattening*). Selanjutnya, fitur tersebut diklasifikasikan melalui sejumlah *neuron* dan mencapai *neuron* keluaran, dengan jumlah yang setara dengan jumlah kelas yang ada dalam rangka klasifikasi.

### *Convolution Layer*

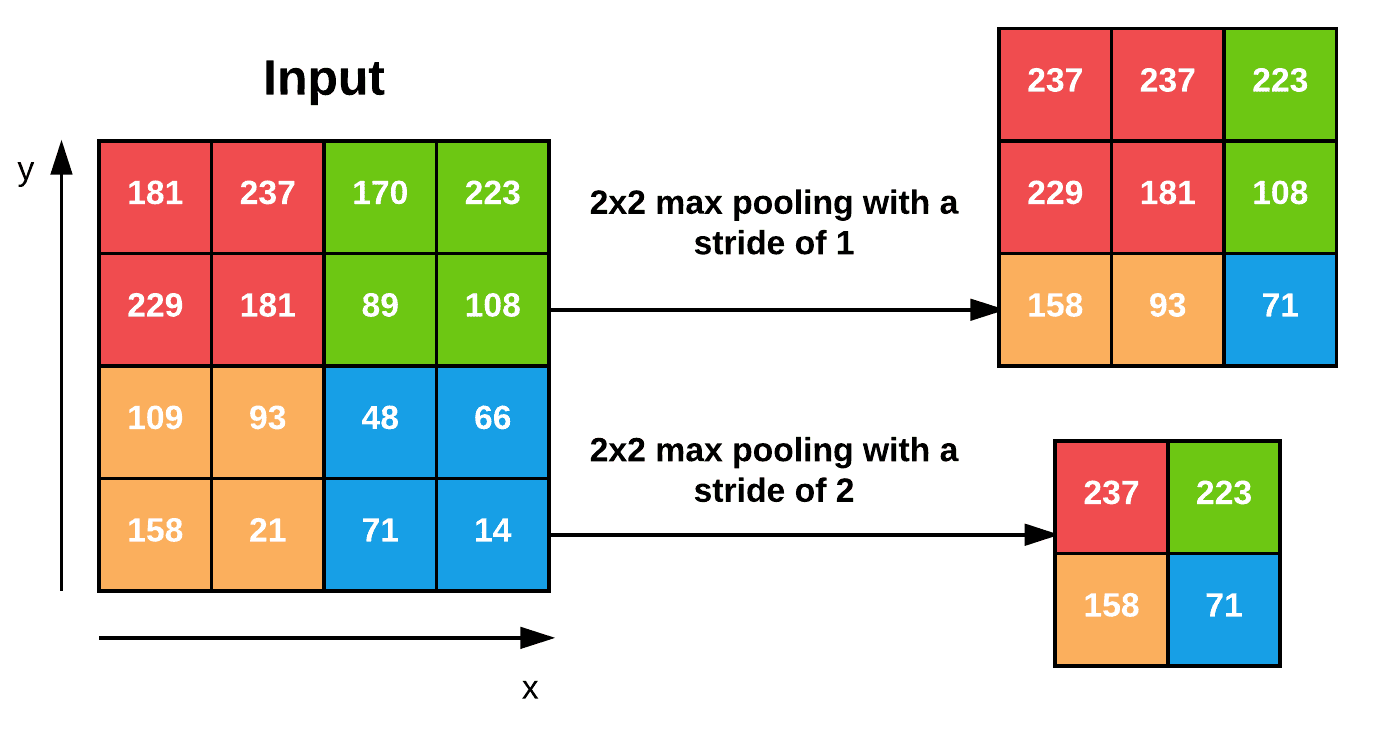
*Convolution Layer* atau Lapisan konvolusional adalah elemen kunci dalam arsitektur Jaringan Saraf Tiruan Konvolusional. Komponen ini terdiri dari kumpulan filter yang diterapkan melalui operasi konvolusi pada input yang disajikan. Operasi ini menghasilkan apa yang dikenal sebagai peta fitur. Dalam konteks lapisan konvolusional, setiap filter diwakili oleh matriks yang mengandung serangkaian nilai yang unik. Nilai-nilai ini, yang berfungsi mirip dengan bobot, akan mengalami proses pembaruan sepanjang fase pelatihan sesudah dilakukannya konvolusi. Operasi konvolusi itu sendiri melibatkan perkalian matriks antara filter dan lapisan konvolusional, dengan output dari operasi ini kemudian diakumulasikan untuk menghasilkan peta fitur yang diinginkan (Denata et al., 2021).



Gambar 2. 5 Proses Konvolusi (Mostafa & Fang-Xiang Wu, 2021)

### *Pooling Layer*

*Pooling Layer* adalah sebuah komponen penting yang terletak antara lapisan konvolusi dalam arsitektur jaringan saraf tiruan. Fungsi utama dari layer ini adalah untuk melakukan pengurangan volume pada keluaran dari peta fitur, yang dapat menghasilkan percepatan komputasi pada lapisan berikutnya (Denata et al., 2021).



Gambar 2. 6 Proses Pooling (Rosebrock, 2021)

### *Fully Connected Layer*

*Fully connected layer* dalam jaringan saraf tiruan adalah struktur yang menghubungkan setiap *neuron* aktivasi dari lapisan sebelumnya secara langsung dengan setiap *neuron* pada lapisan berikutnya. Perbedaan mendasar antara lapisan terhubung penuh dan lapisan konvolusi adalah bahwa dalam lapisan konvolusi, *neuron-neuron* hanya terhubung dengan wilayah tertentu dari input, sementara dalam lapisan terhubung penuh, setiap *neuron* terhubung dengan semua *neuron* pada lapisan sebelumnya(Denata et al., 2021). Pada titik ini, semua fitur yang telah dipelajari oleh jaringan dari gambar akan digunakan untuk melakukan klasifikasi. Lapisan terhubung penuh dapat dianggap sebagai lapisan dari jaringan saraf tiruan tradisional (*multilayer perceptrons*) yang digunakan untuk mengklasifikasikan gambar berdasarkan fitur-fitur yang telah diidentifikasi oleh lapisan sebelumnya.

## TensorFlow

TensorFlow adalah sebuah *framework* komputasi yang terkenal dalam pembangunan model pembelajaran mesin, menawarkan serangkaian alat yang beragam yang memfasilitasi pembangunan model pada berbagai tingkat abstraksi. Dengan kemampuannya yang luar biasa, TensorFlow mampu menjalankan grafik komputasi pada berbagai platform hardware seperti CPU, GPU, dan TPU (Hikmatia A. E. & Ihsan Zul, 2021). Fleksibilitas ini memungkinkan para peneliti dan praktisi untuk memanfaatkan sumber daya komputasi yang tersedia dengan cara yang paling efisien dan efektif untuk kebutuhan pengembangan model mereka. Sebagai tambahan, *TensorFlow* juga memungkinkan adaptabilitas yang tinggi dalam merancang dan mengimplementasikan model pembelajaran mesin, memastikan bahwa pengguna dapat menyesuaikan model mereka dengan kebutuhan khusus dan infrastruktur yang tersedia dengan optimal.

## *Keras*

*Keras* merupakan perangkat lunak berbasis Python yang dirancang untuk fasilitasi komputasi cerdas, termasuk dalam hal ini adalah pengolahan citra, konversi formula, dan proses klasifikasi. Sebagai pustaka jaringan saraf yang *user-friendly*, *Keras* beroperasi dengan menggunakan *TensorFlow* dan *Theano* sebagai *backend*, memungkinkan penggunaannya secara efisien baik pada unit pemrosesan sentral (CPU) maupun unit pemrosesan grafis (GPU) (Ihsan, 2021). *Framework* ini menawarkan dukungan luas terhadap berbagai arsitektur jaringan saraf, mulai dari jaringan yang sepenuhnya terkoneksi, jaringan konvolusional, lapisan *pooling*, jaringan berulang, hingga lapisan *embedding*. *Keras* memungkinkan pengguna untuk mengintegrasikan model-model tersebut dalam pembuatan sistem yang lebih kompleks dan multifungsi. Berkat desainnya yang berbasis *Python*, *Keras* tidak hanya mudah untuk dipasang dan digunakan tetapi juga mendukung eksplorasi dan pembelajaran yang mendalam terhadap pemrograman jaringan saraf. Lebih jauh, integrasi *Keras* dengan *backend* seperti *TensorFlow* dan *Theano* menjamin skalabilitas dan optimasi kinerja, memastikan bahwa aplikasi dapat dikembangkan dengan fleksibilitas tinggi sesuai dengan kebutuhan spesifik pengguna.

## Android

Android merupakan sistem operasi untuk perangkat bergerak yang dikembangkan di atas kernel Linux, mencakup tidak hanya sistem operasi itu sendiri tetapi juga perangkat lunak antara (*middleware*) dan berbagai aplikasi. Sistem ini menawarkan sebuah platform yang bersifat terbuka, memungkinkan para developer untuk merancang dan mengembangkan aplikasi mereka dengan lebih leluasa. Sebagai sebuah ekosistem yang komprehensif, Android dirancang untuk mendukung inovasi dan kreativitas pengembang aplikasi, dengan menyediakan akses ke fungsi-fungsi perangkat keras perangkat mobile melalui *API* yang kaya dan fleksibel. Melalui pendekatan ini, Android berkontribusi pada evolusi teknologi *mobile*, memperkaya pengalaman pengguna dengan beragam aplikasi yang dapat disesuaikan dan ditingkatkan sesuai dengan kebutuhan dan preferensi individu(Azis et al., 2020).

## React Native

React Native adalah sebuah framework yang menggunakan JavaScript sebagai basisnya, digunakan untuk mengembangkan aplikasi seluler yang dapat berjalan pada kedua platform, yaitu Android dan iOS, secara simultan. Kerangka kerja ini memungkinkan pengembang untuk menulis kode sekali dan menggunakan kembali komponen-komponen yang sama untuk kedua platform tersebut, meningkatkan efisiensi dan produktivitas dalam proses pengembangan aplikasi. Dengan demikian, React Native memberikan solusi yang efektif dan efisien bagi pengembang dalam menciptakan aplikasi seluler yang mendukung kedua sistem operasi utama secara bersamaan(Nurhidayah et al., 2023).

## Penelitian Terdahulu

Berikut ini adalah acuan penelitian terdahulu.

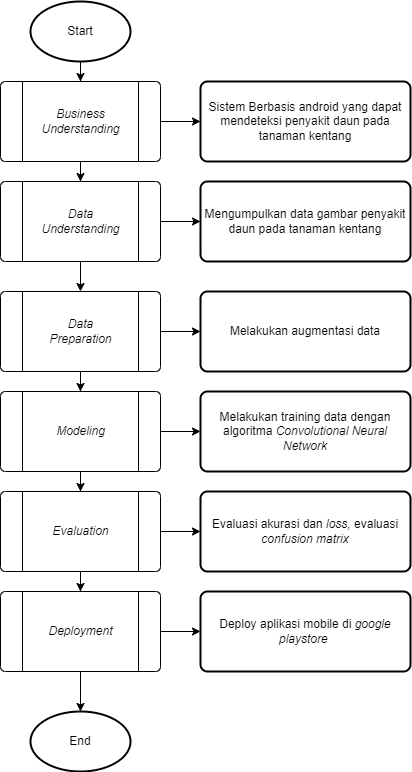
Tabel 2. 1 Penelitian terdahulu

| Penulis | Judul | Metode | Hasil Penelitian |
| --- | --- | --- | --- |
| Anwar Fuadi, Aries Suharso | Perbandingan Arsitektur *Mobilenet* Dan  *NASNetMobile* Untuk Klasifikasi Penyakit Pada Citra Daun Kentang | Menggunkan metode *Convlutional Neural Network* dengan arsitektur *Mobilenet dan NASNetMobile* | Data yang digunakan pada penelitian tersebut terbagi menjadi gambar daun kentang sehat, gambar daun kentang terinfeksi *Early Blight*, dan gambar daun kentang terinfeksi *Late Blight*. Di akhir penelitian ditemukan bahwa pada skema pengujian menggunakan arsitektur *NASNetMobile* dengan perbandingan data latih dan data uji sebesar 90:10 menghasilkan *accuracy* sebesar 90.96%, *precision* sebesar 90.86%, *recall* sebesar 91.11%, dan f1 score sebesar 92.93%. |
| Abdul Jalil Rozaqi, Andi Sunyoto, Rudyanto Arief | Deteksi Penyakit pada Daun Kentang Menggunakan Pengolahan Citra  dengan Metode *Convolutional Neural Network* | Metode yang digunakan adalah *Convolutional Neural Network* | Dengan pembagian dataset 80% dan 20% dan gambar yang digunakan ukuranya dirubah 150x150. Pada *epoch* ke 10 dengan *batch\_size* 20 dengan total data *training* 922 gambar dan data *testing* 230 gambar menghasilkan nilai akurasi 95% dan untuk akurasi validasi menghasilkan 94% |
| Luthfiyah Amatullah, Ivtytah Ein, Mayanda Mega Santoni | Identifikasi Penyakit Daun Kentang Berdasarkan Fitur Tekstur dan Warna Dengan Menggunakan Metode *K-Nearest Neighbor* | Menggunakan metode *K- Nearest Neighbors* dan *Gray Level Co-occurrence Matrix* (GLCM) untuk melakukan ekstraksi fitur tekstur | Penelitian ini mengidentifikasi penyakit pada daun kentang melalui ekstraksi fitur tekstur menggunakan *Gray Level Co-occurrence Matrix* (GLCM) dan fitur warna menggunakan metode K-*Nearest Neighbor* (KNN) pada implementasi sistem. Dataset yang digunakan terdiri dari 270 citra daun kentang, yang dibagi menjadi tiga kelompok: 90 citra untuk data pelatihan, 90 citra untuk data pengujian, dan 90 citra untuk data validasi. Klasifikasi KNN diuji dengan nilai K sebesar 2, 3, 5, 7, 9, dan 10. Hasil pengujian menunjukkan bahwa nilai K terbaik adalah 3, dengan akurasi mencapai 80%. Namun, terdapat 21 data citra uji yang diklasifikasikan secara berbeda dari kelas sebenarnya. |
| Irvan Denata, Tedy Rismawan, Ikhwan Ruslianto | Implementasi *Deep Learning* untuk Klasifikasi Jenis Jeruk dengan Menggunakan Metode *Convolutional Neural Network* | Metode yang digunakan adalah *Convolutional Neural Network* | Penelitian ini menunjukkan bahwa meskipun jumlah epoch dalam pelatihan model meningkat hingga *epoch* ke-8, akurasi latih tidak selalu mencerminkan akurasi uji yang optimal. Model pada *epoch* ke-9, yang memiliki nilai *loss* terendah, menghasilkan akurasi uji tertinggi (94,81%). Oleh karena itu, peningkatan jumlah epoch tidak menjamin hasil yang lebih baik. Model paling optimal adalah yang memiliki nilai *loss* terendah, seperti yang terjadi pada epoch ke-9, dengan nilai *presisi, recall*, dan F1-measure tertinggi sebesar 94,90%, 94,81%, dan 94,85% serta akurasi pengujian sebesar 94,81%. |
| M. Hamsy Romario, Eko Ihsanto, Trie Maya Kadarina | Sistem Hitung Dan Klasifikasi Objek Dengan Metode Convolutional *Neural Network* | Metode yang digunakan adalah *Convolutional Neural Network* | Hasil penelitian menunjukkan bahwa program ini efektif dalam memproses gambar dengan dimensi tertentu, meskipun terdapat kesalahan dalam mendeteksi pola wajah manusia pada gambar yang sebenarnya tidak mengandung wajah manusia. Namun, program berhasil membedakan objek manusia dengan tingkat keberhasilan sekitar 85%. Program juga mampu mengenali detail wajah manusia dengan baik, bahkan pada gambar karikatur. Selain itu, ketika diuji dengan gambar kartun SpongeBob SquarePants, program berhasil membedakan tanpa menampilkan frame objek manusia, mencapai tingkat keberhasilan 95%. |
| Abwabul Jinan, B. Herawan Hayadi | Klasifikasi Penyakit Tanaman Padi Mengunakan Metode *Convolutional Neural Network* Melalui Citra Daun (*Multilayer Perceptron*) | Metode yang digunakan adalah *Convolutional Neural Network* | Implementasi algoritma CNN untuk mengklasifikasikan citra daun padi yang terserang penyakit melibatkan pencarian arsitektur terbaik dengan membandingkan beberapa parameter. Arsitektur terbaik menggunakan citra 100x100 piksel, *kernel* 3x3, *learning* *rate* 0,01, *optimizer* Adam, epoch 150, *batch size* 30, dan skenario dataset 90%:10% dengan citra RGB. Hasil pengujian menunjukkan akurasi 91,7% pada data testing. |
| Qudsiah Nur Azizah | Klasifikasi Penyakit Daun Jagung Menggunakan Metode *Convolutional Neural Network AlexNet* | Metode yang digunakan adalah *Convolutional Neural Network* denganarsitektur *AlexNet* | Sistem yang memanfaatkan arsitektur *AlexNet* berhasil mengklasifikasi penyakit daun jagung dengan akurasi mencapai 90%. Hal ini menunjukkan bahwa metode yang diusulkan efektif dalam pengenalan jenis penyakit pada tanaman daun jagung. |
| Nining Putri Ningsih, Emi Suryadi, Lalu Darmawan Bakti, Bahtiar Imran | Klasifikasi Penyakit *Early Blight* Dan *Late Blight* Pada  Tanaman Tomat Berdasarkan Citra Daun Menggunakan Metode  Cnn Berbasis Website | Metode yang digunakan adalah *Convolutional Neural Network* | Metode Convolutional Neural Network (CNN) digunakan untuk mengklasifikasi penyakit pada daun tomat. Proses training menggunakan learning rate 0,0001 dan batch size 20. Pada epoch 100, tercapai loss 5%, akurasi 99%, dan recall 85%, hasil terbaik dari percobaan. Training melibatkan 4.000 citra dan 2 jenis penyakit. Akurasi dan recall pada data uji mencapai 80% dan 85% secara berturut-turut, menunjukkan kemampuan CNN dalam mengidentifikasi penyakit pada daun tomat secara efektif. |

# METODOLOGI PENELITIAN

## Kerangka Berpikir

Berikut gambar 3.1 kerangka berpikir dari penelitian ini.



Gambar 3. 1 Kerangka berpikir

Pengembangan model pembelajaran dalam pada penelitian ini menerapkan kerangka kerja CRISP-DM, yang dibagi menjadi enam tahap. Tahap-tahap tersebut dimulai dengan Business Understanding, diikuti oleh *Data Understanding*, kemudian berlanjut ke *Data Preparation*, *Modeling, Evaluation*, dan berakhir dengan *Deployment*.

## *Business Understanding*

Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi seluler yang mampu mengklasifikasikan penyakit pada daun tanaman kentang menggunakan teknik pembelajaran terarah dengan memanfaatkan algoritma *Convolutional Neural Network*. Aplikasi ini akan mengidentifikasi penyakit pada daun tanaman kentang dengan mengambil gambar melalui kamera ponsel, dan selanjutnya akan menghasilkan output berupa tipe penyakit berdasarkan gambar yang diperoleh.

## *Data Understanding*

*Dataset* yang digunakan dalam penelitian ini terdiri dari gambar penyakit pada daun tanaman kentang, yang diakses dari situs *Kaggle*. Gambar dalam dataset ini menampilkan variasi ukuran piksel dan mencakup tiga *class* atau label berbeda. Berikut ini *class* tersebut:

Tabel 3. 1 *Class* *dataset*

|  |  |  |
| --- | --- | --- |
| *Class* | Definisi | Sumber |
| *Healthy* | Gambar daun tanaman kentang yang sehat | <https://www.kaggle.com/datasets/abdallahalidev/plantvillage-dataset> (Ali, 2019) |
| *Late Blight* | Gambar daun tanaman kentang yang terinfeksi penyakit busuk daun |
| *Early Blight* | Gambar daun tanaman kentang yang terinfeksi penyakit bercak kering |

## *Data Preparation*

Sebelum dilakukan training model dataset perlu untuk dilakukan *data augmentation.* Menurut D. Seita (dalam Sanjaya & Ayub, 2020) augmentasi data adalah suatu metode yang membolehkan para praktisi untuk menambah variasi data yang ada untuk pelatihan model tanpa perlu mengumpulkan data baru. Beberapa contoh teknik augmentasi data antara lain adalah pemotongan (*cropping*), penambahan tepi (*padding*), dan pembalikan gambar secara horizontal (*horizontal* *flipping*). Berikut adalah *data augmentation* yang diterapkan pada penelitian ini:

1. *Rescale*

Dalam penelitian ini, proses mengubah skala dilakukan pada citra dataset untuk memastikan semua gambar memiliki ukuran yang seragam, yaitu 256 piksel. Hal ini bertujuan untuk memudahkan komputasi selama proses pelatihan model.

1. *Split Dataset*

Dalam penelitian ini, dataset dibagi menjadi dua bagian, yaitu 80% untuk data pelatihan, 10% untuk data validasi dan 10% untuk data uji. Pembagian ini bertujuan untuk memenuhi kebutuhan pelatihan, validasi dan uji model.

1. *Rotation*

Berfungsi untuk melakukan rotasi pada sebuah citra. Penelitian ini dilakukan rotasi sebesar 20 derajat.

1. *Horizontal* dan *Vertical Flip*

*Horizontal* dan *vertical flip* berfungsi untuk membalik sebuah citra menjadi *horizontal* ataupun *vertical*

## *Modeling*

## *Evaluation*

Pada tahap ini, model yang telah dilatih akan dievaluasi dengan melihat nilai dari grafik tingkat akurasi, yang mencakup *training accuracy* dan *validation accuracy*, serta grafik tingkat *error* yang mencakup *training loss* dan *validation loss*. Jika nilai-nilai tersebut rendah atau terjadi *overfitting* atau *underfitting*, akan dilakukan perancangan ulang dan pelatihan model kembali.

Tahap kedua dari evaluasi melibatkan pembuatan *confusion matrix* untuk model yang telah dibangun. Menurut D. Putra (dalam Normawati & Prayogi, 2021) *confusion matrix* adalah sebuah tabel yang menunjukkan jumlah data uji yang diklasifikasikan secara benar dan jumlah data uji yang diklasifikasikan secara salah.

## *Deployment*

Melakukan *deployment* aplikasi *mobile* sehingga aplikasi dapat di unduh dan dapat digunakan oleh siapapun yang menggunakan *smartphone* dengan sistem operasi Android*.*

DAFTAR PUSTAKA

Ali, A. (2019). *PlantVillage Dataset*. kaggle. https://www.kaggle.com/datasets/abdallahalidev/plantvillage-dataset

Auliatuzahra, E. (2024). ANALISIS PERAN BIOTEKNOLOGI REKAYASA GENETIKA UNTUK PENINGKATAN KUALITAS PADA TOMAT DAN KENTANG. *HUMANITIS: Jurnal Humaniora, Sosial dan Bisnis*, *2*(1), 113–123.

Aumatullah, L., Ein, I., & Santoni, M. M. (2021). Identifikasi Penyakit Daun Kentang Berdasarkan Fitur Tekstur dan Warna Dengan Menggunakan Metode K-Nearest Neighbor. *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)*, *April*, 783–791.

Azis, N., Gali Pribadi, & Manda Savitrie Nurcahya. (2020). Analisa dan Perancangan Aplikasi Pembelajaran Bahasa Inggris Dasar Berbasis Android. *IKRA-ITH Informatika*, *4*(3), 251–255.

Denata, I., Rismawan, T., Ruslianto, I., Rekayasa Sistem Komputer, J., & MIPA Universitas Tanjungpura, F. (2021). Implementasi Deep Learning untuk Klasifikasi Jenis Jeruk dengan Menggunakan Metode Convolutional Neural Network. *Jurnal Informatika dan Teknologi Informasi*, *18*(3), 297–307. https://doi.org/10.31515/telematika.v18i3.5541

Fuadi, A., & Suharso, A. (2022). Perbandingan Arsitektur Mobilenet Dan Nasnetmobile Untuk Klasifikasi Penyakit Pada Citra Daun Kentang. *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, *7*(3), 701–710. https://doi.org/10.29100/jipi.v7i3.3026

Harani, N. H., Prianto, C., & Hasanah, M. (2019). Deteksi Objek Dan Pengenalan Karakter Plat Nomor Kendaraan Indonesia Menggunakan Metode Convolutional Neural Network (CNN) Berbasis Python. *Jurnal Teknik Informatika*, *11*(3), 47–53. https://ejurnal.ulbi.ac.id/index.php/informatika/article/view/658

Hasanah, M. A., Soim, S., & Handayani, A. S. (2021). Implementasi CRISP-DM Model Menggunakan Metode Decision Tree dengan Algoritma CART untuk Prediksi Curah Hujan Berpotensi Banjir. *Journal of Applied Informatics and Computing*, *5*(2), 103–108. https://doi.org/10.30871/jaic.v5i2.3200

Hikmatia A. E., N., & Ihsan Zul, M. (2021). Aplikasi Penerjemah Bahasa Isyarat Indonesia menjadi Suara berbasis Android menggunakan Tensorflow. *Jurnal Komputer Terapan*, *7*(1), 74–83.

Ihsan, C. N. (2021). Klasifikasi Data Radar Menggunakan Algoritma Convolutional Neural Network (CNN). *DoubleClick: Journal of Computer and Information Technology*, *4*(2), 115. https://doi.org/10.25273/doubleclick.v4i2.8188

Kurniawan, D., & Yasir, M. (2022). Optimization Sentimen Analysis using CRISP-DM and Naive Bayes Methods Implemented on Social Media. *Cyberspace: Jurnal Pendidikan Teknologi Informasi*, *6*(2), 74. https://doi.org/10.22373/cj.v6i2.12793

Mostafa, S., & Fang-Xiang Wu. (2021). Neural Engineering Techniques for Autism Spectrum Disorder. *Imaging and Signal Analysis*, *1*, 23–38. https://doi.org/https://doi.org/10.1016/B978-0-12-822822-7.00003-X

Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, *5*(2), 697–711.

Nurhidayah, L., Salsabillah, A., & Yanti, F. R. (2023). Perancangan Aplikasi Pemesanan Tiket Bioskop di Kota Medan Berbasis Android. *Laila Nurhidayah*, *2*(2), 305–314.

Pambudi, A. (2023). Penerapan Crisp-Dm Menggunakan Mlr K-Fold Pada Data Saham Pt. Telkom Indonesia (Persero) Tbk (Tlkm) (Studi Kasus: Bursa Efek Indonesia Tahun 2015-2022). *Jurnal Data Mining dan Sistem Informasi*, *4*(1), 1. https://doi.org/10.33365/jdmsi.v4i1.2462

Plantix. (2023a). *Bercak kering*. Plantix.net. https://plantix.net/id/library/plant-diseases/100321/early-blight/

Plantix. (2023b). *Hawar Daun Kentang*. Plantix.net. https://plantix.net/id/library/plant-diseases/100040/potato-late-blight/

Radikto, Mulyana, D. I., Rofik, M. A., & Zakaria, Mo. Z. Z. (2022). Klasifikasi Kendaraan pada Jalan Raya menggunakan Algoritma Convolutional Neural Network ( CNN ). *Jurnal Pendidikan Tambusai*, *6*(1), 1668–1679.

Romario, M. H., Ihsanto, E., & Kadarina, T. M. (2020). Sistem Hitung dan Klasifikasi Objek dengan Metode Convolutional Neural Network. *Jurnal Teknologi Elektro*, *11*(2), 108. https://doi.org/10.22441/jte.2020.v11i2.007

Rosebrock, A. (2021). *Convolutional Neural Networks (CNNs) and Layer Types*. pyimagesearch. https://pyimagesearch.com/2021/05/14/convolutional-neural-networks-cnns-and-layer-types/

Rozaqi, A. J., Sunyoto, A., & Arief, M. rudyanto. (2021). Deteksi Penyakit Pada Daun Kentang Menggunakan Pengolahan Citra dengan Metode Convolutional Neural Network. *Creative Information Technology Journal*, *8*(1), 22. https://doi.org/10.24076/citec.2021v8i1.263

Saha, S. (2018). *A Guide to Convolutional Neural Networks — the ELI5 way*. Sturn Cloud Blog. https://saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way/

Salam, N. F. S., Manap Rifai, A., & Ali, H. (2021). Faktor Penerapan Disiplin Kerja: Kesadaran Diri, Motivasi, Lingkungan (Suatu Kajian Studi Literatur Manajemen Pendidikan Dan Ilmu Sosial). *Jurnal Manajemen Pendidikan Dan Ilmu Sosial*, *2*(1), 487–508. https://doi.org/10.38035/jmpis.v2i1.503

Sallaby, A. F., & Kanedi, I. (2020). Perancangan Sistem Informasi Jadwal Dokter Menggunakan Framework Codeigniter. *Jurnal Media Infotama*, *16*(1), 48–53. https://doi.org/10.37676/jmi.v16i1.1121

Sanjaya, J., & Ayub, M. (2020). Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup. *Jurnal Teknik Informatika dan Sistem Informasi*, *6*(2), 311–323. https://doi.org/10.28932/jutisi.v6i2.2688

Sidik, A. D., & Ansawarman, A. (2022). Prediksi Jumlah Kendaraan Bermotor Menggunakan Machine Learning. *Formosa Journal of Multidisciplinary Research*, *1*(3), 559–568. https://doi.org/10.55927/fjmr.v1i3.745

Utami, G. R., Rahayu, M. S., & Setiawan, A. (2015). Penanganan Budidaya Kentang (Solanum tuberosum L.) di Bandung, Jawa Barat. *Buletin Agrohorti*, *3*(1), 105–109. https://doi.org/10.29244/agrob.v3i1.14833

Vincent E. Rubatzky, & Mas Yamaguchi. (1998). *World Vegetables: Principles, Production, and Nutritive Values*. Chapman & Hall.