

Identifikasi Pneumonia Pada Anak Berbasis Citra X-Ray Paru-Paru Menggunakan Metode *Convolutional Neural Network* (CNN)

Natasya Umi Fitria¹, Yana Adharani^{1*}, Popy Meilina¹, Rita Dewi Risanty¹

¹Teknik Informatika, Universitas Muhammadiyah Jakarta, Jl. Cempaka Putih Tengah 27, Jakarta Pusat, 10510

* Corresponding Author: yana.adharani@umj.ac.id

Abstrak

Pneumonia merupakan salah satu penyebab utama kematian anak balita di seluruh dunia. Penyakit ini disebabkan oleh bakteri, virus, atau jamur yang menyebabkan paru-paru terisi nanah dan cairan sehingga menyulitkan anak-anak dalam bernafas. Organisasi Kesehatan Dunia (WHO) menyebutkan 99% kematian anak akibat *pneumonia* terjadi di negara berkembang. Salah satu cara pengidentifikasi penyakit *pneumonia* adalah dengan melakukan *rontgen* x-ray. Hasil *rontgen* x-ray termasuk citra kompleks sehingga diperlukan metode yang sesuai untuk melakukan identifikasi. *Convolutional Neural Network* (CNN) merupakan salah satu metode dalam *deep learning* yang dapat digunakan untuk melakukan klasifikasi. Pada penelitian ini dilakukan klasifikasi terhadap hasil *rontgen* x-ray paru-paru anak dengan menggunakan CNN. Klasifikasi dibagi kedalam dua kelas, yaitu terindikasi *pneumonia* dan normal. Hasil uji coba menunjukkan CNN dapat digunakan untuk melakukan klasifikasi apakah seorang anak terindikasi *pneumonia* atau tidak berdasarkan hasil *rontgen* x-ray paru-paru anak dengan tingkat akurasi 0.9632 dan *loss* 0.1139.

Kata kunci: *Pneumonia, Anak, Rontgen X-Ray, Paru-Paru, CNN*

Abstract

Pneumonia is one of the leading causes of *toddler's* mortality in the world. The disease caused by bacteria, viruses, or fungi that cause the lungs to fill with pus and fluid, making children difficult to breathe. World Health Organization (WHO) states that 99% of child deaths due to *pneumonia* occur in developing countries. One of the strategies to identify *pneumonia* is by performing an x-ray. X-ray results is complex images, so that an appropriate method is needed to identify them. *Convolutional Neural Network* (CNN) is one of deep learning method that can be used to perform classification. In this study, classification was carried out on the results of x-rays of children's lungs using CNN. The classification is divided into two classes, indicated *pneumonia* and normal. The results showed that CNN can be used to classify whether a child is indicated *pneumonia* or not based on the results of the child's lung x-ray with an accuracy level of 0.9632 and a loss of 0.1139.

Keywords : *Pneumonia, Child, X-ray, Lungs, CNN*

PENDAHULUAN

1. Pendahuluan

Penyakit *pneumonia* merupakan penyebab utama kematian anak balita. Penyakit ini disebabkan oleh bakteri, virus atau jamur yang mengakibatkan anak-anak sulit bernafas karena paru-paru terisi nanah dan cairan. *Pneumonia* ditandai dengan gejala batuk dan

atau kesulitan bernapas seperti napas cepat dan tarikan dinding dada bagian bawah ke dalam. Berdasarkan data WHO, 99% kematian *pneumonia* anak terjadi dinegara berkembang (Kementerian Kesehatan RI, 2015). Menurut Kevin Watkins, Chief Executive of *Save the Children*, *pneumonia* sebagai epidemi global yang terabaikan yang perlu segera ditindaklanjuti, nyawa jutaan anak terancam

karena ketiadaan vaksin, antibiotic dan pengobatan dengan oksigen secara teratur (UNICEF, 2019).

Berdasarkan Riset Kesehatan Dasar (Riskesdas) melaporkan bahwa kejadian *pneumonia* sebulan terakhir mengalami peningkatan pada tahun 2007 sebesar 2,1% menjadi 27% pada tahun 2013 (Anwar & Dharmayanti, 2014). Diperkirakan 1,8 juta atau 20% dari kematian anak diakibatkan karena *pneumonia*. Dari laporan WHO tahun 2017, 15% dari kematian anak dibawah 5 tahun atau 5,5 juta disebabkan *pneumonia*. Menurut (Henrietta Fore, Executive Director UNICEF) setiap hari, terdapat 2.200 anak balita yang meninggal akibat *pneumonia*. Di Indonesia pada tahun 2018 diperkirakan sekitar 19.000 anak meninggal dunia akibat *pneumonia* (UNICEF, 2019).

Kematian *pneumonia* dapat dicegah dengan meningkatkan akses dan kualitas pelayanan kesehatan pada balita, disamping itu juga keluarga berperan aktif dalam kesehatan anak dengan cara memberikan ASI eksklusif 6 bulan menyusui ditambah MPASI selama 2 tahun, imunisasi lengkap pada anak, kecukupan gizi seimbang dan menerapkan pola hidup bersih (Kementerian Kesehatan RI, 2020). Penyakit *pneumonia* dapat dicegah dengan vaksin dan menggunakan antibiotik apabila didiagnosis secara cepat. Diagnosis *pneumonia* pada anak dapat melibatkan pemeriksaan fisik seperti *rontgen* x-ray dada dan pemeriksaan laboratorium, namun masih terdapat kesulitan dalam mengidentifikasi *pneumonia* hasil dari *rontgen* dada sehingga diperlukannya identifikasi dalam hasil *rontgen* untuk membantu para ahli dokter memberikan diagnosis yang jelas (Unair, 2023).

Sebelumnya telah dilakukan beberapa penelitian terkait identifikasi kondisi paru-paru, sebagai contoh pada penelitian (Putri, Diana, & Purnamasari, 2020) menggunakan metode K-NN untuk mengidentifikasi penyakit paru-paru berdasarkan tingkat kemiripan dengan mendapatkan hasil akurasi sebesar 96.96%, kemudian pada penelitian yang dilakukan oleh (Yopento, Ernawati, & Coastera, 2022) menggunakan metode CNN berdasarkan ekstraksi fitur sobel pada citra x-ray paru-paru guna mendeteksi *pneumonia* atau tidak, dimana dalam penelitian ini

mendapatkan hasil akurasi sebesar 92.59%, dengan metode CNN terdapat peluang untuk meningkatkan hasil *akurasi*. Pada penelitian serupa juga dilakukan oleh Juliandri Estomihi menggunakan metode *Backpropagation* mendapatkan hasil akurasi sebesar 93.3%.

Pada penelitian ini, akan dilakukan identifikasi *pneumonia* dengan metode CNN. Penggunaan metode CNN dipilih karena berdasarkan penelitian sebelumnya metode ini cukup efektif dalam mengekstraksi fitur dari citra medis kompleks seperti CT scans atau radiografi paru-paru. Oleh karena itu, dalam penelitian ini akan dilakukan identifikasi *pneumonia* pada anak berbasis citra x-ray menggunakan Metode *Convolutional Neural Network (CNN)*”.

Pneumonia

Pneumonia merupakan infeksi jaringan paru-paru (alveoli) yang bersifat akut. Penyakit ini bisa disebabkan oleh bakteri dan virus yang ada di lingkungan sekitar, bagi yang terkena *pneumonia* sulit untuk bernafas karena paru-parunya berisi nanah dan cairan (UNICEF, 2020). Salah satu gejala *pneumonia* yaitu menggigil, demam, batuk dan sesak nafas. *Pneumonia* tidak hanya menyerang orang tua bahkan banyaknya menyerang anak-anak dan anak dibawah usia 5 tahun.

Pneumonia merupakan penyebab utama kematian pada anak (usia < 5 tahun) atau pada bayi baru lahir (usia < 28 hari) bahkan bisa terjadi dalam beberapa jam setelah kelahiran (Hasanah & Santik, 2021). Pada anak-anak atau bayi baru lahir kemungkinan terinfeksi *pneumonia* semakin tinggi jika terdapat faktor risiko besar, seperti diakibatkan dari proses yang terjadi saat kehamilan, saat proses persalinan (Meizikri, Yani, & Yusrawati, 2016).

Gangguan Pernafasan

Gangguan pernafasan adalah gangguan yang terjadi pada proses pengikatan oksigen pada saluran pernafasan. Gangguan pernafasan disebabkan oleh bakteri, virus, kuman dan zat lain yang berkaitan dengan *hemoglobin*. Penyakit gangguan pernafasan tidak hanya menyerang orang dewasa, perokok melainkan bisa menyerang terhadap anak-anak atau bayi yang baru lahir. Gangguan pernafasan yang

umum terjadi pada anak salah satunya seperti RDS, *pneumonia neonatal*, yang diakibatkan karena komplikasi (Reuter, Moser, & Baack, 2014).

Pre-Processing

Pre-processing merupakan suatu teknik awal dalam data mining untuk mengambil informasi yang ada dengan cara membersihkan dan menggabungkan data-data tersebut, sehingga menjadi data yang lebih bersih untuk dapat dilakukan pengolahan berikutnya (Nurvinda, 2021). Dalam citra digital, *Pre-processing* merupakan suatu proses untuk menghilangkan *noise* atau bagian yang tidak diperlukan pada gambar. *Pre-processing* merupakan tahapan awal dalam pengenalan karakter (Rohpandi, Sugiharto, & Winara, 2015). *Pre-procesing terdiri dari grayscale image, thresholding image, segmentation, data transformation, dan augmentation image.*

1. Greyscale image

Grayscale adalah sebuah citra yang memiliki warna hitam, keabuan, dan putih. *Grayscale* merupakan hasil konversi dari citra digital yang mempunyai 3 komponen (RED = GREEN = BLUE) untuk menunjukkan tingkat intensitas. Citra grayscale mempunyai nilai intensitas setiap pixel yang berkisar antara 0 sampai dengan 255. Nilai 0 menyatakan hitam dan nilai 255 menyatakan putih. Proses ini akan mengecilkan range warna citra menjadi 0 sampai 255. Proses ini akan memudahkan ketika akan melakukan thresholding citra menjadi citra biner (Hibatullah & Maliki, 2018).

2. Thresholding image

Thresholding merupakan suatu proses dimana hasilnya berupa citra biner dari citra grayscale atau citra berwarna dengan mengatur nilai piksel ke nilai 0 atau 1 dimana *thresholding* sebuah citra digital yang hanya memiliki dua kemungkinan warna pixelnya hitam dan putih dari nilai ambang batasnya, dalam gambar *grayscale*, *thresholding* dapat digunakan untuk memisahkan objek yang terang dari latar belakang yang gelap atau sebaliknya (Hibatullah & Maliki, 2018). Biasanya pixel object diberi nilai 1 sementara pixel background diber inilai 0. Proses awal

yang banyak dilakukan dalam image processing adalah mengubah citra berwarna menjadi citra grayscale, hal ini digunakan untuk menyederhanakan model citra.

3. Segmentation

Segmentasi merupakan proses pembagian sebuah citra menjadi daerah pilihan dari citra secara keseluruhan. Hal-hal yang mempengaruhi proses segmentasi agar bisa dilakukan yaitu tekstur, kecerahan, serta intensitas jumlah pixel. Teknik *segmentasi* yaitu dengan membagi citra menjadi beberapa bagian untuk mengetahui batasannya, sedangkan teknik selanjutnya yaitu dengan memberi index terhadap warna tiap piksel yang menunjukkan bagian dalam suatu segmentasi (Bhahri & Rachmat, 2018).

4. Data Transformasi

Data transformasi adalah proses mengubah data dari satu bentuk atau format ke bentuk atau format lainnya. *Transformasi* data seringkali dilakukan dalam konteks analisis data atau pengolahan data untuk memungkinkan analisis yang lebih efektif, visualisasi, atau penyimpanan data yang lebih baik. Pada tahap ini, data akan dinormalisasi dan digeneralisasikan (Oliver, 2023). *Normalisasi* adalah salah satu teknik persiapan data yang paling sering digunakan. Pada machine learning dan data mining, proses ini membantu mengubah nilai kolom numerik dalam dataset untuk menggunakan skala umum.

5. Augmentation Image

Augmentation Image merupakan teknik mengubah data yang ada untuk membuat lebih banyak data saat proses pelatihan model. *Image augmentation* suatu proses menciptakan contoh pelatihan baru yang sudah ada dengan membuat sampel terbaru. Dalam *augmentation image* terdiri atas *resizing* atau *rescale*, *flipping* dan *cropping*. (Laraswati, 2023).

Rescale merupakan salah satu bagian dari *image augmentation* yang berfungsi untuk menormalisasikan nilai setiap pixel pada gambar menjadi nilai 0 sampai 1, dalam artian jika gambar diperbesar maka faktor skala harus lebih besar dari 1, jika gambar diperkecil maka faktor skala harus antara 0 dan 1. Nilai pixel

dalam rentang [0.255], maka setelah rescale perlu memastikan bahwa rentan nilai pixel tetap berada dalam [0.255]. dapat diketahui dengan perumusan:

$$\text{Pixelbaru} = \text{pixellama} \times \text{faktorskala} \quad (1)$$

Dengan keterangan:

Pixel_lama : nilai pixel asli pada gambar

Factor_baru : faktor skala yang dipilih

Flipping merupakan salah satu bagian dari *image augmentation* yang berfungsi untuk membalikan posisi secara vertical maupun horizontal.

Cropping atau proses pemotongan merupakan salah satu dari *image augmentation* yang dilakukan secara random, namun diberikan parameter *size* untuk menentukan saat melakukan *crop*.

Convolution Neural Network (CNN)

Convolution Neural Network (CNN) adalah pengembangan dari Multilayer Perceptron (MLP) yang didesain untuk mengolah data dua dimensi (Suartika E. P, Wijaya, & Soelaiman, 2016). *Convolution Neural Network (CNN)* merupakan salah satu jenis *neural network* yang digunakan pada data *image*. *Convolution Neural Network (CNN)* juga bisa digunakan untuk mendeteksi dan mengenali object pada sebuah *image*. *Convolution Neural Network (CNN)* ini terdiri dari neuron yang memiliki weight, bias dan *activation function*. *Convolution Neural Network (CNN)* memanfaatkan proses konvolusi dengan mengerjakan sebuah kernel konvolusi (filter) berukuran tertentu ke sebuah gambar (Lina, 2019).

Secara garis besar, tahapan CNN dibagi menjadi dua fase, yaitu *features learning* dan *classification*. Alur proses pertama adalah input data, lalu diproses dengan *convolution* + *ReLU1*, lapisan ini memiliki beberapa filter yang digeser untuk mengekstrak fitur lanjutan. Kemudian proses *pooling1* dimana nilai maksimum dihitung dan diambil dari *feature map*. Lalu mengulang pada proses *convolution* + *ReLU2* kemudian kembali ke proses *pooling2* untuk mengurangi dimensi *feature map*. Dari hasil *pooling* terakhir dilanjutkan

proses *classification*, yaitu *flatten*, *fully connected* dan *softmax*. Pada proses *flatten image* dipecah menjadi *image* yang sudah diolah dan di *filter* dari tahap *feature learning*.

Pada *convolution neural network* terdapat *dense* dan *dropout*. *Dense* atau disebut juga *Fully Connected Layer* merupakan suatu lapisan yang terdiri dari sejumlah besar *neuron* atau unit yang saling terhubung penuh. Setiap *neuron* dalam lapisan *dense* menerima masukan dari tiap *neuron* dalam lapisan sebelumnya. Ini berarti setiap *neuron* dalam lapisan *dense* terhubung ke semua *neuron* dalam lapisan sebelumnya.

Dropout adalah teknik regulasi yang bertujuan untuk mengurangi kompleksitas model dengan tujuan mencegah *overfitting*. *Dropout* hanya diterapkan selama pelatihan, dengan mengubah skala aktivitas *neuron*. Pada *dropout* ini ditentukannya dengan angka 0,5 untuk mengindikasikan bahwa sekitar 50% dari *neuron* dalam lapisan sebelumnya akan dinonaktifkan secara acak selama pelatihan. Dengan perumusan:

$$\text{Output} = x * \text{mask} \quad (2)$$

x : input ke fungsi ReLU

Mask : hasil keluaran nilai acak yang menentukan unit akan dinonaktifkan atau tidak.

Mask akan bernilai 1 jika unit tetap aktif atau 0 jika unit dinonaktifkan.

Pada tahapan *Convolution Neural Network* memiliki beberapa layer yang berfungsi untuk melakukan *filter* pada setiap prosesnya yang disebut proses *training*. Proses *training* terdapat beberapa tahap yaitu *convolution layer*, *pooling layer* (Santoso & Ariyanto, 2018).

ReLU

Aktivasi ReLU merupakan lapisan aktivasi pada model *convolutional neural network* yang dapat mengaplikasikan fungsi *thresholding* dengan nilai nol terhadap nilai pixel input. Fungsi ReLU akan mengubah nilai negative menjadi nol atau ReLU akan mengembalikan nilai 0 jika hasil negative dan jika didapatkan lebih dari 0 maka nilai tersebut menghasilkan nilai itu sendiri, dengan rumus:

$$\text{ReLU}(x) = \max(0, x) \quad (3)$$

Dengan keterangan:

x : input ke fungsi ReLU

$\text{ReLU}(x)$: hasil keluaran

Dapat dikatakan $\text{ReLU}(x)$ menghasilkan nilai 0 jika x kurang dari atau sama dengan 0 dan akan menghasilkan nilai x itu sendiri jika $x > 0$.

Softmax

Fungsi *Softmax* menghitung probabilitas dari setiap kelas target atas semua kelas target yang memungkinkan dan akan membantu untuk menentukan kelas target untuk input yang diberikan. Keuntungan utama menggunakan *Softmax* adalah rentang probabilitas output dengan nilai 0 hingga 1, dan jumlah semua probabilitas akan sama dengan satu. *Softmax* menggunakan eksponensial dari nilai input yang diberikan dan jumlah nilai eksponensial dari semua nilai dalam input. Maka rasio eksponensial dari nilai input dan jumlah nilai eksponensial adalah output dari fungsi softmax.

Softmax function mengonversi nilai-nilai numerik menjadi probabilitas. Maka fungsi *softmax* dengan perumusan:

$$s(o_i) = \frac{e^{o_i}}{\sum_{j=1}^n e^{o_j}} \quad (4)$$

Dengan keterangan:

$s(o_i)$ adalah output ke- i dari softmax

n adalah jumlah kelas.

Confusion Matriks

Confusion matriks merupakan suatu model yang digunakan untuk menampilkan hasil pengklasifikasian dengan mudah. *Confusion matriks* digunakan untuk menampilkan performa dari model-model algoritma lain dan juga untuk melakukan perhitungan akurasi pada konsep data mining. Dalam *confusion matriks* terdapat beberapa parameter yang digunakan sebagai pengatur performa, yaitu menghitung tingkat akurasi dan tingkat *loss*. Berikut penjelasan perhitungan *confusion matrix* mencari akurasi dan *loss*.

		Actual Values	
		1 (Positive)	0 (Negative)
Predicted Values	1 (Positive)	TP (True Positive)	FP (False Positive) Type I Error
	0 (Negative)	FN (False Negative) Type II Error	TN (True Negative)

Gambar 1. *Confusion matrix*

Pada Gambar 1 *confusion matrix* terdapat *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)* dan *False Negative (FN)*.

- True Positive (TP)*: Merupakan data positif yang diprediksi benar.
- True Negative (TN)*: Merupakan data negatif yang diprediksi benar.
- False Positive (FP)/ Type I Error*: Merupakan data negatif namun diprediksi sebagai data positif.
- False Negative (FN)/ Type II Error*: Merupakan data positif namun diprediksi sebagai data negatif.

Confusion matrix dapat dilakukan untuk mengukur kinerja model *convolution neural network*. Pada pengujiannya dilakukan untuk melihat akurasi dan *loss*.

- Akurasi : seberapa akuratnya model untuk klasifikasi dengan benar.

$$\text{accuracy} = \frac{TP + TN}{(TP + FP + FN + TN)} \quad (5)$$

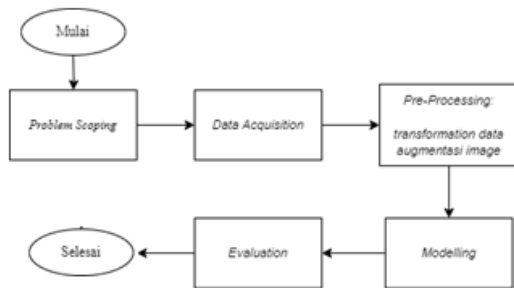
- Loss*

$$\text{loss} = -\frac{1}{N} + \sum_{i=1}^N (y_i \log(p_i) + (1 - y_i) \log(1 - p_i)) \quad (6)$$

N adalah jumlah sampel, y_i adalah label aktual dan p_i adalah probability prediksi.

METODE

Tahapan penelitian dilakukan sebagaimana tertera dalam Gambar 2.



Gambar 2. Tahapan penelitian

Problem Scoping

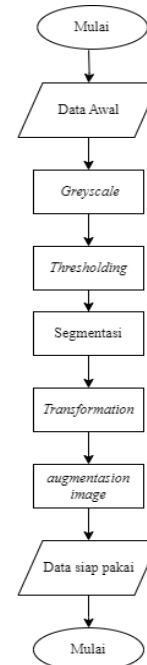
Permasalahan yang diangkat mengenai identifikasi kondisi paru-paru pada anak berdasarkan hasil *rontgen x-ray*. Penelitian ini dilakukan untuk melihat kondisi paru-paru pada anak yang rentan terhadap gangguan pernafasan *pneumonia* yang dapat menyebabkan kematian.

Data Acquisition

Penelitian ini menggunakan data yang didapatkan dari *kaggle* yaitu *image* hasil x-ray paru-paru anak sebanyak 1354 data, yang terdiri dari 624 data *test* (normal dan *pneumonia*) dan 730 data *train* (normal dan *pneumonia*). Data yang digunakan berupa data *image* yang terbagi menjadi data paru-paru normal dan data paru-paru *pneumonia*.

Pre-processing

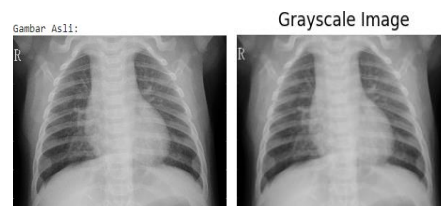
Tahapan *pre-processing* data:

Gambar 3. Tahapan *pre-processing*

Berdasarkan Gambar 3, *preprocessing* terdiri dari 5 tahap, yaitu *grayscale image*, *thresholding*, *segmentasi*, *transformation*, dan *augmentation image*.

1. Greyscale Image

Langkah awal yang perlu dilakukan dalam mengidentifikasi *pneumonia* pada anak adalah dengan merubah citra hasil *rontgen x-ray* ke dalam bentuk *greyscale* seperti pada gambar 4.

Gambar 4. *Grayscale image*

2. Thresholding Image

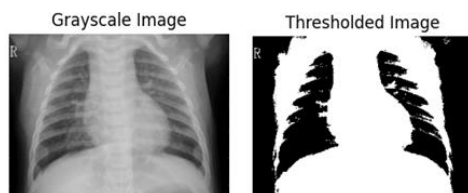
Setelah diubah dalam bentuk *grayscale* selanjutnya *thresholding*, *thresholding* digunakan untuk memisahkan objek yang terang dari latar belakang yang gelap atau sebaliknya, *thresholding* memiliki matriks yang sama ukurannya dengan citra masukan, di mana setiap piksel telah diubah menjadi 0 atau 255 seperti pada *code* berikut:

```
def thresholding(image):
    _, thresholded_image = cv2.threshold(image, 127, 255, cv2.THRESH_BINARY)
    return thresholded_image
```

Thresholding dengan ambang 127, piksel-piksel yang lebih besar dari 127 akan menjadi 255, dan yang lebih kecil atau sama dengan 127 akan menjadi 0. Contoh dalam bentuk matriks:

Matriks piksel 4x4:		Matriks piksel hasil thresholding:
[[137 142 103 100]]	→	[[255 255 0 0]]
[138 170 139 110]]		[255 255 255 0]]
[146 148 174 144]]		[255 255 255 255]]
[[88 182 187 140]]		[[0 255 255 255]]

Matriks ini didapatkan dari gambar yang digunakan dan hasil seperti pada gambar 5.



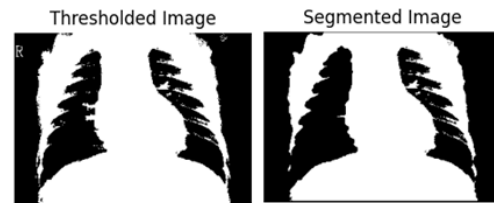
Gambar 5. *Thresholding image*

3. *Segmentation image*

Tahap ini membagi gambar menjadi beberapa segmen atau wilayah yang memiliki kesamaan karakteristik, dalam pengenalan objek, segmentasi dapat mengidentifikasi dan memisahkan objek dalam gambar dari latar belakangnya, dimana menggunakan parameter -1 yang menandakan bahwa semua kontur yang ditemukan digambar. Menggunakan (0, 255, 0) untuk menggambarkan kontur pada gambar seperti pada *code* berikut:

```
def segmentation(image):
    contours_ = cv2.findContours(image, cv2.RETR_EXTERNAL,
    cv2.CHAIN_APPROX_SIMPLE)
    segmented_image = cv2.drawContours(image.copy(), contours, -1, (0, 255,
    0), 2)
    return segmented_image
```

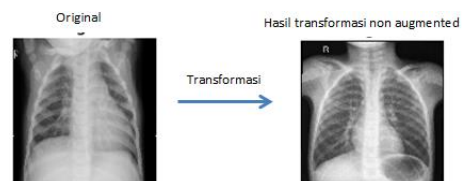
Hasil *segmentation image* dapat dilihat pada gambar 6.



Gambar 6. Hasil *segmentation*

4. *Data Transformasi*

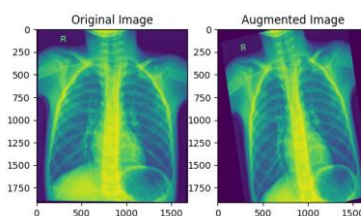
Pada tahap ini dilakukan normalisasi hasil *rontgen* paru-paru anak dengan *rescale:1/255*. Proses *transformasi* dilakukan terhadap 734 data *train*, dan 624 data *test* dengan mengubah *target_size* dan *batch_size*. Contoh hasil *transformasi* citra paru-paru anak dapat dilihat pada gambar 7.



Gambar 7. Hasil transformasi paru-paru anak

5. *Augmentation Image*

Setelah *image rontgen* paru-paru anak di normalisasi selanjutnya untuk perbaikan data model dilakukan *augmentasi image* dengan rotasi range 45° dan 50°. Untuk zoom dilakukan 2 kali percobaan dengan besaran zoom yang berbeda-beda yaitu 20% dan 30%. Contoh hasil *augmentation image* untuk hasil *rontgen* x-ray anak pada paru-paru normal yang dijadikan sebagai data *train* dapat dilihat pada gambar 8.



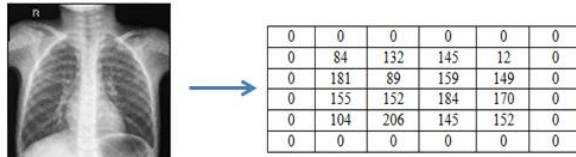
Gambar 8. Hasil *augmentation image*

Modelling

1. *Convolution Neural Network*

Pada proses *modelling convolution neural network* dilakukan perhitungan *convolution*

layer hingga *polling layer*. Berikut proses konvolusi dengan memberikan nilai filter pada matrix. Pixel citra ini dengan 1 *channel* yaitu *grey* dari data asli, diambil data pixelnya masing-masing dan ditambahkan dengan padding 0 disetiap pixelnya sehingga didapatkan pixel seperti gambar 9.



Gambar 9. Nilai *pixel dataset* paru-paru yang normal

Pada proses percobaan ini menggunakan kernel 3x3 dengan nilai pada gambar 10.

1	0	-1
1	0	-1
1	0	-1

Gambar 10. Kernel 3x3

Selanjutnya dilakukan proses perhitungan dengan mengkalikan hasil pixel pada gambar 10 dengan kernel 3x3 pada gambar 11. Tahapan ini dilakukan berulang dengan pergeseran kernel sebanyak 1 *strides* setiap *channel*nya sehingga didapatkan perhitungan dengan nilai sebagai berikut:

0	0	0	0	0	0
0	84	132	145	12	0
0	181	89	159	149	0
0	155	152	184	170	0
0	104	206	145	152	0
0	0	0	0	0	0

Posisi 1
 $(1*0)\pm(0*0)+((-1)*0)+$
 $(1*0)\pm(0*84)+((-1)*132)+$
 $(1*0)\pm(0*181)+((-1)*89) = -221$

0	0	0	0	0	0
0	84	132	145	12	0
0	181	89	159	149	0
0	155	152	184	170	0
0	104	206	145	152	0
0	0	0	0	0	0

Posisi 2
 $(1*0)\pm(0*0)+((-1)*0)+$
 $(1*84)\pm(0*132)+((-1)*145)+$
 $(1*181)\pm(0*89)+((-1)*159) = -39$

0	0	0	0	0	0
0	84	132	145	12	0
0	181	89	159	149	0
0	155	152	184	170	0
0	104	206	145	152	0
0	0	0	0	0	0

Posisi 3
 $(1*0)\pm(0*0)+((-1)*0)+$
 $(1*132)\pm(0*145)+((-1)*12)+$
 $(1*89)\pm(0*159)+((-1)*149) = 60$

0	0	0	0	0	0
0	84	132	145	12	0
0	181	89	159	149	0
0	155	152	184	170	0
0	104	206	145	152	0
0	0	0	0	0	0

Posisi 4
 $(1*0)\pm(0*0)+((-1)*0)+$
 $(1*145)\pm(0*12)+((-1)*0)+$
 $(1*159)\pm(0*149)+((-1)*0) = 304$

0	0	0	0	0	0
0	84	132	145	12	0
0	181	89	159	149	0
0	155	152	184	170	0
0	104	206	145	152	0
0	0	0	0	0	0

Posisi 12
 $(1*159)\pm(0*149)+((-1)*0)+$
 $(1*184)\pm(0*170)+((-1)*0)+$
 $(1*145)\pm(0*152)+((-1)*0) = 488$

0	0	0	0	0	0
0	84	132	145	12	0
0	181	89	159	149	0
0	155	152	184	170	0
0	104	206	145	152	0
0	0	0	0	0	0

Posisi 13
 $(1*0)\pm(0*155)+((-1)*152)+$
 $(1*0)\pm(0*104)+((-1)*206)+$
 $(1*0)\pm(0*0)+((-1)*0) = -358$

0	0	0	0	0	0
0	84	132	145	12	0
0	181	89	159	149	0
0	155	152	184	170	0
0	104	206	145	152	0
0	0	0	0	0	0

Posisi 14

$$(1*155)\pm(0*152)+((-1)*184)+$$

$$(1*104)\pm(0*206)+((-1)*145)+$$

$$(1*0)\pm(0*0)+((-1)*0) = -70$$

0	0	0	0	0	0
0	84	132	145	12	0
0	181	89	159	149	0
0	155	152	184	170	0
0	104	206	145	152	0
0	0	0	0	0	0

Posisi 15

$$(1*152)\pm(0*184)+((-1)*170)+$$

$$(1*206)\pm(0*145)+((-1)*152)+$$

$$(1*0)\pm(0*0)+((-1)*0) = 36$$

0	0	0	0	0	0
0	84	132	145	12	0
0	181	89	159	149	0
0	155	152	184	170	0
0	104	206	145	152	0
0	0	0	0	0	0

Posisi 16

$$(1*184)\pm(0*170)+((-1)*0)+$$

$$(1*145)\pm(0*152)+((-1)*0)+$$

$$(1*0)\pm(0*0)+((-1)*0) = 329$$

Gambar 11. Perhitungan konvolusi layer

Setelah perhitungan pergeseran konvolusi layer diatas, didapatkan hasil dari tiap posisinya seperti pada gambar 12.

-221	-39	60	304
-373	-68	42	488
-447	-48	-24	488
-358	-70	36	329

Gambar 12. Hasil perhitungan konvolusi layer

Hasil dari perhitungan konvolusi layer ini kemudian diubah dengan fungsi *activation* RELU, dimana setiap nilai yang negative diubah menjadi angka 0 sehingga didapatkan hasil seperti pada gambar 13.

-221	-39	60	304
-373	-68	42	488
-447	-48	-24	488
-358	-70	36	329

$$f(x) = \max(0, x)$$

0	0	60	304
0	0	42	488
0	0	0	488
0	0	36	329

Gambar 13. Hasil *convolutional 1*

Proses ini dilakukan secara berulang sampai semua citra dengan menggunakan 32 filter yang berbeda sehingga *output* dari tahapan konvolusi akan menghasilkan banyak *feature map*.

2. Pooling Layer

Output tahap konvolusi akan dijadikan sebagai *input pooling layer*, pada tahap ini menggunakan *maxpooling* dengan kernel 2x2, dimana yang diambil nilai terbesar. Proses *maxpooling* hasil diatas ditunjukkan pada gambar 14.

0	0	60	304
0	0	42	488
0	0	0	488
0	0	36	329

$$\rightarrow$$

0	488
0	488

0	0	60	304
0	0	42	488
0	0	0	488
0	0	36	329

$$\rightarrow$$

0	488
0	488

Gambar 14. Proses perhitungan *maxpooling*

Gambar 14 merupakan tahapan *maxpooling*, tahapan ini dilakukan dengan kernel 2x2, nilai kernel akan diambil dari nilai tertinggi sehingga didapatkan nilai pertama 0. Selanjutnya dilakukan pergeseran kernel sebanyak 2 sehingga didapatkan nilai tertinggi 488.

Selanjutnya dari *output 2x2x32* akan didapatkan bentuk vector baru seperti pada gambar 15.

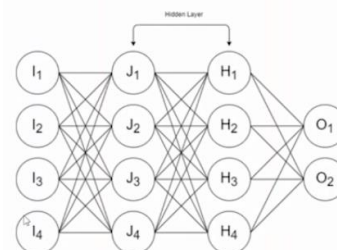
0	488
0	488

$$\rightarrow$$

0
488
0
488

Gambar 15. *Flatten* vektor baru

Gambar 15 merupakan proses *flatten* matrix 2x2 yang diubah menjadi vector matrix 4x1. *Output* tahap ini menjadi vector 1 dimensi yang akan digunakan sebagai *input fully connected layer (dense)*. Berikut ilustrasi *fully connected layer* seperti pada gambar 16.

Gambar 16. Ilustrasi proses *dense + softmax*

Keterangan:

- I = mempresentasikan nilai dari *flatten layer* sebelumnya,
- J = mempresentasikan *hidden layer* pertama dari hasil nilai *flatten layer* dikalikan dengan bobot (v),

- H = sebagai *hidden layer* kedua dimana hasil dari *hidden layer* pertama dikalikan dengan bobot (v) yang berbeda,
- O = mempresentasikan *output* dimana *output* pada penelitian ini memiliki 2 kelas (normal dan *pneumonia*).
- V = Bobot (0.2, 0.4, 0.5, 0.6).

Nilai $I1=0$, $I2=488$, $I3=0$ dan $I4=488$. Selanjutnya dilakukannya proses *fully connected layer* pada setiap *hidden layer*:

$$J_1 = (0*0.2) + (488*0.2) + (0*0.2) + (488*0.2) = 195,2$$

$$J_2 = (0*0.4) + (488*0.4) + (0*0.4) + (488*0.4) = 390,4$$

$$J_3 = (0*0.5) + (488*0.5) + (0*0.5) + (488*0.5) = 488$$

$$J_4 = (0*0.6) + (488*0.6) + (0*0.6) + (488*0.6) = 585,6$$

Hidden layer pertama dengan $j1$, $j2$, $j3$, $j4$, setiap *neuron* dari $I1$, $I2$, $I3$ dan $I4$ dikalikan dengan nilai *weight* yang berbeda-beda sehingga menghasilkan nilai $J1=195,2$, $J2=390,4$, $J3=488$ dan $J4=585,6$. Setiap *neuron* dari $J1$, $J2$, $J3$ dan $J4$ akan dikalikan lagi dengan *weight* yang berbeda-beda untuk menghasilkan nilai $H1$, $H2$, $H3$ dan $H4$ sebagai *hidden layer*:

$$H_1 = (195,2*0.4) + (390,4*0.4) + (488*0.4) + (585,6*0.4) = 663,68$$

$$H_2 = (195,2*0.5) + (390,4*0.5) + (488*0.5) + (585,6*0.5) = 829,6$$

$$H_3 = (195,2*0.2) + (390,4*0.2) + (488*0.2) + (585,6*0.2) = 331,84$$

$$H_4 = (195,2*0.6) + (390,4*0.6) + (488*0.6) + (585,6*0.6) = 995,52$$

Setiap *neuron* dari $H1$, $H2$, $H3$ dan $H4$ akan dikalikan lagi dengan *weight* yang berbeda-beda untuk menghasilkan nilai $O1$ dan $O2$ sebagai *output*:

$$O_1 = (663,68*0.5) + (829,6*0.5) + (331,84*0.5) + (995,52*0.5) = 1037,32$$

$$O_2 = (663,68*0.6) + (829,6*0.6) + (331,84*0.6) + (995,52*0.6) = 1659,2$$

Setelah selesai, selanjutnya adalah perhitungan *softmax* dengan rumus $O1$ dibagi dengan total eksponensial $O1$ dan eksponensial $O2$. Tahap ini dilakukan 2 kali sehingga diperoleh:

$$s(o_i) = \frac{e^{o_i}}{\sum_{j=1}^n e^{o_j}} \quad s(O_1) = \frac{e^{1037,32}}{e^{1037,32} + e^{1659,2}}$$

$$s(O_1) = 0.4$$

$$s(o_i) = \frac{e^{o_i}}{\sum_{j=1}^n e^{o_j}} \quad s(O_2) = \frac{e^{1659,2}}{e^{1037,32} + e^{1659,2}}$$

$$s(O_2) = 0.6$$

$$s(O_1) + s(O_2) = 1$$

$$0.4 + 0.6 = 1$$

Sehingga didapatkan bobot nilai probabilitas yang lebih besar pada $O2$ yaitu 0.6 yang berarti *input* citra yang dimasukan memprediksi normal.

HASIL DAN PEMBAHASAN

Dalam penelitian ini dilakukan klasifikasi citra x-ray anak menggunakan model Convolution Neural Network untuk mengidentifikasi apakah anak tersebut menderita *pneumonia* atau normal. Dalam penggunaan metode ini untuk mendapatkan hasil dengan tingkat akurasi terbaik, maka dilakukan beberapa percobaan dengan banyaknya neuron, dropout dan epoch yang bervariasi seperti pada Tabel 1. Sebelumnya pada proses augmentation image dilakukan percobaan dengan rotation 45° dan 60° dengan zoom 20% dan 30% dengan hasil tidak mendapatkan perubahan pada proses convolutional neural network.

Tabel 1 menunjukkan hasil pengujian untuk menentukan model convolution neural network dengan jumlah epoch, neuron dan dropout yang berbeda-beda. Banyaknya neuron dan nilai dropout yang memberikan hasil akurasi terbaik akan digunakan untuk pemodelan CNN

Tabel 1 Pengujian model

Epoch	Neuron	Dropout					
		0,2	0,4	0,5	0,6	0,7	0,8
10	16	0.8134	0.7861	0.8011	0.7548	0.8025	0.7360
	128	0.9183	0.8747	0.8338	0.8610	0.8201	0.7845
	512	0.8910	0.8733	0.9128	0.8542	0.8524	0.8110
	1024	0.8937	0.8896	0.8883	0.8515	0.9005	0.8392
20	16	0.9223	0.9033	0.8951	0.8365	0.9125	0.7231
	128	0.9319	0.9251	0.8965	0.9251	0.9035	0.7875
	512	0.9232	0.9251	0.9223	0.9122	0.8751	0.8311
	1024	0.9223	0.9155	0.9087	0.9124	0.9169	0.9264
30	16	0.9033	0.9278	0.8978	0.8597	0.8678	0.8922
	128	0.9387	0.9292	0.9278	0.9319	0.9281	0.9301
	512	0.9307	0.9346	0.9360	0.9319	0.9356	0.9218
	1024	0.9510	0.9441	0.9251	0.9278	0.9319	0.9126
40	16	0.9125	0.9178	0.8987	0.8611	0.8871	0.8969
	128	0.9212	0.9285	0.9256	0.9189	0.9052	0.9162
	512	0.9311	0.9378	0.9469	0.9423	0.9301	0.9387
	1024	0.9428	0.9466	0.9390	0.937	0.9332	0.9281
50	16	0.9145	0.9167	0.9021	0.8915	0.8964	0.915
	128	0.9223	0.9267	0.9287	0.9179	0.9212	0.9316
	512	0.9321	0.9356	0.9378	0.9389	0.9311	0.9469
	1024	0.9482	0.9501	0.9487	0.9469	0.9442	0.9318
60	16	0.9254	0.9278	0.9301	0.9256	0.9325	0.9257
	128	0.9263	0.9285	0.9314	0.9105	0.9275	0.9115
	512	0.9311	0.9425	0.9632	0.9365	0.9281	0.9525
	1024	0.9412	0.9578	0.9552	0.9632	0.9611	0.9402
70	16	0.9165	0.8856	0.8992	0.9046	0.9155	0.9187
	128	0.9365	0.9133	0.9401	0.9253	0.9164	0.9189
	512	0.9375	0.9269	0.9659	0.9578	0.9187	0.9264
	1024	0.9429	0.9433	0.9520	0.9564	0.9542	0.9302

Dari hasil uji coba pada tabel 1, nilai akurasi terbaik diperoleh pada *epoch* ke-70 yaitu sebesar 0,9659 dengan menggunakan 512 *neuron* dan *dropout* terbaik pada 0.5. Berdasarkan hal tersebut maka selanjutnya proses pemodelan menggunakan CNN untuk mengidentifikasi *pneumonia* pada anak berdasarkan citra x-ray akan menggunakan 512 *neuron* dengan *dropout* 0.5.

Berdasarkan hasil tersebut dilakukan evaluasi model CNN kembali dengan jumlah *epoch* yang berbeda-beda untuk mengetahui tingkat akurasi terbaik pada klasifikasi paru-paru anak dengan menggunakan 512 *neuron* dan *dropout* 0.5.

Tabel 2. Hasil evaluasi model

Epoch	Hasil Percobaan			
	Loss	Akurasi	Val loss	Val akurasi
Epoch 10	0.2368	0.9128	0.5000	0.7837
Epoch 19 dari 20	0.2244	0.9223	0.4531	0.8365
Epoch 30	0.1745	0.9360	0.3670	0.8510
Epoch 40	0.1472	0.9469	0.5366	0.8045
Epoch 50	0.1389	0.9496	0.3416	0.8862
Epoch 57 dari 60	0.1139	0.9632	0.2708	0.8942
Epoch 67 dari 70	0.0984	0.9659	0.2485	0.9119

Hasil evaluasi pada tabel 2, untuk jumlah neuron sebanyak 512 dan *dropout* 0,5 diperoleh

tingkat akurasi terbaik sebesar 0,9659 dengan loss 0,0984 pada *epoch* ke-67 dan 70. Dengan demikian dapat disimpulkan bahwa metode CNN dapat digunakan untuk mengidentifikasi penyakit *pneumonia* pada anak berdasarkan citra rontgen X-Ray paru-paru.

SIMPULAN DAN SARAN

1. Hasil *rontgen* citra x-ray paru-paru dapat digunakan sebagai dasar untuk mengidentifikasi penyakit *pneumonia* pada anak.
2. Tingkat akurasi terbaik untuk mengidentifikasi penyakit *pneumonia* pada anak dengan pemodelan CNN berdasarkan citra x-ray paru-paru diperoleh pada *epoch* ke-67 dan 20 dengan menggunakan 512 neuron dan *dropout* 0,5.
3. Metode CNN dapat digunakan untuk mengidentifikasi penyakit *pneumonia* pada anak dengan tingkat akurasi 0.9659 dan loss 0,0984.

Saran

Perlu dilakukan penelitian lebih lanjut untuk mengatasi kesulitan dalam identifikasi *pneumonia* melalui hasil rontgen dada, termasuk pengembangan metode identifikasi yang lebih akurat dan efisien.

UCAPAN TERIMA KASIH

Terimakasih ditujukan kepada Prodi Teknik Informatika, Fakultas Teknik Universitas Muhammadiyah Jakarta yang telah mendukung penelitian ini.

DAFTAR PUSTAKA

- Anwar, A., & Dharmayanti, I. (2014). *Pneumonia pada Anak Balita di Indonesia. Jurnal Kesehatan Masyarakat Nasional*, 359-361.
- Bahri, S., & Rachmat. (2018). Transformasi Citra Biner Menggunakan Metode Thresholding Dan Otsu Thresholding. *Jurnal Sistem Informasi Dan Teknologi Informasi*, 199-202.
- Hasanah, U., & Santik, Y. D. (2021). Faktor Intrinsik dan Ekstrinsik yang Berhubungan dengan Kejadian *Pneumonia* di Wilayah. *Jurnal*

- Kesehatan Masyarakat Indonesia, Vol.16*, 84-90.
- Hibatullah, A., & Maliki, I. (2018). Penerapan Metode Convolutional Neural Network Pada Pengenalan Pola Citra Sandi Rumput. *Jurnal Teknik Informatika*, 2-3.
- Kementerian Kesehatan RI. (2015). Rencana Aksi Program Pengendalian Penyakit dan Penyehatan Tahun 2015-2019. 14.
- Kementerian Kesehatan RI. (2020, November 12). *Pneumonia Pada Anak bisa Dicegah dan Diobati*. Retrieved Januari 30, 2024, from kemkes: <https://p2p.kemkes.go.id/pneumonia-pada-anak-bisa-dicegah-dan-diobati/>
- Laraswati, B. D. (2023, Februari 10). *Augmentasi Data, Teknik Modifikasi Untuk Perbanyak Data Anda*. Retrieved Oktober 16, 2023, from algoritma.blog: <https://blog.algoritma.blog/augmentasi-data/>
- Lina, Q. (2019, Januari 2). *Apa itu Convolutional Neural Network*. Retrieved oktober 8, 2023, from <https://medium.com/@16611110/apa-itu-convolutional-neural-network-836f70b193a4>
- Meizikri, R., Yani, F. F., & Yusrawati. (2016). Hubungan Kejadian Pneumonia Neonatus dengan Beberapa. *Jurnal Kesehatan Andalas*, 2.
- Nurvinda, G. (2021, April 11). *Pentingnya Preprocessing dalam Pengolahan Data Statistik*. Retrieved Oktober 8, 2023, from DQLab Belajar Data Science: <https://dqlab.id/pentingnya-preprocessing-dalam-pengolahan-data-statistik>
- Oliver, A. (2023, Agustus 19). *Bikin Data Lebih Mudah Dibaca, yuk, Kenalan dengan Data Preprocessing*. Retrieved Oktober 15, 2023, from <https://glints.com/id/lowongan/data-preprocessing-adalah/>
- Putri, T. A., Diana, & Purnamasari, S. D. (2020). Identifikasi Penyakit Paru-Paru Menggunakan Metode K Nearest Neighbor (KNN). *Bina Darma Conference on Computer Science*, 43-44.
- Reuter, S., Moser, C., & Baack, M. (2014, Oktober). *Gangguan Pernapasan pada Bayi Baru Lahir*. Retrieved Oktober 17, 2023, from National Library of Medicine: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4533247/>
- Rohpandi, D., Sugiharto, A., & Winara, G. A. (2015). Aplikasi Pengolahan Citra Dalam Pengenalan Pola Huruf Ngalagena Menggunakan MATLAB. *Konferensi Nasional Sistem & Informatika 2015*, 2.
- Santoso, A., & Ariyanto, G. (2018). Implementasi Deep Learningberbasis Kerasuntuk Pengenalan Wajah. *Jurnal Teknik Elektro*, 15-18.
- Suartika E. P, I., Wijaya, A. Y., & Soelaiman, R. (2016). Klasifikasi Citra Menggunakan Convolutional. *Jurnal Teknik ITS Vol.5, No.1*, A65.
- Unair. (2023, Maret 31). *Deteksi Pneumonia pada Gambar Rontgen Dada Anak Menggunakan Jaringan Syaraf Konvolusional*. Retrieved Oktober 10, 2023, from Unair News: <https://unair.ac.id/deteksi-pneumonia-pada-gambar-rontgen-dada-anak-menggunakan-jaringan-syaraf-konvolusional/>
- UNICEF. (2019, November 12). *Lembaga kesehatan dan anak memperingatkan satu anak meninggal akibat pneumonia setiap 39 detik*. Retrieved Januari 30, 2024, from UNICEF Indonesia: <https://www.unicef.org/indonesia/id/si-aran-pers/lembaga-kesehatan-dan-anak-memperingatkan-satu-anak-meninggal-akibat-pneumonia-setiap-39>
- UNICEF. (2020, Januari 31). *Kenali 6 Fakta tentang Pneumonia pada Anak*. Retrieved Oktober 17, 2023, from unicef.org: <https://www.unicef.org/indonesia/id/kesehatan/cerita/kenali-6-fakta-tentang-pneumonia-pada-anak>
- Yopento, J., Ernawati, & Coastera, F. F. (2022). Identifikasi pneumonia Pada Citra X-Ray Paru-Paru Menggunakan Metode Convolutional Neural

Network (Cnn) Berdasarkan Ekstraksi
Fitur Sobel. *Jurnal Rekursif*, 44-46.