

# Perbandingan Metode Deep Residual Network 50 dan Deep Residual Network 152 untuk Deteksi Penyakit *Pneumonia* pada Manusia

RIFQI RIZQULLAH EKA PRASETYO, MUHAMMAD ICHWAN

Program Studi Informatika, Institut Teknologi Nasional Bandung  
Email: [ikiwriz17@gmail.com](mailto:ikiwriz17@gmail.com)

Received 7 Agustus 2021 | Revised 1 November 2021 | Accepted 25 November 2021

## ABSTRAK

*Pneumonia merupakan salah satu masalah Kesehatan yang sering dijumpai dan mempunyai dampak yang signifikan di seluruh dunia. Insiden pneumonia dilaporkan meningkat sesuai dengan bertambahnya usia. Pneumonia merupakan diagnosis terbanyak ketiga. Dalam penelitian ini penulis mengidentifikasi citra paru-paru dalam bentuk citra x-ray dengan metode ResNet-50 dan ResNet-152 sebagai ekstraksi ciri dan klasifikasinya. Performa sistem diukur berdasarkan nilai akurasi, presisi, recall, dan f-measure. Eksperimen dilakukan pada dataset paru-paru dengan menggunakan dua metode tersebut dan didapatkan akurasi terbaik pada ResNet-152. Hasil menunjukkan nilai rata-rata terbaik accuracy 89,3%, precision 88,8%, recall 89,6%, dan f-measure 89%. Hasil tersebut dipengaruhi oleh jumlah dataset dari citra training, citra validation, dan citra uji.*

**Kata kunci:** *Pneumonia, Deep Residual Network, RESNET-50, RESNET-152*

## ABSTRACT

*Pneumonia is one of the most common health problems and has a significant impact throughout the world. The incidence of pneumonia is reported to increase with age. Pneumonia is the third most common diagnosis. In this study, the authors identified lung images in the form of x-ray images using the ResNet-50 and ResNet-152 methods as feature extraction and classification. System performance is measured based on the values of accuracy, precision, recall, and f-measure. Experiments were carried out on lung datasets using these two methods and the best accuracy was obtained on ResNet-152. The results show the best average value for accuracy is 89.3%, precision is 88.8%, recall is 89.6%, and f-measure is 89%. These results are influenced by the number of datasets from training images, validation images, and test images.*

**Keywords:** *Pneumonia, Deep Residual Network, RESNET-50, RESNET-152*

## 1. PENDAHULUAN

*Pneumonia* adalah infeksi pada ujung *bronkiolus* dan *alveolus* dan infeksi yang paling sering menyebabkan kematian secara terus-menerus, penyakit terbatas dibandingkan dengan populasi yang menyebabkan penundaan berturut-turut dalam pemberian untuk pasien *pneumonia*. *Pneumonia* sangat mungkin merupakan masalah medis yang paling dikenal luas dan memiliki efek kritis di seluruh dunia, terutama pada populasi yang lebih tua. *Convolutional Neural Organization* (CNN) adalah salah satu pengaturan dalam melakukan karakterisasi terprogram pada gambar sinar-X (**Maysanjaya, 2020**). CNN adalah perhitungan pembelajaran mendalam yang dapat mengambil semua sendiri dari kasus-kasus masa lalu. Apalagi dari pemeriksaan yang telah dilakukan, hasilnya menunjukkan bahwa CNN dapat mendeteksi *pneumonia* dengan kecepatan pencapaian normal di atas 90%. Ketepatan yang dikembangkan lebih lanjut diperoleh dengan mendorong perhitungan CNN baik melalui penentuan harga barang atau pekerjaan aktuasi yang berpotensi (**Jimmi Lin, 2021**).

Pengelompokan gambar memiliki banyak pekerjaan. Ini sangat baik dapat digunakan untuk mengenali berbagai daerah tergantung pada jenis penggunaan lahan. Informasi penggunaan lahan digunakan secara luas untuk persiapan metropolitan. Simbolisme tujuan tinggi juga digunakan selama peristiwa bencana seperti banjir, gunung berapi, dan musim kemarau yang serius untuk melihat efek dan bahaya (**Vance et al., 2016**).

Peningkatan desain CNN dipengaruhi oleh kontes ILSVRC (*ImageNet Huge Scope Visual Acknowledgment Rivalry*) yang diadakan setiap tahun oleh *ImageNet* yang merupakan norma dalam pengakuan artikel *benchmark*. Di pihak oposisi, penilaian tingkat kesalahan dari setiap perhitungan CNN diselesaikan untuk mengenali protes dan mengkarakterisasi gambar untuk cakupan yang sangat besar. Pada tahun 2012 rekayasa CNN bernama *AlexNet* memiliki kecepatan kesalahan 16,4% dan menjadikannya juara tahun ini (**Feriawan & Swanjaya, 2020**). Kemudian, pada saat itu, pada tahun 2013 dengan kode *Clarifai* bisa menang dengan kecepatan kesalahan 11,7%. Kemudian, pada saat itu, teknik *VGG* dengan jumlah 19 lapisan menemukan cara untuk mengurangi nilai kesalahan menjadi 7,3%. Dengan desain *GoogLeNet* jumlah 22 lapisan dengan kecepatan kesalahan hanya 6,7% berhasil menjadi juara pada tahun 2014. Juara oposisi pada tahun 2015 dengan *ResNet engineering* dari *Microsoft* memperluas penggunaan lapisan hingga 152. lapisan yang menemukan cara untuk mengurangi nilai kesalahan hingga 3,57%. Kecepatan kesalahan desain *DenseNet* sebesar 5,17%. Meskipun mendapatkan nilai kesalahan sebesar 3,57% namun lapisan yang digunakan dalam rekayasa *ResNet* adalah 152 lapisan, tugas komputasi yang diperlukan lebih tinggi dari sebelumnya. Sehingga desain *GoogLeNet* dapat menjadi rekayasa keputusan untuk melakukan pengenalan objek dengan presisi tinggi namun tidak memerlukan tugas komputasi yang tinggi seperti rekayasa *ResNet* (**Russakovsky et al., 2015**).

Perhitungan CNN sering digunakan oleh para ahli untuk mengidentifikasi suatu item, karena perhitungan ini telah diakui sebagai model perhitungan terbaik dalam menangani masalah pengenalan artikel (**Eka Putra, 2016**). Model CNN yang berbeda diusulkan oleh beberapa ahli termasuk *LeNet*, *AlexNet*, *ZFNet*, *GoogLeNet*, *VGGNet*, dan *ResNet* (**Hakim, 2018**). *ResNet engineering* dapat mengalahkan struktur yang berbeda dalam mengatur gambar pada dataset *ImageNet* dengan tingkat ketepatan yang signifikan mencapai 80,62%. Juga, memiliki beberapa struktur berbeda seperti *AlexNet*, *GoogLeNet*, dan *VGGNet* (**Munir et al., 2019**).

Pada penelitian yang di lakukan oleh (**Gao et al., 2020**) mengarahkan pemeriksaan untuk mengenali epilepsi menggunakan strategi *Convolutional Neural Organization* yang

menciptakan kecepatan ketepatan normal 92% dan dalam eksplorasi dipimpin (**Yousif et al., 2018**) mengarahkan eksplorasi untuk membedakan (tidak berbahaya) dan berbahaya (mengancam) penyimpangan dari gambar otak sinar-X menggunakan organisasi saraf *backpropagation*. Teknik *backpropagation* mempersiapkan organisasi untuk mendapatkan keselarasan antara mempersiapkan dan menguji organisasi. Ketepatan penyiapan *backpropagation* yang paling tinggi adalah 88,89% dari 9 informasi penyiapan.

Melihat gambaran masalah yang digambarkan di atas, eksplorasi yang diarahkan oleh pencipta adalah untuk membangun kerangka kerja pengaturan *pneumonia* manusia dengan menggunakan strategi CNN. Karena ada berbagai jenis desain yang dapat digunakan dalam strategi CNN, dalam tinjauan ini model komposisi yang digunakan adalah *ResNet50* dan *Resnet152* untuk melihat dan menemukan nilai terbaik pada dengan alasan bahwa di masa lalu ujian hanya menggunakan salah satunya dan belum bisa melihat yang terbaik. Pemeriksaan ini diandalkan untuk memberikan hasil terbaik sehingga cenderung digunakan untuk membantu para ahli klinis dalam membedah *pneumonia* pada manusia. Konsekuensi dari temuan *pneumonia* dapat membantu dalam menentukan jenis pengobatan yang tepat untuk individu dengan pneumonia, untuk membatasi angka kematian di Indonesia. Selain itu, eksplorasi ini juga dapat membantu upaya otoritas publik untuk mengikuti dan bekerja pada kesejahteraan umum.

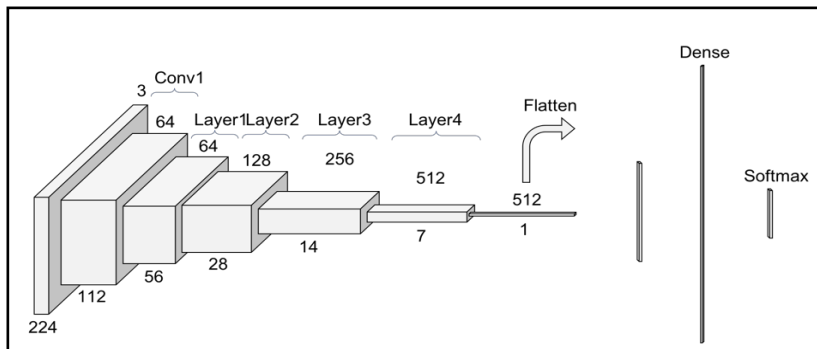
## 2. METODE PENELITIAN

### 2.1. Convolutional Neural Network

*Convolutional Neural Organization* (CNN) ditujukan untuk menangani informasi yang berupa cluster sebanyak-banyaknya, misalnya gambar bayangan yang terdiri dari pameran 2D yang berisi piksel dalam tiga nada, khususnya Merah, Hijau, dan Biru. Ada berbagai jenis CNN: 1D untuk tanda dan pengaturan biasanya digunakan untuk bahasa, 2D untuk gambar atau suara; dan 3D untuk rekaman atau gambar volumetrik (**LeCun et al., 2015**). *Convolutional Neural Organization* (CNN) adalah kemajuan dari strategi *Multi-facet Perceptron* (MLP) yang dikenang untuk *Residual Network* karena kedalaman lapisan yang berbeda dan umumnya diterapkan pada informasi gambar. Pembelajaran mendalam dipisahkan menjadi tiga klasifikasi pendekatan, yaitu pembelajaran yang diatur secara spesifik, pembelajaran tanpa bantuan, dan pembelajaran pendukung. Salah satu kemungkinan pembelajaran yang mendalam adalah untuk menggantikan elemen buatan tangan dengan perhitungan mahir untuk pembelajaran progresif tunggal (tanpa bantuan *highlight*) atau Semi-dikelola termasuk pembelajaran (semi-teratur) dan berbagai ekstraksi sorotan berjenjang (termasuk ekstraksi) (**Schmidhuber et al., 2014**). Model CNN secara efektif diterapkan dengan nama *LeNet* oleh LeChun dari AT&T Ringer Labs di Holmdel, New Jersey, AS dalam penelitian tentang pengakuan angka dan huruf. CNN telah mengalahkan strategi AI lainnya seperti *SVM*, *K-NN*, dan lainnya dalam urutan objek dalam gambar (**Eka Putra, 2016**).

### 2.2. Arsitektur Residual Network (ResNet)

*Residual Network* (*ResNet*) memenangkan ILSVRC pada tahun 2015. Organisasi ini memungkinkan pembuatan Organisasi Saraf dengan 152 lapisan dan dapat mengurangi kerumitan dibandingkan dengan *VGGNet*. Tingkat kesalahan yang dicapai dari rekayasa ini adalah 3,57% dengan tujuan dapat mengalahkan presentasi tingkat manusia pada dataset yang diberikan.



Gambar 1. Contoh Jaringan ResNet

Sumber : (He et al., 2016)

Deep residual network atau biasa disebut ResNet adalah salah satu struktur CNN yang diusulkan oleh He. pada tahun 2016. Desain ini dibuat untuk mengatasi masalah dengan pemahaman mendalam tentang persiapan, karena fakta bahwa persiapan mendalam secara keseluruhan membosankan dan terbatas pada beberapa lapisan tertentu. Jawaban untuk masalah yang diajukan oleh ResNet adalah dengan menerapkan asosiasi lewati atau rute alternatif. Kelebihan model ResNet dibandingkan dengan model struktural CNN lainnya adalah tampilan model ini tidak berkurang meskipun desainnya semakin maju. Diantaranya ada beberapa layer yang menghasilkan aspek gambar yang lebih sederhana (He et al., 2016) seperti digambarkan pada Gambar 1.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112x112			7x7, 64, stride 2		
conv2_x	56x56			3x3 max pool, stride 2		
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28x28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14x14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7x7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1x1			average pool, 1000-d fc, softmax		
FLOPs		$1.8 \times 10^9$	$3.6 \times 10^9$	$3.8 \times 10^9$	$7.6 \times 10^9$	$11.3 \times 10^9$

Gambar 2. Arsitektur ResNet

Sumber : (He et al., 2016)

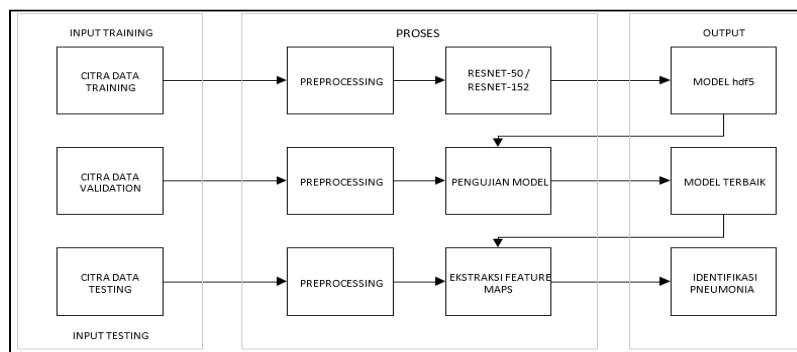
Untuk resnet50 dimana setiap komposisi lapisan diawali dengan lapisan conv1 berisikan convolution 7x7 dengan output dimensi 112x112, kemudian lapisan conv2 terdiri dari max pooling 3x3, dan batch normalization 1x1, ReLU 1x1 dan convolution 3x3 proses ini dikalikan 3x menghasilkan output dimensi 56x56. kemudian lapisan conv3 terdiri dari batch normalization 1x1, ReLU 1x1 dan Convolution 3x3 proses ini dikalikan 4x menghasilkan output dimensi 28x28. kemudian lapisan conv4 terdiri dari batch normalization 1x1, ReLU 1x1 dan convolution 3x3 proses ini dikalikan 6x menghasilkan output dimensi 14x14. kemudian lapisan conv5 terdiri dari batch normalization 1x1, ReLU 1x1 dan convolution 3x3 proses ini dikalikan 3x menghasilkan output dimensi 7x7. kemudian lapisan terakhir terdiri dari avarge max pooling dan softmax menghasilkan output dimensi 1x1 dimana hasil akhir dari arsitektur resnet50. Untuk resnet152 dimana setiap komposisi lapisan diawali dengan

lapisan conv1 berisikan *convolution* 7x7 dengan *output* dimensi 112x112, kemudian lapisan conv2 terdiri dari *max pooling* 3x3, dan *batch normalization* 1x1, *ReLU* 1x1 dan *convolution* 3x3 proses ini dikalikan 3x menghasilkan *output* dimensi 56x56. kemudian lapisan conv3 terdiri dari *batch normalization* 1x1, *ReLU* 1x1 dan *convolution* 3x3 proses ini dikalikan 8x menghasilkan *output* dimensi 28x28. kemudian lapisan conv4 terdiri dari *batch normalization* 1x1, *ReLU* 1x1 dan *convolution* 3x3 proses ini dikalikan 36x menghasilkan *output* dimensi 14x14. kemudian lapisan conv5 terdiri dari *batch normalization* 1x1, *ReLU* 1x1 dan *convolution* 3x3 proses ini dikalikan 3x menghasilkan *output* dimensi 7x7. kemudian lapisan terakhir terdiri dari *average max pooling* dan *softmax* menghasilkan *output* dimensi 1x1 dimana hasil akhir dari arsitektur *resnet*. Yang diilustrasikan pada Gambar 3.

## 2.5. Perancangan Umum

### 2.5.1. Block Diagram

Dalam penelitian ini ada beberapa tahapan untuk mengidentifikasi citra paru-paru. pada penelitian ini merupakan penelitian yang didasarkan berdasarkan beberapa penelitian yang sudah ada dan berikut ini pada gambar merupakan tahapan secara umum untuk mengidentifikasi citra paru-paru pada Gambar 3.



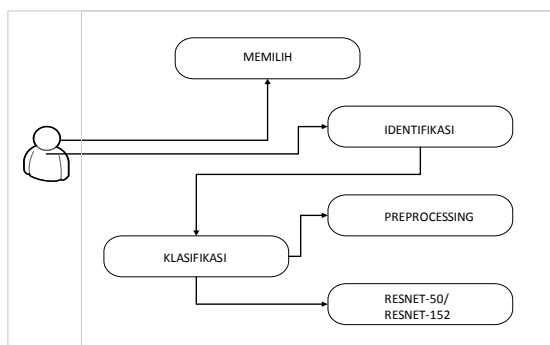
Gambar 3. Block Diagram

Persiapkan data *pneumonia* untuk melakukan *training* seperti data *training*, data *validation*, dan data *testing*. Pada *input training* dilakukan tahapan *training* yang ada pada data *training* dengan data *validation*. Pada tahap proses nya melakukan *preprocessing* menggunakan *resize* menjadi 224x224 piksel dan *normalized* data sesudah di *resize* data tersebut. Pada citra data *training* menggunakan proses dengan model *ResNet-50* dan *ResNet-152* yang terdapat proses *feature extraction* dan *classification*. Pada model *ResNet-50* dan *ResNet-152* terdapat operasi umum *batch normalization*, *ReLU activation*, dan *convolution* (Halprin Abhirawa, Jondri, 2017). Model *ResNet-50* dengan 50 layer dan *ResNet-152* dengan 152 layer yang memiliki proses *Residual block 1*, *transition layer 1*, *Residual block 2*, *transition layer 2*, *Residual block 3*, *transition layer 3*, *Residual block 4* dan *classification layer* yang menghasilkan *output* model dengan format *hdf5*. Pada data *validation* proses dilakukan pengujian model dengan data *training* yang akan menghasilkan *output* dengan model yang terbaik dalam sisi bobot.

Pada citra data *testing* masuk kedalam proses *preprocessing* dengan *resize* dengan menjadi 224x224 piksel dan *normalized* data kemudian melakukan proses ekstraksi *feature maps* dengan mengambil nilai bobot terbaik dari data *training* sehingga akan menghasilkan *output* identifikasi dari *pneumonia* yang memiliki label 0 untuk normal dan 1 untuk *pneumonia*.

## 2.5.2. Use Case Diagram

Pada Gambar 4 terdapat *use case* diagram dari sistem yang akan dibangun, dan berinteraksi antara aktor atau *user* dengan sistem. Dimana aktor atau *user* akan memilih gambar pada sistem sebagai gambar yang akan diuji kemudian aktor atau *user* dapat melakukan proses identifikasi *pneumonia* pada manusia yang terdiri dari proses klasifikasi gambar yang didahului oleh preprocessing kemudian masuk kedalam proses arsitektur *ResNet-50* dan *ResNet-152* pada citra *input*.

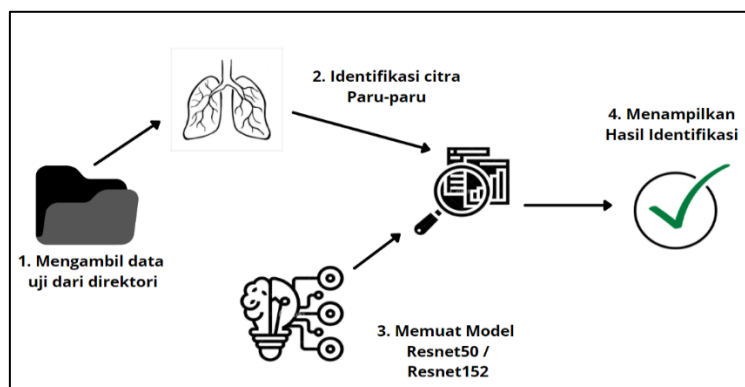


Gambar 4. Use Case Diagram

## 2.6. Pembangunan *Prototype*

### 2.6.1. Pemodelan Sistem

Pada tahap pemodelan sistem akan menjelaskan tentang *user* memilih gambar untuk data uji pada direktori dan sistem akan menampilkan hasil identifikasi paru-paru.



Gambar 5. Model Sistem

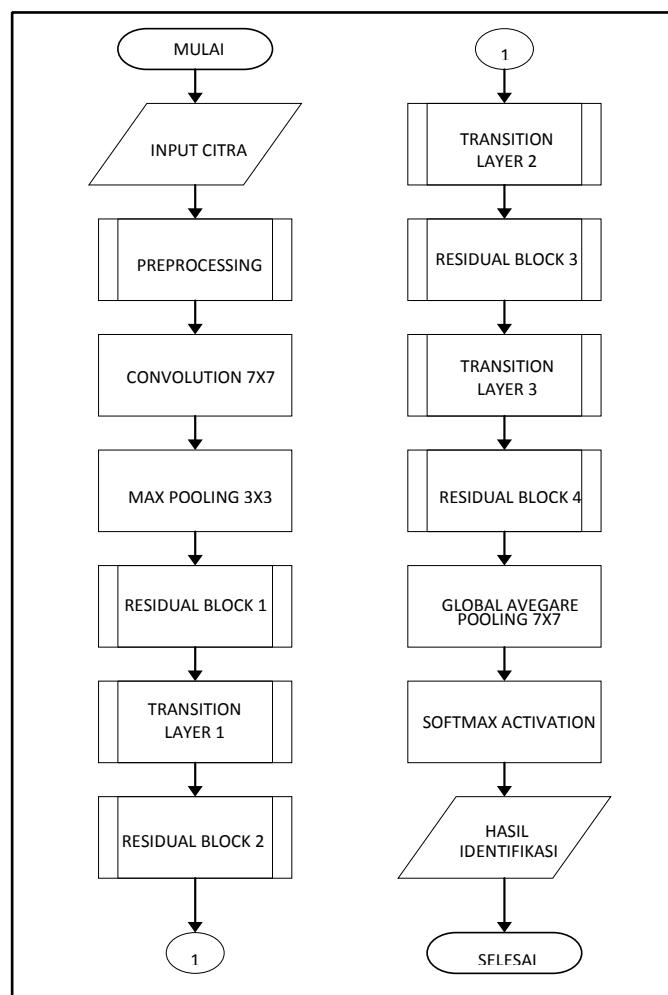
Pada Gambar 5 diilustrasikan terdapat 4 langkah proses antara *user* dengan sistem untuk mengidentifikasi citra paru-paru:

1. Langkah pertama *user* memilih file gambar yang akan diuji dari direktori.
2. Langkah kedua sistem akan mendeteksi citra paru-paru.
3. Langkah ketiga tersambung dengan langkah kedua adalah *load* model yang sudah di *training* menggunakan model *ResNet-50* dan *ResNet-152* yang berisi bobot dari model tersebut.

4. Langkah terakhir sistem akan menampilkan hasil dari identifikasi dari gambar yang di masukan oleh *user*.

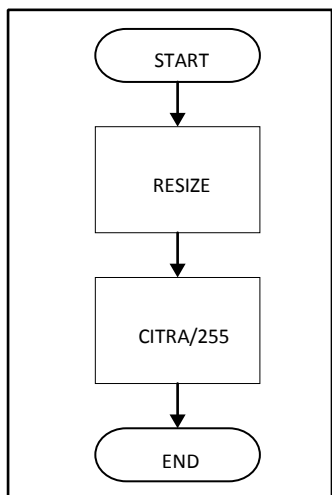
### 2.6.2. Flowchart

Pada Gambar 6 diilustrasikan *flowchart* keseluruhan sistem dalam menentukan identifikasi *pneumonia* pada manusia. Sistem dimulai dengan *input* citra atau gambar kemudian melakukan *preprocessing* kemudian setelah melakukan *preprocessing* kemudian melakukan operasi *convolution 7x7* dengan *strides 2*, operasi dilanjutkan dengan *max pooling 3x3* dengan *strides 2* dan akan mendapatkan suatu nilai matriks yang diproses ke operasi *Residual block 1*, *transition layer 1*, *Residual block 2*, *transition layer 2*, *Residual block 3*, *transition layer 3*, *Residual block 4* dan kemudian nilai matriks yang sudah diproses oleh operasi sebelumnya di proses kembali pada tahapan *classification layer* dengan operasi *global average pooling 7x7* dan kemudian *softmax activation* adalah operasi terakhir dari model *ResNet-50* dan *ResNet-152* ini untuk menentukan nilai yang mendekati antara 0 dan 1 untuk 0 identifikasi "Normal" dan untuk nilai 1 identifikasi "*Pneumonia*".



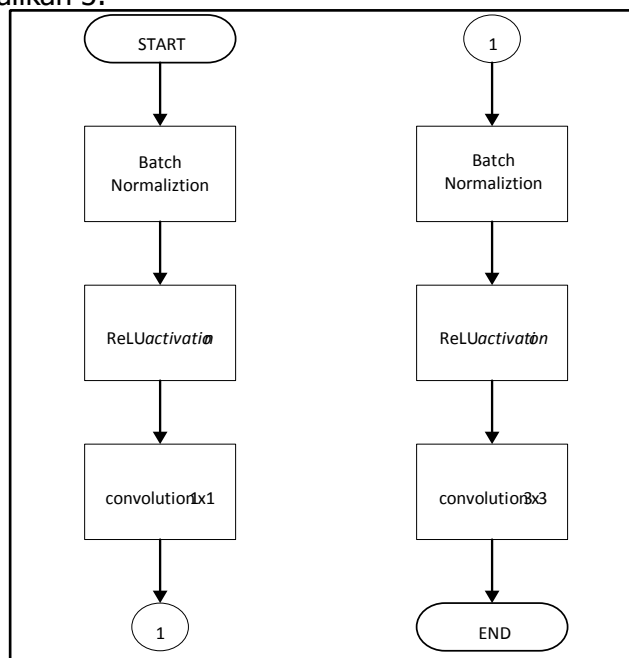
**Gambar 6. Flowchart Keseluruhan Sistem**

Untuk tahapan sub proses *preprocessing* seperti Gambar 7 yang diilustrasikan sesudah citra dimasukan ke tahap *preprocessing* yang berupa ada tahap *resize* 224x224 piksel sesudah di *resize* menjadi 224x224 piksel citra tersebut di *normalized* data citra/255 sehingga nilai matriks pada citra tersebut menjadi 0 sampai 1.



**Gambar 7. Flowchart Sub Proses Preprocessing**

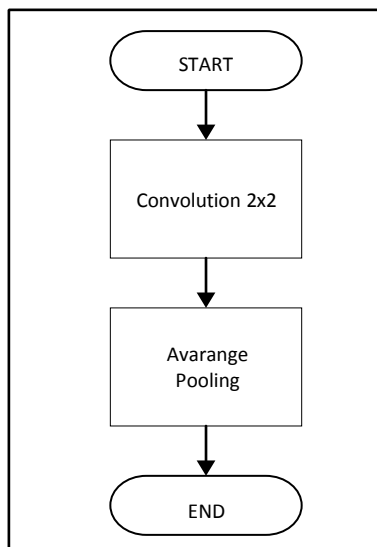
Pada Gambar 8 untuk tahap sub proses *Residual block* terdapat operasi *batch normalization*, *ReLU activation*, *convolution 1x1* operasi ini dinamakan *bottleneck* pada didalam proses *Residual block* tersebut kemudian operasi selanjutnya *batch normalization*, *ReLU activation*, *convolution 3x3* dimana nilai matriks tersebut saling keterhubungan satu sama lain sehingga nilai matriks tersebut digabungkan. Pada proses *Residual block* terdapat operasi yang dikalikan sehingga mencapai layer 50, *Residual block 1* terdapat operasi *convolution* yang dikalikan 3, *Residual block 2* terdapat operasi *convolution* yang dikalikan 4, *Residual block 3* terdapat operasi *convolution* yang dikalikan 6, dan *Residual block 4* terdapat operasi *convolution* yang dikalikan 3.



**Gambar 8. Flowchart Sub Proses Residual Block**

Pada Gambar 9 yang diilustrasikan sub proses *transition layer* terdapat operasi *convolution 2x2* dan *average pooling 1x1* dengan *strides 2*, di proses ini *transition layer* terdapat 3 proses yaitu *transition layer 1*, *transition layer 2*, dan *transition layer 3*. Proses ini dilakukan sesudah proses *Residual block* agar proses transisi antara *Residual block* dengan *transition layer*.





**Gambar 9. Flowchart Sub Proses Transition Layer**

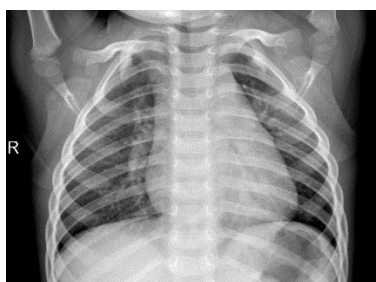
### 3. HASIL DAN PEMBAHASAN

#### 3.2. Penggunaan Dataset

Citra yang digunakan sebagai dataset merupakan citra yang didapat dari [www.kaggle.com](http://www.kaggle.com) yang di *upload* oleh Paul Mooney pada tahun 2017. Data citra yang digunakan pada proses *training*, *validation*, dan *testing* merupakan citra yang berbeda. Banyak data latih dan data uji yang di gunakan bisa dilihat pada Tabel 1.

**Tabel 1. Penggunaan Dataset**

<b>Kelas</b>	<b>Training</b>	<b>Validation</b>	<b>Testing</b>
<i>Pneumonia</i>	1.108 citra	377 citra	188 citra
Normal	1.320 citra	317 citra	158 citra



**Gambar 10. Contoh Dataset**

Sumber : (Paul Mooney, 2017)

Gambar 10 merupakan contoh citra dataset *x-ray* paru-paru manusia. Dimana citra dalam bentuk format *.jpeg* yang di dapat dari *website kaggle*.

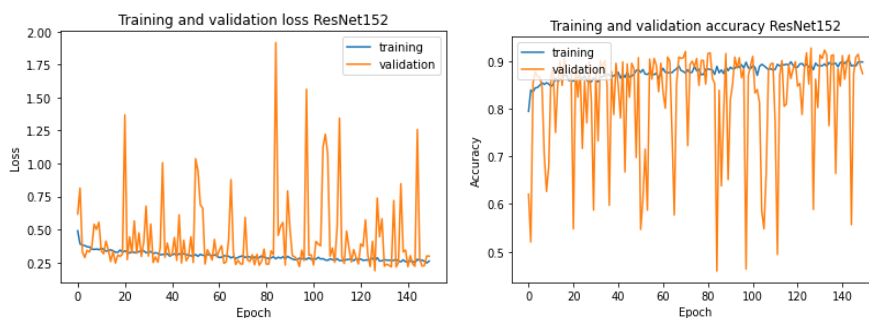
### 3.3. Pengujian Sistem

Pengujian dilakukan dengan menggunakan citra paru-paru *x-ray*, dengan menguji 3.468 citra. Dengan mengukur kinerja arsitektur *ResNet-50* dan *ResNet-152* menggunakan *confusion matrix* dimana akan mengukur kinerja sistem secara keseluruhan seperti *accuracy*, *precision*, *recall*, dan *f-measure*.

Pada Tabel 2 ditampilkan mengenai nilai *training loss*, *training accuracy*, *validation loss*, dan *validation accuracy* yang dihasilkan pada *training* yang memperlihatkan proses *epoch*. Terjadi penurunan pada *loss* tiap data *training* dengan data *validation* setiap proses penambahan *epoch*. Dan nilai akurasi data *training* dengan data *validation* terjadi kenaikan. Nilai pada *epoch* 1 ketika di *training* memiliki nilai *training loss* 0,4903 dan *validation loss* 0,6174. Peningkatan *accuracy* terjadi pada *epoch* 3 mendapatkan nilai dengan *training accuracy* 0,8354 dan *validation accuracy* 0,8547. Pada Gambar 11 merupakan grafik hasil dari *training* data yang sudah dilakukan dengan menggunakan *epoch* 150 dengan model *ResNet-152*.

**Tabel 2. Nilai Hasil Data Training**

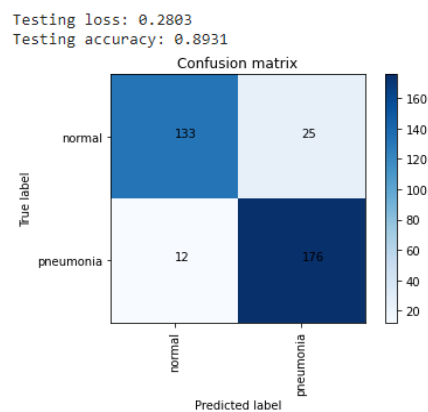
ResNet-152				
Epoch Ke-	Training Loss	Validation Loss	Training Accuracy	Validation Accuracy
1	0.6206	0.7943	0.6173	0.4902
2	0.5203	0.8391	0.8137	0.3910
3	0.6133	0.8644	0.9418	0.3392
4	0.6046	0.8627	0.9738	0.3282
5	0.8706	0.8665	0.2964	0.3162
..	..	..	..	..
146	0.7950	0.8652	0.4085	0.3131
147	0.6148	0.8665	0.5963	0.3081
148	0.8139	0.8685	0.3658	0.3077
149	0.4607	0.8789	0.9611	0.2918
150	0.5973	0.8805	0.5112	0.2965



**Gambar 11. Grafik Nilai Train Loss dan Valid Loss Training Data (Kanan). Grafik Nilai Train Accuracy dan Valid Accuracy Training Data (Kiri). Epoch 150, Learning Rate 0.0001, dan Optimizer Adam untuk ResNet-152.**

### 3.4. Pengujian Kinerja Sistem

Pengujian dilakukan dengan menggunakan citra paru-paru *x-ray*, dengan menguji 3.468 citra. Dengan mengukur kinerja arsitektur *ResNet-50* dan *ResNet-152* menggunakan *confusion matrix* dimana akan mengukur kinerja sistem secara keseluruhan seperti *accuracy*, *precision*, *recall*, dan *f-measure*. Bagian akurasi dari gambar yang tertangkap adalah tepat/signifikan. Sedangkan *recall* sangat penting untuk gambaran yang benar/signifikan yang diambil oleh *framework*. Sedangkan ketepatan atau *accuracy* adalah derajat kedekatan antara nilai yang diantisipasi dengan nilai yang sebenarnya. *F-Measure* adalah proporsi ketepatan pengujian dari estimasi akurasi dan *recall* (**Pardede & Husada, 2016**). Pada Gambar 12. terdapat *confusion matrix* dari hasil klasifikasi *pneumonia* menggunakan *ResNet-50* dan *ResNet-152*.



**Gambar 12. Confusion Matrix Hasil Klasifikasi**

Keterangan:

TN: 133, FN: 25,

FP: 12, TP: 176.

Untuk menghitung *accuracy* dengan Persamaan (1)

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

$$Accuracy = \frac{176 + 133}{176 + 133 + 25 + 12} = 0,893 * 100 = 89,3\%$$

Untuk menghitung *precision* dengan Persamaan (2)

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

$$Precision (pneumonia) = \frac{176}{176 + 12} = 0,936 * 100 = 93,6\%$$

$$Precision (normal) = \frac{133}{133 + 25} = 0,841 * 100 = 84,1\%$$

Untuk menghitung *recall* dengan Persamaan (3)

$$Recall = \frac{TP}{TP+FN} \tag{3}$$

$$Recall (pneumonia) = \frac{176}{176 + 25} = 0,875 * 100 = 87,5\%$$

$$Recall (normal) = \frac{176}{176 + 12} = 0,917 * 100 = 91,7\%$$

Untuk menghitung *F-Measure* dengan Persamaan (4)

$$f - measure = 2x \frac{(Precision \times Recall)}{(Precision + Recall)} \tag{4}$$

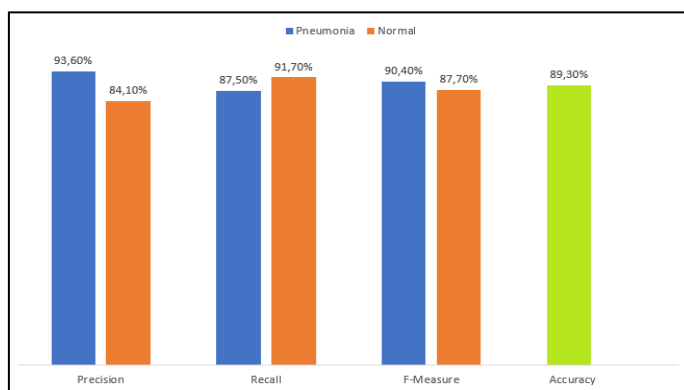
$$f - measure (pneumonia) = 2x \frac{(0,936 \times 0,875)}{(0,936 + 0,875)} = 0,904 * 100 = 90,4\%$$

$$f - measure (normal) = 2x \frac{(0,841 \times 0,917)}{(0,841 + 0,917)} = 0,877 * 100 = 87,7\%$$

**Tabel 3. Nilai Rata-Rata Accuracy, Precision, Recall, dan F-Measure Sistem dalam Mengidentifikasi Citra Pneumonia**

<b>ResNet-152</b>					
No	Kelas	Precision	Recall	F-Measure	Accuracy
1	Pneumonia	93,6%	87,5%	90,4%	89,3%
2	Normal	84,1%	91,7%	87,7%	
<b>Rata - Rata</b>		88,8%	89,6%	89%	

Hasil pengujian sistem pada *resnet152* dalam mengidentifikasi *pneumonia* dimana didapatkan nilai rata-rata dari *precision* 88.8%, *recall* 89.6%, *f-measure* 89% dan *accuracy* 89,3% bisa dilihat pada Tabel 3.



**Gambar 13. Grafik Rata-Rata Accuracy, Precision, Recall, dan F-Measure Identifikasi Citra *Pneumonia***

Hasil pengujian sistem pada *resnet152* dalam mengidentifikasi *pneumonia* dapat dilihat bentuk grafik nilai dari *precision* 84.1% dan 93.6%, *recall* 87.5% dan 91.7%, *f-measure* 90.4% dan 87,7%, dan nilai *accuracy* mencapai 89,3% bisa dilihat pada Gambar 13.

**Tabel 4. Tabel Perbandingan *ResNet-50* dan *ResNet-152* dengan *Epoch 150***

ResNet	Jumlah Epoch	Learning Rate	Optimizer	Average			Accuracy
				Precision	Recall	F1-Measure	
50	150	0,0001	Adam	0.88	0.89	0.88	0.887
152	150	0,0001	Adam	0.89	0.90	0.89	0.893

Pada pengujian ini dapat dilihat pada Tabel 4 dalam mengidentifikasi *pneumonia* dengan model *ResNet-152* mendapatkan nilai rata-rata *precision* 89%, *recall* 90% dan *f-measure* 89% dengan nilai *accuracy* 89,3%. Dan model *ResNet-50* mendapatkan nilai rata-rata *precision* 88%, *recall* 89% dan *f-measure* 88% dengan nilai *accuracy* 88,7%.

**Tabel 5. Tabel Tabel Perbandingan *ResNet-50* dan *ResNet-152* dengan *Epoch 100***

Jumlah Epoch	Jumlah Epoch	Learning Rate	Optimizer	Average			Accuracy
				Precision	Recall	F1-Measure	
50	100	0,0001	Adam	0.88	0.88	0.88	0.881
152	100	0,0001	Adam	0.85	0.87	0.86	0.861

Pada pengujian ini dapat dilihat pada Tabel 5 dalam mengidentifikasi *pneumonia* dengan model *ResNet-152* mendapatkan nilai rata-rata *precision* 85%, *recall* 87% dan *f-measure* 86% dengan nilai *accuracy* 86,1%. Dan model *ResNet-50* mendapatkan nilai rata-rata *precision* 88%, *recall* 88% dan *f-measure* 88% dengan nilai *accuracy* 88,1%.

**Tabel 6. Tabel Tabel Perbandingan *ResNet-50* dan *ResNet-152* dengan *Epoch 50***

Jumlah Epoch	Jumlah Epoch	Learning Rate	Optimizer	Average			Accuracy
				Precision	Recall	F1-Measure	
50	50	0,0001	Adam	0.88	0.89	0.89	0.887
152	50	0,0001	Adam	0.83	0.85	0.84	0.841

Pada pengujian ini dapat dilihat pada Tabel 6 dalam mengidentifikasi *pneumonia* dengan model *ResNet-152* mendapatkan nilai rata-rata *precision* 83%, *recall* 85% dan *f-measure* 84% dengan nilai *accuracy* 84,1%. Dan model *ResNet-50* mendapatkan nilai rata-rata *precision* 88%, *recall* 89% dan *f-measure* 89% dengan nilai *accuracy* 88,7%.

#### 4. KESIMPULAN

Penelitian ini telah membandingkan metode CNN dari arsitektur *ResNet-50* dengan *ResNet-152* untuk mengidentifikasi *pneumonia*. Dapat dilihat dari kedua metode tersebut bahwa *ResNet-152* dapat mengungguli *ResNet-50*. Hasil tersebut menunjukkan nilai rata-rata

terbaik *accuracy* 89,3%, *precision* 88,8%, *recall* 89,6%, dan *f-measure* 89% dengan menggunakan *epoch* 150, *learning rate* 0,0001, dan *optimizer* Adam pada *ResNet-152* setelah dilakukan pengujian dengan menguji mulai dari 50 epoch sampai dengan 150 dengan *learning rate* (0,0001). Dapat diartikan pada dilihat dari penelitian ini bahwa banyaknya jumlah lapisan layer mempengaruhi kinerja dari metode tersebut. Nilai-nilai tersebut dipengaruhi oleh dataset dari citra *training* sebesar 2.428 citra, citra *validation* sebesar 694 citra, dan citra uji sebesar 346 citra. Berdasarkan hasil pengujian bahwa ukuran citra *pneumonia* pada manusia berbeda-beda dan masih dapat melakukan klasifikasi karena sistem dapat melakukan *resize* citra.

## DAFTAR RUJUKAN

- Eka Putra, W. S. (2016). Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101. *Jurnal Teknik ITS*, 5(1). <https://doi.org/10.12962/j23373539.v5i1.15696>
- Feriawan, J., & Swanjaya, D. (2020). Perbandingan Arsitektur Visual Geometry Group dan MobileNet Pada Pengenalan Jenis Kayu. *Seminar Nasional Inovasi Teknologi*, 185–190. <https://proceeding.unpkediri.ac.id/index.php/inotek/article/view/84>
- Gao, X., Yan, X., Gao, P., Gao, X., & Zhang, S. (2020). Automatic detection of epileptic seizure based on approximate entropy, recurrence quantification analysis and convolutional neural networks. *Artificial Intelligence in Medicine*, 102, 101711. <https://doi.org/10.1016/j.artmed.2019.101711>
- Hakim, L. (2018). Seloka : Jurnal Pendidikan Bahasa dan Sastra Indonesia Learning Writing Explanatory Text Using Group Investigation Models Based on Learning Style. *Seloka: Jurnal Pendidikan Bahasa Dan Sastra Indonesia*, 7(3), 259–266.
- Halprin Abhirawa, Jondri, A. A. (2017). *Pengenalan Wajah Menggunakan Convolutional Neural Network*. 4(3), 4907–4916.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp. 770–778). <https://doi.org/10.1109/CVPR.2016.90>
- Jimmi Lin, H. I. (2021). *Klasifikasi Pneumonia Pada Citra X-Rays Paru-Paru Menggunakan GLCM Dan LVQ 1,2*. 1(2), 184–194.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. <https://doi.org/10.1038/nature14539>
- Maysanjaya, I. M. D. (2020). Klasifikasi Pneumonia pada Citra X-rays Paru-paru dengan

Convolutional Neural Network. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, 9(2), 190–195. <https://doi.org/10.22146/jnteti.v9i2.66>

Munir, K., Elahi, H., Ayub, A., Frezza, F., & Rizzi, A. (2019). Cancer Diagnosis Using Deep Learning: A Bibliographic Review. *Cancers*, 11(9). <https://doi.org/10.3390/cancers11091235>

Pardede, J., & Husada, M. (2016). Comparison of VSM, GVSM, and LSI in information retrieval for Indonesian text. *Jurnal Teknologi*, 78. <https://doi.org/10.11113/jt.v78.8637>

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3), 211–252. <https://doi.org/10.1007/s11263-015-0816-y>

Schmidhuber, J., S., Dalle, I., & Galleria, S. (2014). *Deep Learning in Neural Networks: An Overview*. 1–88.

Vance, T. C., Merati, N., Yang, C., & Yuan, M. (2016). Cloud computing for ocean and atmospheric science. *OCEANS 2016 MTS/IEEE Monterey*, 1–4. <https://doi.org/10.1109/OCEANS.2016.7761270>

Yousif, N., Cole, J., Rothwell, J. C., Diedrichsen, J., Zelik, K. E., Winstein, C. J., Kay, D. B., Wijesinghe, R., Protti, D. A., Camp, A. J., Quinlan, E., Jacobs, J. V, Henry, S. M., Horak, F. B., Jacobs, J. V, Fraser, L. E., Mansfield, A., Harris, L. R., Merino, D. M., ... Dublin, C. (2018). Implementasi Deep Learning Untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Wayang Golek. *Journal of Physical Therapy Science*, 9(1), 1–11. <http://dx.doi.org/10.1016/j.neuropsychologia.2015.07.010>  
<http://dx.doi.org/10.1016/j.visres.2014.07.001>  
<https://doi.org/10.1016/j.humov.2018.08.006>  
<http://www.ncbi.nlm.nih.gov/pubmed/24582474>  
<https://doi.org/10.1016/j.gaitpost.2018.12.007>