

Penerapan CNN untuk Identifikasi Pneumonia berdasarkan Citra Rontgen

Louis Caesa Kesuma (13521069)
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13521069@std.stei.itb.ac.id

Abstract—Pneumonia adalah penyakit radang paru-paru yang disebabkan oleh infeksi virus, bakteri, atau jamur. Pneumonia merupakan salah satu penyebab kematian tertinggi pada anak-anak di seluruh dunia, sehingga diperlukan metode pengidentifikasian pneumonia yang akurat untuk meminimalisir dampaknya. Makalah ini menerapkan *Convolutional Neural Network* (CNN) untuk mengidentifikasi pneumonia berdasarkan citra rontgen dada. Dataset yang digunakan dalam makalah ini terdiri dari citra rontgen dada pasien dengan kondisi normal dan pasien yang terdiagnosis pneumonia. Penelitian dilakukan untuk menunjukkan potensi CNN dalam membantu diagnosis medis untuk penyakit seperti pneumonia didasari oleh evaluasi kinerja model yang dikembangkan.

Keywords—CNN, Pneumonia, Citra Rontgen, Diagnosis Medis

I. PENDAHULUAN

Pneumonia adalah penyakit radang paru-paru yang disebabkan oleh infeksi virus, bakteri, atau jamur. Penyakit ini menjadi salah satu penyebab utama kematian, terutama pada anak-anak di seluruh dunia. Data dari Organisasi Kesehatan Dunia (WHO) menunjukkan bahwa pneumonia menyumbang persentase yang signifikan terhadap angka kematian anak-anak, khususnya di negara-negara berkembang. Sehingga ada urgensi akan metode atau teknik yang dapat mendiagnosis pneumonia secara akurat untuk mengurangi tingkat kematian akibat penyakit ini.

Salah satu pendekatan yang umum digunakan untuk membantu pendiagnosaan pneumonia adalah melalui analisis citra rontgen dada. Proses ini umumnya membutuhkan keahlian dan pengalaman khusus dari seorang radiolog untuk menafsirkan citra secara akurat, namun tentunya pendekatan ini dapat memakan waktu yang lama dan masih berkemungkinan salah dikarenakan oleh kesalahan manusia. Sehingga akan lebih baik jika ada metode yang dapat melakukan diagnosis atau membantu dalam proses pendiagnosaan pneumonia agar kesalahannya dapat diminimalisir.

Dalam beberapa tahun terakhir, perkembangan teknologi *Deep Learning* terutama *Convolutional Neural Network* (CNN) telah membuka potensi-potensi baru dalam pemrosesan citra terutama pada bidang medis. CNN identik dengan kemampuannya dalam mengenali pola visual, sehingga CNN sangat ideal untuk analisis citra seperti pengenalan pola pneumonia pada citra rontgen dada.

Makalah ini bertujuan untuk menerapkan CNN dalam mengidentifikasi pneumonia berdasarkan citra rontgen dada. Dataset yang digunakan mencakup citra rontgen dada dari pasien dengan kondisi normal dan pasien yang terdiagnosis mengidap pneumonia. Penelitian ini dilakukan untuk mengevaluasi kinerja model CNN dalam mendiagnosa atau membantu pendiagnosaan pneumonia secara akurat menggunakan citra rontgen dada. Diharapkan, penelitian ini dapat memberikan kontribusi yang signifikan dalam memperluas penerapan teknologi *Deep Learning* untuk mendiagnosa pneumonia atau penyakit lainnya di masa depan.

II. LANDASAN TEORI

A. Citra

Citra adalah sinyal dwimatra yang bersifat menerus/kontinu yang dapat diamati oleh sistem visual manusia. Secara matematis citra dapat direpresentasikan sebagai $f(x,y)$, yang di mana (x,y) adalah koordinat dan $f(x,y)$ adalah intensitas cahaya pada titik (x,y) . Setiap citra dapat direpresentasikan menjadi sebuah matriks yang dimana setiap elemennya menyatakan piksel (*picture element*). Piksel memiliki nilai tertentu seperti intensitas warna, bayangan, atau opasitas yang berfungsi sebagai penyusun citra digital.

Secara umum citra dapat dibagi menjadi 3 jenis, yaitu

1. Citra biner



Gambar 2.1.1. Citra Biner (Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2024-2025/03-Pembentukan-Citra-dan-Digitalisasi-Citra-2024.pdf>)

Citra biner direpresentasikan dengan 1 bit saja untuk setiap pikselnya, sehingga citranya hanya berwarna hitam atau putih.

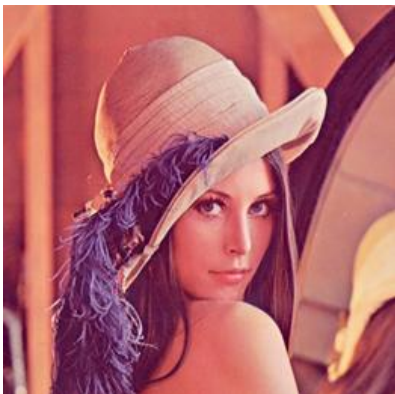
2. Citra *grayscale*



Gambar 2.1.2. Citra *Grayscale* (Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2024-2025/03-Pembentukan-Citra-dan-Digitalisasi-Citra-2024.pdf>)

Citra *grayscale* adalah citra yang setiap pikselnya direpresentasikan dengan 8-bit, sehingga setiap pikselnya memiliki nilai dalam rentang 0 sampai 255. Nilai piksel tersebut memungkinkan citra *grayscale* untuk memiliki warna dalam rentang hitam hingga putih. Citra rontgen juga termasuk kedalam citra *grayscale* karena citranya memiliki warna dalam rentang hitam hingga putih.

3. Citra berwarna

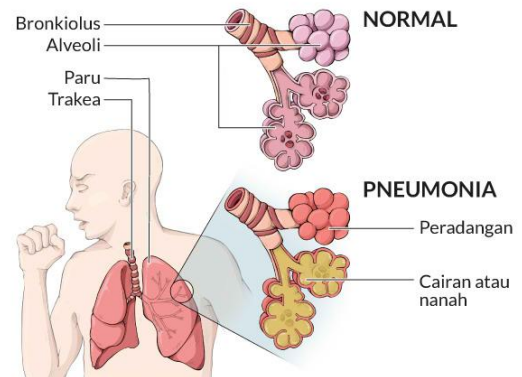


Gambar 2.1.3. Citra Berwarna (Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2024-2025/03-Pembentukan-Citra-dan-Digitalisasi-Citra-2024.pdf>)

Citra berwarna adalah citra yang memiliki 3 kanal, yaitu *red* (R), *green* (G), dan *blue* (B). Untuk masing-masing kanal, setiap pikselnya direpresentasikan dalam 8-bit. Sehingga setiap pikselnya direpresentasikan dalam 24-bit.

B. *Pneumonia*

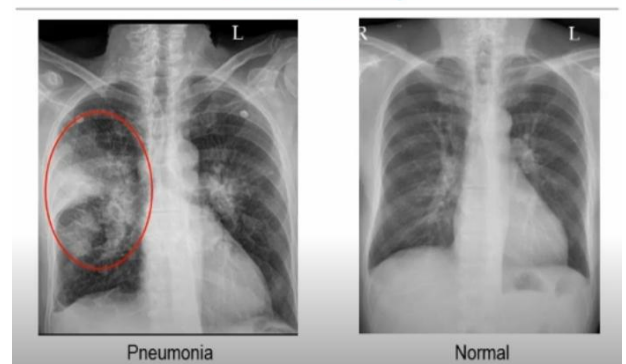
Pneumonia adalah peradangan paru-paru yang disebabkan oleh infeksi mikroorganisme seperti virus, bakteri, atau jamur. Selain itu, terdapat jenis baru dari *pneumonia* yang disebabkan oleh COVID-19. *Pneumonia* juga kerap dikenal dengan sebutan paru-paru basah. Pada kondisi ini, infeksi menyebabkan peradangan pada kantong-kantong udara (*alveoli*) di salah satu atau kedua paru-paru. Akibatnya, *alveoli* dipenuhi cairan atau nanah sehingga membuat penderitanya sulit bernapas.



Gambar 2.2.1. Ilustrasi *Pneumonia* Pada Paru-paru (Sumber: <https://www.alodokter.com/pneumonia>)

Pneumonia identik dengan adanya inflamasi pada satu atau kedua paru-paru. Inflamasi tersebut dapat terlihat pada citra rontgen (X-Ray) sebagai bercak-bercak putih yang ada di dalam paru-paru.

Chest x-ray

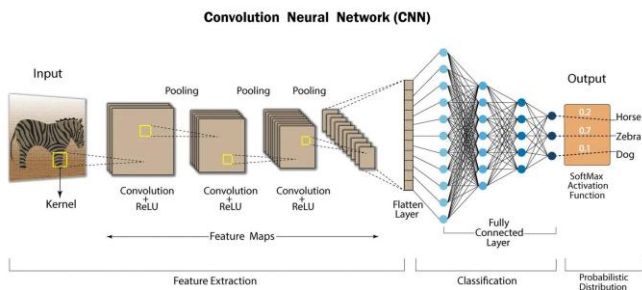


Gambar 2.2.2. Ilustrasi *Pneumonia* Pada Citra Rontgen Paru-paru (Sumber:

<https://www.radiologyinfo.org/en/info/pneumonia#8ffc413d83f64f049a4de596f859984d>)

C. *Convolutional Neural Network (CNN)*

Convolutional Neural Network (CNN) adalah algoritma pembelajaran mendalam yang populer, umumnya digunakan untuk memproses data yang memiliki topologi seperti *grid* seperti citra. *CNN* merupakan arsitektur jaringan untuk pembelajaran mendalam yang belajar langsung dari data, dengan menghilangkan kebutuhan untuk melakukan ekstraksi fitur secara manual.



Gambar 2.3.1. Arsitektur CNN (Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2024-2025/21-CNN-2024.pdf>)

CNN terdiri dari 3 lapisan utama, yaitu:

1. *Convolutional Layer* (+ ReLU)

Melakukan operasi konvolusi pada citra masukan dengan sejumlah penapis yang akan menghasilkan luaran yang disebut dengan *feature map*. Kemudian ditambah dengan layer ReLU yang lebih cepat dan efektif dalam memetakan nilai negatif ke nol dan mempertahankan nilai positif.

2. *Pooling Layer*

Berfungsi untuk mengurangi ukuran spasial dari matriks fitur hasil konvolusi guna mengurangi daya komputasi yang diperlukan untuk memproses data melalui pengurangan dimensi. Umumnya dibagi menjadi 2 jenis, yaitu *max pooling* (mengembalikan nilai maksimum dari kernel) dan *average pooling* (mengembalikan nilai rata-rata dari kernel).

3. *Fully-Connected Layer*

Berupa lapisan terakhir dalam CNN yang menghasilkan vektor berdimensi K (banyaknya kelas) yang berisikan probabilitas untuk setiap gambar yang diklasifikasikan.

III. IMPLEMENTASI

Pada Bab ini akan dijelaskan lebih lanjut mengenai pengimplementasian solusi.

A. *Pengumpulan dan Persiapan Data*

Data yang digunakan pada eksperimen ini didapat dari 2 sumber yang tersedia secara publik. Sumber pertama adalah Chest X-Ray Images (Pneumonia) yang tersedia pada website Kaggle, dengan distribusi data sebagai berikut:

Tabel 3.1.1. Distribusi Data Chest X-Ray Images (Pneumonia)

Test	Normal	234 data
	Pneumonia	390 data
Train	Normal	1341 data
	Pneumonia	3875 data
Val	Normal	8 data

	Pneumonia	8 data
--	-----------	--------

Sumber kedua adalah Chest X-ray (Covid-19 & Pneumonia), dengan distribusi data sebagai berikut:

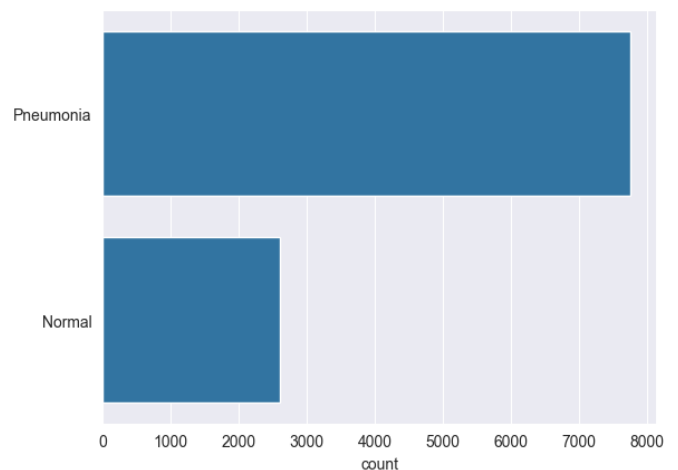
Tabel 3.1.2. Distribusi Data Chest X-ray (Covid-19 & Pneumonia)

Test	Covid	116 data
	Normal	317 data
	Pneumonia	855 data
Train	Covid	460 data
	Normal	1266 data
	Pneumonia	3418 data

Data dari kedua sumber tersebut digabungkan kedalam 3 bagian, yaitu *test*, *train*, dan *val*. Untuk setiap bagian tersebut dibagi lagi menjadi 2 jenis, yaitu normal dan pneumonia. Sehingga didapat data akhir dengan distribusi sebagai berikut:

Tabel 3.1.3. Distribusi Data Gabungan

Test	Normal	551 data
	Pneumonia	1361 data
Train	Normal	2607 data
	Pneumonia	7753 data
Val	Normal	8 data
	Pneumonia	8 data
Total		12288 data



Gambar 3.1.1. Distribusi Kelas Pada Data Latih

Namun data latih yang digunakan masih tidak seimbang antara data pneumonia dan data normalnya yang dimana data normal berjumlah jauh lebih sedikit disbanding data pneumonia. Sehingga perlu dilakukan augmentasi data untuk

menambah data normal agar jumlahnya seimbang dengan data pneumonia. Pendekatan yang digunakan untuk menambahkan data normal adalah menambahkan variasi dari data normal yang telah ada. Penambahan variasi tersebut juga dapat membuat model yang dikembangkan lebih fleksibel dalam melakukan prediksi, sehingga tidak ada tipe atau format untuk citra yang dapat digunakan. Variasi yang dilakukan adalah:

1. Melakukan rotasi kepada citra sebesar 30 derajat
2. Memperbesar (*zoom*) citra sebesar 20%
3. Menggeser citra sejauh 10% lebar secara horizontal
4. Menggeser citra sejauh 10% tinggi secara vertikal
5. Membalikkan gambar secara horizontal

Penambahan variasi tersebut dilakukan dengan menggunakan kode program sebagai berikut:

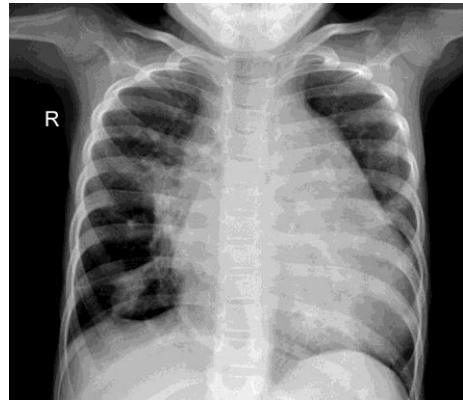
```
datagen = ImageDataGenerator(
    featurewise_center=False,          # set input mean to 0
    over the dataset
    samplewise_center=False,          # set each sample mean
    to 0
    featurewise_std_normalization=False, # divide inputs by
    std of the dataset
    samplewise_std_normalization=False, # divide each
    input by its std
    zca_whitening=False,              # apply ZCA whitening
    rotation_range = 30,              # randomly rotate images
    in the range (degrees, 0 to 180)
    zoom_range = 0.2,                 # Randomly zoom image
    width_shift_range=0.1,            # randomly shift images
    horizontally (fraction of total width)
    height_shift_range=0.1,          # randomly shift images
    vertically (fraction of total height)
    horizontal_flip = True,           # randomly flip images
    vertical_flip=False)              # randomly flip images
```

Sehingga distribusi data akhir dengan data latih yang seimbang adalah sebagai berikut:

Tabel 3.1.4. Distribusi Data Akhir

Test	Normal	551 data
	Pneumonia	1361 data
Train	Normal	7774 data
	Pneumonia	7753 data
Val	Normal	8 data
	Pneumonia	8 data
Total		17455 data

Berikut contoh citra untuk data pneumonia dan data paru-paru pasien normal:



Gambar 3.1.2. Citra Pneumonia dari Dataset



Gambar 3.1.3. Citra Normal dari Dataset

B. Pelatihan Model CNN

Model CNN yang digunakan dalam eksperimen ini memiliki 22 lapisan, yaitu:

Tabel 3.2.1. Arsitektur Model CNN yang Diusulkan

Layer	Output	Param
Conv2D	(None, 150, 150, 32)	320
BatchNormalization	(None, 150, 150, 32)	128
MaxPooling2D	(None, 75, 75, 32)	0
Conv2D	(None, 75, 75, 64)	18,496
Dropout	(None, 75, 75, 64)	0
BatchNormalization	(None, 75, 75, 64)	256
MaxPooling2D	(None, 38, 38, 64)	0
Conv2D	(None, 38, 38, 64)	36,928
BatchNormalization	(None, 38, 38, 64)	256
MaxPooling2D	(None, 19, 19, 64)	0
Conv2D	(None, 19, 19, 128)	73,856

Dropout	(None, 19, 19, 128)	0
BatchNormalization	(None, 19, 19, 128)	512
MaxPooling2D	(None, 10, 10, 128)	0
Conv2D	(None, 10, 10, 256)	295,168
Dropout	(None, 10, 10, 256)	0
BatchNormalization	(None, 10, 10, 256)	1024
MaxPooling2D	(None, 5, 5, 256)	0
Flatten	(None, 6400)	0
Dense	(None, 128)	819,328
Droupout	(None, 128)	0
Dense	(None, 1)	129

```
model.add(Dense(units = 1 , activation = 'sigmoid'))
model.compile(optimizer = "rmsprop" , loss = 'binary_crossentropy' , metrics = ['accuracy'])
```

Kemudian dilakukan pelatihan model CNN menggunakan data latih yang telah disiapkan sebelumnya. Pelatihan dilakukan sebanyak 12 pengulangan (*epoch*), dan dibantu dengan perubahan kecepatan belajar (*learning rate*) seiring dengan berkurangnya peningkatan performa selama pelatihan. Pelatihannya dilakukan dengan menggunakan kode program berikut:

```
learning_rate_reduction = ReduceLROnPlateau(monitor='val_accuracy', patience = 2, verbose=1, factor=0.3, min_lr=0.000001)
history = model.fit(x_train, y_train, batch_size = 32, epochs = 12, validation_data = datagen.flow(x_val, y_val), callbacks = [learning_rate_reduction])
```

Model tersebut diimplementasikan dengan menggunakan kode berikut:

```
model = Sequential()
model.add(Conv2D(32 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu' , input_shape = (150,150,1)))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Conv2D(64 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))
model.add(Dropout(0.1))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Conv2D(64 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Conv2D(128 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))
model.add(Dropout(0.2))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Conv2D(256 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))
model.add(Dropout(0.2))
model.add(BatchNormalization())
model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))
model.add(Flatten())
model.add(Dense(units = 128 , activation = 'relu'))
model.add(Dropout(0.2))
```

C. Pengevaluasian Kinerja Model

Pengevaluasian kinerja model dilakukan dengan kode program berikut:

```
predictions = model.predict(x_test)

# Calculate precision, recall, and thresholds
precision, recall, thresholds = precision_recall_curve(y_test, predictions)

# Calculate F1 scores for each threshold
f1_scores = 2 * (precision * recall) / (precision + recall)

# Find the index of the best F1 score
best_threshold_index = f1_scores.argmax()
best_threshold = thresholds[best_threshold_index]

# Apply the best threshold to classify the predictions
classes = (predictions > best_threshold).astype(int).flatten()
print("Optimal Threshold:", best_threshold)
print("Classified Classes:", classes)

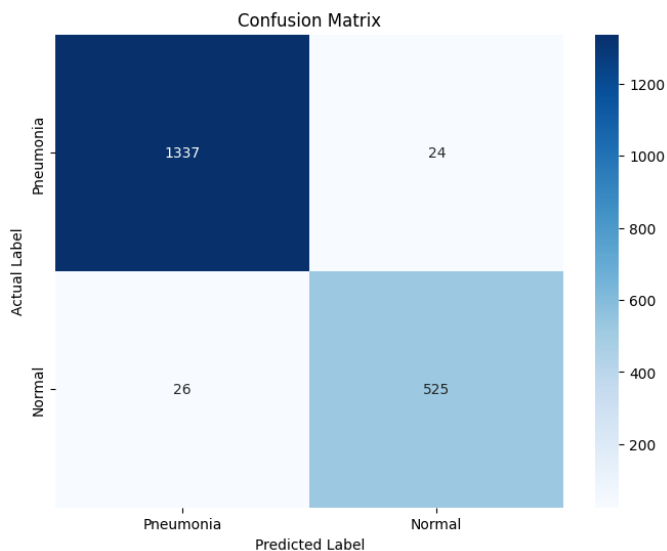
print(classification_report(y_test, classes, target_names = ['Pneumonia (Class 0)', 'Normal (Class 1)']))

cm = confusion_matrix(y_test, classes)
```

```
cm_df = pd.DataFrame(cm, index=['Pneumonia', 'Normal'],
columns=['Pneumonia', 'Normal'])
```

```
plt.figure(figsize=(8, 6))
sns.heatmap(cm_df, annot=True, fmt='g', cmap='Blues')
plt.title('Confusion Matrix')
plt.ylabel('Actual Label')
plt.xlabel('Predicted Label')
plt.show()
```

Berdasarkan model yang telah dibuat dan data yang telah disiapkan, sistem memiliki akurasi sebesar 97.07% menggunakan data validasi. Ketika diuji dengan menggunakan data uji, sistem mendapatkan akurasi sebesar 97% dengan matriks kebingungan (*confusion matrix*) dan laporan klasifikasi (*classification report*) sebagai berikut:



Gambar 3.3.1. Confusion Matrix Kinerja Model

	precision	recall	f1-score	support
Pneumonia (Class 0)	0.98	0.98	0.98	1361
Normal (Class 1)	0.96	0.95	0.95	551
accuracy			0.97	1912
macro avg	0.97	0.97	0.97	1912
weighted avg	0.97	0.97	0.97	1912

Gambar 3.3.2. Classification Report Kinerja Model

Berikut contoh citra-citra yang secara akurat diklasifikasikan oleh model CNN yang telah dikembangkan:



Gambar 3.3.3. Citra Pneumonia yang Terklasifikasikan Secara Akurat



Gambar 3.3.4. Citra Paru-paru Normal yang Terklasifikasikan Secara Akurat

IV. KESIMPULAN

Eksperimen yang telah dilakukan menunjukkan bahwa CNN dengan arsitektur yang diusulkan memiliki kinerja yang bagus dalam mendeteksi pneumonia berdasarkan citra rontgen. Meskipun model telah memiliki akurasi yang tergolong bagus, yaitu sebesar 97%, model masih memiliki potensi yang dapat ditelusuri untuk meningkatkan akurasinya lebih lanjut. Salah satu caranya adalah dengan menggunakan dataset yang lebih besar dengan variasi yang lebih banyak. Dengan dataset yang lebih bervariasi dan lebih besar tersebut, model tidak hanya dapat meningkatkan akurasinya, melainkan juga memperbanyak kasus-kasus citra rontgen yang dapat ditanganinya.

LINK GITHUB

<https://github.com/Ainzw0rth/PneumoniaCNN.git>

LINK VIDEO YOUTUBE

<https://youtu.be/MF5E13hOMTI>

UCAPAN TERIMA KASIH

Penulis mengucapkan Syukur kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat yang diberikan-Nya, penulis dapat menyelesaikan tugas makalah ini dengan baik. Penulis juga ingin berterima kasih kepada Pak Rinaldi Munir selaku dosen pengampuh mata kuliah Pemrosesan Citra Digital IF4073 yang telah memberikan materi dan pengetahuan untuk Menyusun makalah ini. Selain itu, penulis juga ingin berterima kasih kepada keluarga dan teman-teman penulis yang telah memberikan dukungan selama proses pengerjaan dan penulisan makalah ini.

REFERENSI

- [1] Al Reshan, M. S., Gill, K. S., Anand, V., Gupta, S., Alshahrani, H., & Sulaiman, A. (2023). Detection of pneumonia from chest X-ray images utilizing MobileNet model. *Healthcare*, 11(11), 1561. <https://doi.org/10.3390/healthcare11111561>
- [2] Alodokter. (n.d.). Pneumonia. Diakses pada 14 Januari 2025, dari <https://www.alodokter.com/pneumonia>
- [3] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Cell*, 168(5), 1101–1110.e10. <https://doi.org/10.1016/j.cell.2017.01.035>
- [4] Madz2000. (n.d.). Pneumonia Detection using CNN (92.6% accuracy). Diakses pada 14 Januari 2025, dari <https://www.kaggle.com/code/madz2000/pneumonia-detection-using-cnn-92-6-accuracy>
- [5] Munir, R. (2024). Convolutional Neural Network (CNN). Diakses pada 14 Januari 2025, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2024-2025/21-CNN-2024.pdf>
- [6] Munir, R. (2024). Image Enhancement Bagian 1. Diakses pada 14 Januari 2025, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2024-2025/08-Image-Enhancement-Bagian1-2024.pdf>
- [7] Munir, R. (2024). Pembentukan Citra. Diakses pada 14 Januari 2025, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2024-2025/03-Pembentukan-Citra-dan-Digitalisasi-Citra-2024.pdf>
- [8] Munir, R. (2024). Pengantar Pemrosesan Citra Digital. Diakses pada 14 Januari 2025, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2024-2025/01-Pengantar-Pemrosesan-Citra-Digital-Bag1-2024.pdf>
- [9] Mooney, P. T. (n.d.). Chest X-Ray Images (Pneumonia). Diakses pada 14 Januari 2025, dari <https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia>
- [10] Prashant. (n.d.). Chest X-Ray: COVID-19 & Pneumonia Dataset. Diakses pada 14 Januari 2025, dari <https://www.kaggle.com/datasets/prashant268/chest-xray-covid19-pneumonia>
- [11] RadiologyInfo.org. (n.d.). Pneumonia. Diakses pada 14 Januari 2025, dari <https://www.radiologyinfo.org/en/info/pneumonia%20that%20identify%20an%20infection>.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 15 Januari 2024



Louis Caesa Kesuma (13521069)