

ANALISIS PERFORMA PRE-TRAINED MODEL CONVOLUTIONAL NEURAL NETWORK DALAM MENDETEKSI PENYAKIT TUBERKULOSIS

Ovy Rochmawanti^{*1}, Fitri Utamingrum², Fitra A. Bachtiar³

^{1,2,3}Fakultas Ilmu Komputer Universitas Brawijaya

Email: ¹ovyrochmawanti@student.ub.ac.id, ²f3_ningrum@ub.ac.id, ³fitra.bachtiar@ub.ac.id

*Penulis Korespondensi

(Naskah masuk: 01 Desember 2020, diterima untuk diterbitkan: 21 Juli 2021)

Abstrak

Tuberkulosis (TB) merupakan salah satu penyakit berbahaya yang dapat menular lewat udara dan sering menyebabkan kematian apabila tidak cepat ditangani. Penyakit TB bisa disembuhkan dengan deteksi dini sehingga penderita dapat segera mendapatkan pengobatan yang tepat. Metode Convolutional Neural Network (CNN) digunakan untuk mendeteksi penyakit TB melalui foto rontgen dada. Penelitian ini bertujuan untuk menentukan model CNN yang mampu menghasilkan performa paling baik dalam mendeteksi penyakit TB. Pengujian dilakukan dengan menggunakan lima pre-trained model yang telah disediakan oleh Keras yaitu *ResNet50*, *DenseNet121*, *MobileNet*, *Xception*, *InceptionV3*, dan *InceptionResNetV2*. Perbedaan ukuran gambar yang digunakan pada saat pelatihan dan pengujian juga akan dianalisis pengaruhnya terhadap nilai akurasi yang dihasilkan dan waktu komputasinya. Hasil pengujian menunjukkan bahwa model *DenseNet121* mampu menghasilkan nilai akurasi tertinggi dalam mendeteksi penyakit TB, yaitu 91,57%. Sedangkan model *MobileNet* merupakan model dengan waktu komputasi tercepat untuk semua ukuran gambar yang diuji. Semakin besar ukuran citra maka semakin tinggi nilai akurasinya, namun di sisi lain waktu komputasi juga akan semakin lama.

Kata kunci: *tuberculosis, TB, pre-trained, augmentation, CNN*

PERFORMANCE ANALYSIS OF PRE-TRAINED CONVOLUTIONAL NEURAL NETWORK MODELS WHEN DETECTING TUBERCULOSIS DISEASE

Abstract

Tuberculosis (TB) is one of the dangerous disease that can be transmitted through the air and often causes death if not treated quickly. This illness can be cured with early detection, so that sufferers can immediately get the right treatment. The Convolutional Neural Network (CNN) method is used to detect TB disease through chest X-rays. This study aims to determine which CNN model is able to produce the best performance in detecting TB disease. Testing was carried out using five pre-trained models provided by Keras namely ResNet50, DenseNet121, MobileNet, Xception, InceptionV3, and InceptionResNetV2. The difference in image size used during training and testing will also be analyzed for its effect on the resulting accuracy value and its computation time. The test results showed that the DenseNet121 model was able to produce the highest accuracy value in detecting TB disease, namely 91.57%. Meanwhile, the MobileNet model is the model with the fastest computation time for all image sizes tested. The bigger the image size, the higher the accuracy value, but on the other hand the computation time will also be longer.

Keywords: *tuberculosis, TB, pre-trained, augmentation, CNN*

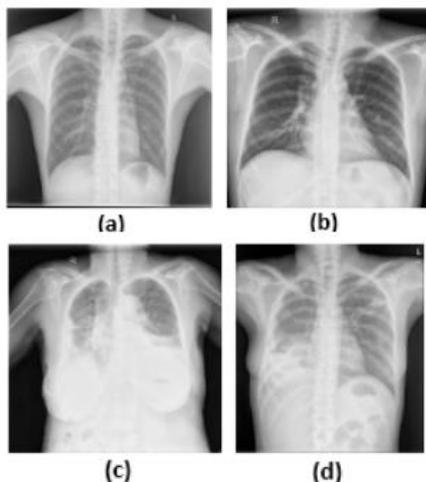
1. PENDAHULUAN

Tuberkulosis (TB) adalah penyakit menular yang disebabkan oleh *Mycobacterium Tuberculosis* dan umumnya menyerang paru-paru. Berdasarkan data *World Health Organization*, pada tahun 2018, sekitar 10 juta orang menderita penyakit TB dan 98.000 orang diantaranya meninggal dunia sehingga TB termasuk dalam salah satu penyakit paling mematikan di dunia. Kasus TB yang telah

ditemukan tersebar di berbagai negara namun kasus terbanyak berada di wilayah Asia Tenggara, Afrika dan Pasifik Barat. Indonesia menempati peringkat ketiga dengan jumlah kasus terbanyak di dunia setelah India dan China yaitu sebanyak 845 ribu kasus.

Penyakit TB bisa disembuhkan, namun apabila terlambat mendapatkan penanganan maka dapat meningkatkan resiko kematian. Deteksi dini terhadap penyakit TB menjadi sangat penting

dilakukan agar penderita TB bisa segera diobati dan meminimalisir



Gambar 1. Contoh foto rontgen dada yang terdapat pada dataset

risiko terjadinya komplikasi. Penyakit TB umumnya dapat didiagnosis melalui pemeriksaan x-ray dada. Oleh karena itu diperlukan suatu sistem yang mampu membedakan foto rontgen normal (tanpa TB) dan abnormal (dengan TB) dengan akurat dan cepat.

Teknologi *Computer Vision (CV)* dan *Machine Learning (ML)* saat ini sudah sangat berkembang dan telah dimanfaatkan di berbagai bidang termasuk salah satunya di bidang kesehatan. Kombinasi kedua metode tersebut sering digunakan dalam berbagai penelitian untuk mendeteksi dan mengklasifikasikan beberapa penyakit seperti : pneumothorax (Chan, Zeng, Wu, Wu, & Sun, 2018), kanker payudara (Kaushal, Bhat, Koundal, & Singla, 2019), tumor otak (Rajan & Sundar, 2019), batu ginjal (Nithya, Appathurai, Venkatadri, Ramji, & Anna, 2020), pneumonia (Biswas, Ghosh, Bhattacharyya, Platos, Snasel, & Chakrabarti, 2020) dan kanker kulit (Savera, Suryawan, & Setiawan, 2020).

Pada penelitian sebelumnya, TB bisa dideteksi menggunakan beberapa metode *CV* dan *ML*. Metode *graph cut segmentation* (Jaeger, Karargyris, Candemir, Folio, Siegelman, Callaghan, McDonald, 2013) dan *pyramid histograms of oriented gradients* (Santosh, Vajda, Antani, & Thoma, 2016) digunakan untuk mengekstraksi fitur bentuk maupun tekstur yang terdapat pada gambar. Sedangkan untuk klasifikasi gambar menggunakan beberapa metode *ML* seperti *support vector machine (SVM)*, *alternating decision tree* dan *neural network*. Hasil akurasi yang dihasilkan pada kedua penelitian masih belum optimal dimana nilai akurasi tertinggi yang didapatkan adalah sebesar 86,3%. Pengembangan masih perlu dilakukan untuk mendapatkan performa lebih baik dengan nilai akurasi yang tinggi.

Penelitian ini menggunakan metode *Convolutional Neural Network (CNN)* yang saat ini sedang populer untuk meningkatkan nilai akurasi dalam mendeteksi penyakit TB. CNN sering dipilih

karena bisa memuat keseluruhan informasi dari berbagai skala hingga dapat mengklasifikasikan suatu objek atau gambar dengan lebih akurat (Zhang, Sargent, Pan, Gardiner, Hare, & Atkinson, 2018). Banyak penelitian yang telah dilakukan untuk mengembangkan metode ini diantaranya dengan membuat berbagai model arsitektur yang masing-masing modelnya memiliki kedalaman dan jumlah *layer* yang berbeda-beda. Beberapa model CNN yang sering digunakan untuk klasifikasi dan deteksi gambar adalah VGG (Simonyan & Zisserman, 2014), *InceptionV3* (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2016), ResNet (He, Zhang, Ren, & Sun, 2015), *InceptionResNetV2* (Szegedy, Ioffe, Vanhoucke, & Alemi, 2017), DenseNet (Huang, Liu, Van der Maaten, & Weinberger, 2017), *Xception* (Chollet, 2017) dan *MobileNet* (Howard, Zhu, Chen, Kalenichenko, Wang, Weyand, & Adam, 2017).

Model DenseNet dengan 121 *layer* digunakan oleh (Rajpurkar, Irvin, Zhu, Yang, Mehta, Duan, & Ng, 2017) untuk mengklasifikasikan 14 penyakit berbeda dengan dataset berupa foto rontgen dada. Penelitian yang dilakukan oleh (Saxena, Maheshwari, Tayal, & Maheshwari, 2019) berhasil mengklasifikasikan tumor pada gambar *Magnetic Resonance Imaging (MRI)* dengan menggunakan model arsitektur VGG16, Inception-v3 dan Resnet-50. Sedangkan (Purnama, Hernanda, Ratna, Nurtanio, Hidayati, Purnomo, & Rachmadi, 2019) memilih menggunakan model *MobileNet* dan Inception-V3 untuk mengklasifikasikan tujuh jenis penyakit kulit. Untuk mendeteksi kanker payudara, (Chaves & Gonçalves, Albertini, Lee, Jeon, & Fernandes, 2020) menggunakan 6 model CNN dengan arsitektur yang berbeda. Penelitian-penelitian yang telah disebutkan sebelumnya memperlihatkan bahwa metode CNN dengan berbagai model arsitektur mampu melakukan deteksi dan klasifikasi penyakit dengan nilai akurasi yang cukup tinggi. Oleh karena itu, untuk mendeteksi foto rontgen dengan penyakit TB, penelitian ini juga akan menggunakan metode CNN dengan enam model arsitektur yang berbeda yaitu *ResNet50*, *DenseNet121*, *MobileNet*, *Xception*, *InceptionV3*, dan *InceptionResNetV2*.

Penelitian ini bertujuan untuk mengevaluasi dan membandingkan performa dari masing-masing model CNN untuk menemukan model terbaik yang mampu menghasilkan nilai akurasi yang tinggi. Penelitian ini juga akan menganalisis apakah adanya perbedaan ukuran gambar yang digunakan pada saat proses pelatihan berpengaruh terhadap performa dari model CNN untuk mendeteksi penyakit TB.

2. METODE PENELITIAN

2.1. Dataset

Dataset yang digunakan pada penelitian ini berasal dari Shenzhen No.3 People's Hospital,

Guangdong Medical College Shenzhen China (Jaeger, Candemir, Antani, Wáng, Lu, & Thoma, 2014). Masing-masing gambar telah diberikan tanda sesuai dengan kelasnya yaitu kelas normal maupun kelas TB. Dataset berjumlah 662 gambar yang terbagi atas 326 kasus normal dan 336 kasus TB. Dataset yang tersedia berformat PNG dan gambar rata-rata berukuran 3000x3000. Gambar dengan kelas normal dapat dilihat pada Gambar 1 bagian a dan b, sedangkan bagian c dan d merupakan gambar yang sudah diberikan label dengan TB.

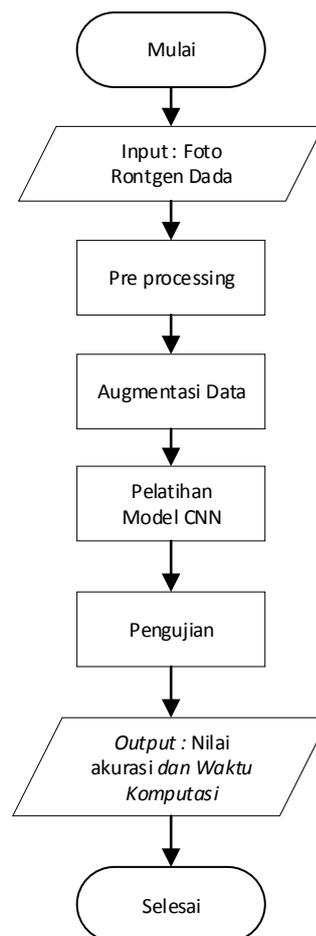
2.2. Desain Sistem

Gambar 2 merupakan alur atau proses yang menjelaskan secara detail percobaan yang dilakukan untuk mengetahui performa dari metode CNN dalam mendeteksi penyakit TB. Tahapan pertama adalah melakukan *preprocessing* pada gambar masukan. Pada tahap ini ukuran masing-masing gambar dalam dataset akan diubah menjadi lebih kecil daripada ukuran aslinya. Kemudian dataset akan dibagi menjadi tiga kelompok yaitu data latih, data validasi dan data uji. Pada tahap berikutnya, proses augmentasi data dilakukan pada data latih dengan cara menggeser, merotasi, memperbesar dan membalik gambar. Hasil augmentasi akan menjadi data masukan untuk proses pelatihan menggunakan metode CNN dengan teknik *transfer learning*. Terdapat enam model CNN yang digunakan pada tahapan ini yaitu *ResNet50*, *DenseNet121*, *InceptionResNetV2*, *Xception*, *InceptionV3*, dan *MobileNet*.

Proses pengujian yang dilakukan pada tahap akhir akan menampilkan performa masing-masing model CNN berupa nilai akurasi yang merupakan hasil prediksi dengan data uji yang telah disiapkan sebelumnya. Hasil keluaran lainnya dari sistem adalah perhitungan waktu komputasi selama proses pelatihan dilakukan.

2.3. Preprocessing

Pada tahapan ini, semua gambar yang digunakan untuk pelatihan dan pengujian akan diperkecil ukurannya ke dalam beberapa ukuran yaitu 128x128, 150x150, 224x224 dan 299x299 untuk melihat pengaruhnya terhadap akurasi dari hasil prediksi pada masing-masing model CNN. Ukuran gambar diperkecil dengan tujuan untuk mengurangi beban komputasi yang berat dan pemrosesan yang lebih cepat (Varshni, Thakral, Agarwal, Nijhawan, & Mittal, 2019). Dataset kemudian dibagi menjadi data latih, data validasi dan data uji. Data latih selanjutnya akan diaugmentasi sedangkan data validasi dan data uji akan digunakan pada saat tahapan pengujian untuk mengetahui performa dari setiap model CNN.



Gambar 2. Diagram alir untuk mendeteksi TB

2.4. Augmentasi Data

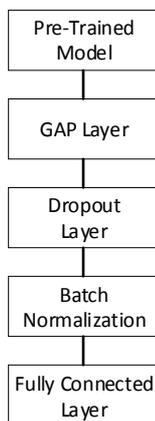
Proses augmentasi akan mengubah dan memodifikasi gambar sedemikian rupa sehingga komputer akan mendeteksi bahwa gambar yang diubah adalah gambar yang berbeda, namun manusia masih dapat mengetahui bahwa gambar yang diubah tersebut adalah gambar yang sama (Perez & Wang., 2017). Teknik augmentasi ini sering dilakukan terutama pada pengujian yang jumlah datanya terbatas (Mahmud, Adiwijaya, & Faraby, 2019). Proses augmentasi yang dilakukan pada penelitian ini adalah membalikkan gambar secara horisontal, merotasi dan melakukan pembesaran (*zoom in*) gambar secara acak serta menggeser gambar secara horisontal dan vertikal.

2.5. Convolutional Neural Network

Convolutional Neural Network (CNN) adalah salah satu metode yang mampu melakukan klasifikasi pada gambar, suara maupun teks. Arsitektur CNN umumnya memiliki tiga tipe *layer* yang menjadi inti yaitu *convolution layer*, *pooling layer*, dan *fully connected layer*. *Convolution layer* dan *pooling layer* umumnya digunakan untuk mengekstraksi informasi yang terdapat pada gambar. Sedangkan *fully connected layer* akan melakukan

prediksi dengan memberikan label yang paling tepat sebagai hasil deteksi dan klasifikasi.

Berbagai penelitian sudah dilakukan untuk mengembangkan metode CNN yaitu dengan membuat beberapa model arsitektur yang diharapkan mampu meningkatkan nilai akurasi dan mengurangi nilai error. Masing-masing model CNN memiliki susunan, bentuk dan jumlah layer yang berbeda.



Gambar 3. Arsitektur Model CNN yang digunakan.

Model CNN yang digunakan pada penelitian ini adalah *ResNet50*, *DenseNet121*, *MobileNet*, *Xception*, *InceptionV3*, dan *InceptionResNetV2*.

Residual Neural Networks (ResNet) mulai populer ketika memperoleh juara pertama pada kompetisi *ImageNet Large Scale Visual Recognition Challenge (ILSVRC)* tahun 2015. Model ini memperkenalkan teknik *residual connection* yang dapat digunakan untuk mencegah terjadinya kehilangan informasi selama proses pelatihan dilakukan sehingga dapat meningkatkan performa dari model (He, Zhang, Ren, & Sun, 2016). *ResNet50* merupakan salah satu varian dari model *ResNet* yang memiliki 50 layer dan sering digunakan untuk klasifikasi karena mampu menunjukkan performa yang tinggi dengan beban komputasi yang rendah.

Arsitektur *Inception deep convolutional* diperkenalkan pada tahun 2015 yang lebih dikenal dengan nama *GoogLeNet*. Model *GoogLeNet* inilah yang menjadi *InceptionV1*. Kemudian dilakukan perbaikan dengan adanya penambahan *batch normalization* sehingga menjadi arsitektur baru yaitu *InceptionV2*. Adanya ide mengenai penambahan faktorisasi menandai lahirnya *InceptionV3*. Model ini berhasil menempati peringkat kedua pada kompetisi ILSVRC tahun 2015. Arsitektur *InceptionResNetV2* merupakan variasi dari *InceptionV3* dengan mengkombinasikan arsitektur *ResNet* dan *Inception* untuk lebih meningkatkan performa. Berdasarkan hasil pengujian yang telah dilakukan, arsitektur ini menghasilkan *error rate* lebih rendah daripada *InceptionV3* (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2016).

Dense convolutional networks (DenseNet) diperkenalkan oleh Huang, Liu dan Maaten melalui

jurnalnya yang dipublikasikan pada tahun 2017. Model ini dibangun dengan struktur dimana masing-masing layer akan terhubung dengan layer-layer selanjutnya (Huang, Liu, Van der Maaten, & Weinberger, 2017). Arsitektur *DenseNet* menggunakan *dense connection* untuk mengurangi terjadinya *overfitting* ketika jumlah datasetnya kecil (Taro, 2019). Model *DenseNet* yang digunakan pada penelitian ini adalah *DenseNet121*.

Arsitektur *Extreme Inception (Xception)* diperkenalkan oleh F.Chollet pada tahun 2017. Arsitektur ini dibagi menjadi tiga bagian utama yaitu *Entry Flow*, *Middle Flow*, dan *Exit Flow*. Model *Xception* dibangun dengan menggunakan *Depthwise Separable Convolutions*. Meskipun *Xception* memiliki jumlah parameter lebih sedikit dari daripada *InceptionV3*, namun performanya masih lebih baik jika dibandingkan *InceptionV3* (Chollet, 2017).

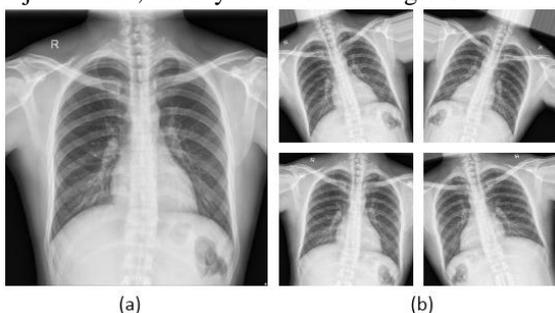
Arsitektur *MobileNet* merupakan model yang tepat untuk digunakan pada aplikasi *mobile* dan *embedded vision*. Model ini juga menggunakan *depthwise separable convolution* yang merupakan kombinasi dari *depthwise convolution* dan *pointwise convolution*. Total jumlah layer yang dimiliki oleh *MobileNet* berjumlah 28 layer (Howard, Zhu, Chen, Kalenichenko, Wang, Weyand, & Adam, 2017).

Model CNN yang diusulkan pada penelitian ini memiliki arsitektur seperti pada Gambar 3. *Pre-trained model* yang digunakan pada penelitian ini memiliki arsitektur yang identik dengan model CNN dengan nama sama yang telah dijelaskan sebelumnya yaitu *ResNet50*, *DenseNet121*, *MobileNet*, *Xception*, *InceptionV3* dan *InceptionResNetV2*, hanya saja tidak menyertakan *fully connected layer* nya. *Pre-trained model* ini telah disediakan oleh Keras. Keras merupakan *high-level API* yang sering digunakan untuk menyelesaikan permasalahan yang berhubungan dengan *deep learning*. Keras dapat dijalankan diatas TensorFlow, CNTK, dan Theano. Jumlah parameter untuk masing-masing model berbeda. Jumlah parameter paling banyak dimiliki oleh *ResNet50* dan jumlah paling sedikit dimiliki oleh *MobileNet* dimana *ResNet50* memiliki 12 kali lebih banyak parameter daripada *MobileNet*.

Pre-trained model yang digunakan pada penelitian ini sudah dilatih sebelumnya menggunakan dataset *ImageNet*. *ImageNet* merupakan dataset gambar berukuran besar yang sering digunakan oleh peneliti untuk menguji model CNN yang sedang dibangun. Bobot yang terdapat pada masing-masing *pre-trained model* akan diunduh secara otomatis ketika model diinstansiasi untuk pertama kali. Selain *pre-trained model*, ada beberapa layer juga ditambahkan seperti yang terlihat pada Gambar 4 yaitu *Global Average Pooling (GAP) layer*, *dropout layer*, *batch normalization* dan *fully connected layer*. *GAP layer*

dan *dropout layer* digunakan untuk meminimalisir terjadinya *overfitting*.

Fungsi aktivasi yang terdapat di *fully connected layer* adalah fungsi *softmax*. Fungsi ini akan menghitung probabilitas yang umum digunakan pada saat melakukan klasifikasi multi kelas. Fungsi ini biasanya digunakan di bagian akhir dari *fully connected layer* untuk menghasilkan nilai probabilitas suatu objek terhadap kelas yang ada (Goodfellow, Bengio, & Courville, 2016). Fungsi *softmax* memiliki rentang nilai probabilitas antara 0 dan 1 serta apabila semua nilai probabilitasnya dijumlahkan, hasilnya akan sama dengan satu.



Gambar 4. Gambar asli (a) dan hasil augmentasi (b)

2.6. Pelatihan

Tahapan pelatihan merupakan tahap dimana model CNN akan dilatih menggunakan data hasil proses augmentasi pada tahap sebelumnya. Jumlah *epoch* dan *learning rate* perlu diinisialisasi terlebih dahulu sebelum pelatihan dilakukan. *Learning rate* merupakan salah satu hyper parameter yang sangat mempengaruhi performa suatu model CNN. Nilai *learning rate* yang digunakan pada penelitian ini adalah 0,0001, sedangkan jumlah *epoch* yang digunakan adalah 10.

Perhitungan waktu komputasi juga dilakukan pada tahap ini. Waktu komputasi dihitung berdasarkan waktu yang diperlukan untuk melatih masing-masing model CNN.

2.7. Pengujian

Jumlah gambar yang digunakan untuk tahap pengujian adalah sebesar 25% dari keseluruhan total jumlah dataset. Gambar untuk pengujian sudah dipisahkan diawal sehingga tidak melalui proses augmentasi maupun pelatihan. Hasil pengujian berupa nilai akurasi yang mencerminkan performa dari masing-masing model CNN dalam mengklasifikasikan data uji normal (tanpa TB) maupun abnormal (dengan TB).

Nilai akurasi merupakan jumlah data uji yang berhasil diprediksi benar dibandingkan dengan jumlah keseluruhan data uji. Nilai akurasi dapat dihitung menggunakan Persamaan 1 dengan empat nilai yang dihasilkan dari *confusion matrix* yaitu *True Negative (TN)*, *True Positive (TP)*, *False Positive (FP)* dan *False Negative (FN)*.

Tabel 1. Waktu komputasi model CNN dengan ukuran gambar 128x128 dan 150x150

Model CNN	Ukuran Gambar	
	128x128	150x150
<i>ResNet50</i>	12 menit	16 menit
<i>DenseNet121</i>	13 menit	17 menit
<i>MobileNet</i>	3 menit	3 menit
<i>InceptionResNetV2</i>	15 menit	19 menit
<i>InceptionV3</i>	7 menit	9 menit
<i>Xception</i>	13 menit	17 menit

$$\text{Akurasi} = \frac{TP+TN}{TP+FP+FN+TN} \times 100\% \quad (1)$$

TN merupakan data negatif yang diklasifikasikan dengan benar sebagai data negatif, sedangkan nilai *TP* adalah data positif yang diklasifikasikan dengan benar sebagai data positif. Nilai *FP* merupakan data negatif yang diklasifikasikan sebagai data positif., sedangkan *FN* adalah data positif namun diklasifikasikan sebagai data negatif (Han & Kamber, 2011).

3. HASIL DAN ANALISIS

Perangkat keras yang digunakan pada penelitian ini memiliki spesifikasi Intel Xeon CPU E3-1220 v3 @3.10 GHz dan RAM 32 GB dengan sistem operasi Windows 7 Professional. Sedangkan perangkat lunak yang digunakan untuk mendapatkan performa masing-masing model CNN adalah Python 3.7 dan Jupyter notebook (Anaconda3).

Jumlah gambar yang terdapat dalam dataset adalah 662 gambar. Dataset kemudian dibagi tiga bagian yaitu 398 gambar untuk data latih, 98 gambar untuk data validasi dan 166 gambar untuk data uji. Pada masing-masing bagian telah diatur agar tidak ada data yang *overlap* satu sama lain.

Semua gambar pada ketiga bagian tersebut akan melalui proses *preprocessing* dimana gambar akan diubah ukurannya menjadi lebih kecil dari ukuran gambar asli. Ada empat skenario pengujian yang digunakan untuk melihat apakah ukuran gambar masukan mempengaruhi performa dari masing-masing model CNN. Keempat skenario menggunakan ukuran gambar yang berbeda yaitu 128x128, 150x150, 224x224 dan 299x299.

Tahap selanjutnya adalah proses augmentasi data yang bertujuan untuk memberikan lebih banyak variasi pada data latih yang telah melalui tahapan *preprocessing* sebelumnya. Tahapan ini membuat jumlah data latih menjadi sepuluh kali lipat dari jumlah asli dari data latih.

Proses augmentasi yang dilakukan pada penelitian ini adalah membalikkan gambar secara horisontal, melakukan pembesaran pada gambar secara acak, memutar gambar acak sebesar 20 derajat dan menggeser gambar secara baik secara vertikal maupun horisontal. Contoh gambar hasil augmentasi dapat dilihat pada Gambar 4. Dari

gambar tersebut terlihat bahwa hanya dengan menggunakan satu gambar saja dapat menghasilkan empat gambar baru yang merupakan variasi dari gambar aslinya. Proses augmentasi hanya dilakukan pada data latih saja dan tidak dilakukan untuk data validasi.

Dataset baru yang berisi hasil proses augmentasi selanjutnya akan digunakan untuk pembentukan model CNN di tahap pelatihan. Data validasi juga digunakan pada tahap ini untuk mengukur kinerja model CNN dengan menghitung tingkat kesalahan

Tabel 2. Waktu komputasi model CNN dengan ukuran gambar 224x224 dan 299x299

Model CNN	Waktu Komputasi	
	224x224	299x299
<i>ResNet50</i>	32 menit	58 menit
<i>DenseNet121</i>	40 menit	1 jam 11 menit
<i>MobileNet</i>	11 menit	21 menit
<i>InceptionResNetV2</i>	39 menit	1 jam 9 menit
<i>InceptionV3</i>	19 menit	33 menit
<i>Xception</i>	36 menit	1 jam 3 menit

prediksi pada model dan mencegah terjadinya *overfitting*.

Untuk mendapatkan model CNN dengan performa terbaik, penelitian ini membandingkan hasil prediksi dari enam model CNN yaitu *ResNet50*, *InceptionV3*, *Xception*, *DenseNet121*, *MobileNet*, dan *InceptionResNetV2*. Nilai *learning rate* dan *epoch* yang digunakan pada saat pelatihan sama untuk semua percobaan yang dilakukan.

Pada saat pelatihan dilakukan, waktu komputasi juga dihitung untuk mengetahui perbandingan waktu yang dibutuhkan oleh masing-masing model CNN dalam menyelesaikan tahap pelatihan. Waktu komputasi yang dibutuhkan keenam model CNN dapat dilihat pada Tabel 1 dan Tabel 2. Ukuran gambar data latih yang digunakan pada Tabel 1 adalah 128x128 dan 150x150. Sedangkan pada Tabel 2 menggunakan ukuran gambar 224x224 dan 229x229.

Hasil percobaan yang terdapat pada Tabel 1 dan Tabel 2 menunjukkan bahwa model *MobileNet* memiliki waktu pelatihan paling singkat untuk semua ukuran data latih. Hal ini dapat terjadi karena arsitektur *MobileNet* didesain sangat efisien sehingga model yang dibangun jauh lebih kecil dan jumlah parameter yang dimiliki lebih sedikit jika dibandingkan dengan keenam model CNN lainnya.

Hasil pengujian pada kedua tabel menunjukkan bahwa semakin besar ukuran gambar yang dilatih maka waktu pelatihan yang dibutuhkan juga semakin lama. Model *DenseNet121* dan *InceptionResNetV2* membutuhkan waktu pelatihan paling lama jika dibandingkan dengan keempat

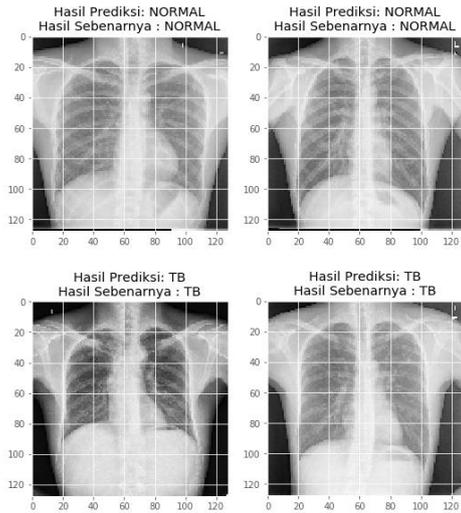
model CNN lainnya terutama apabila ukuran gambar besar seperti 299x299.

Tahap berikutnya adalah proses pengujian yang menggunakan data uji untuk mengetahui sejauh mana model *classifier* berhasil melakukan klasifikasi dengan tepat. Masing-masing model akan diukur performanya dengan menghitung keakuratan dari prediksinya. Contoh hasil prediksi pada proses pengujian dapat dilihat pada Gambar 5. Setelah melalui proses pengujian, masing-masing gambar pada data uji akan mendapatkan tambahan label hasil prediksi berupa nama kelasnya, normal atau TB.

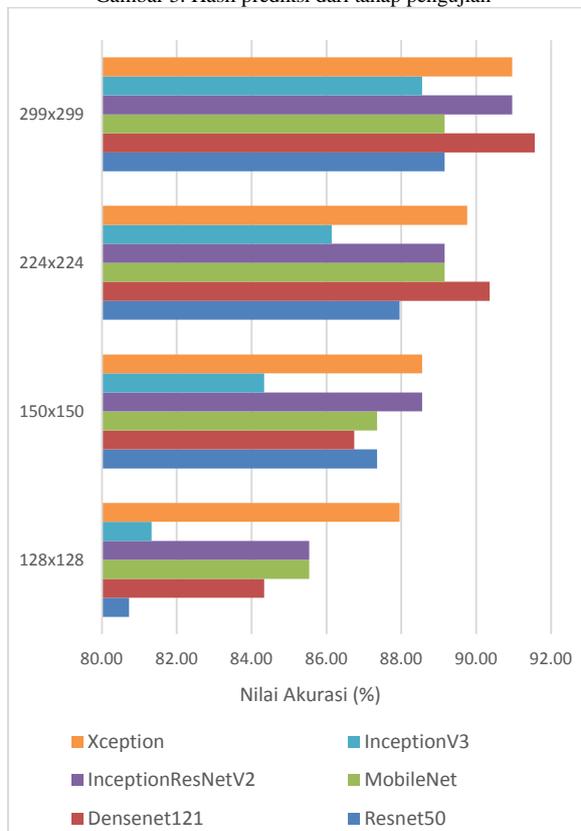
Hasil pengujian model CNN berupa *confusion matrix* yang kemudian dapat digunakan untuk menghitung nilai akurasi menggunakan Persamaan 1. Nilai akurasi yang didapatkan masing-masing model CNN selama proses pengujian dapat dilihat pada Gambar 6. Masing-masing model CNN diuji dengan menggunakan empat ukuran gambar masukan yang berbeda yaitu 128x128, 150x150, 224x224 dan 299x299.

Berdasarkan hasil pengujian yang terdapat pada Gambar 6, terlihat bahwa perbedaan ukuran gambar dan model CNN yang digunakan sangat berpengaruh terhadap nilai akurasi yang dihasilkan. Semakin besar ukuran gambar, nilai akurasi yang didapatkan juga akan semakin tinggi. Hal ini berlaku untuk semua model CNN yang diujikan. Untuk data latih berukuran 128x128 yang merupakan ukuran terkecil pada percobaan ini, model *ResNet50* memiliki nilai akurasi terendah yaitu 80,72% dan nilai akurasi tertinggi dihasilkan oleh model *Xception* dengan 87,95%. Pada pengujian dengan menggunakan ukuran gambar 150x150, model *InceptionResNetV2* dan *Xception* memiliki akurasi terbaik dengan nilai yang sama yaitu 88,55%. Sedangkan pada data latih berukuran 224x224 dan 299x299, nilai akurasi tertinggi didapatkan dengan menggunakan model *DenseNet121* dengan nilai akurasi sebesar 90,36% dan 91,57%.

Dari Gambar 6 juga terlihat bahwa model *InceptionV3* dan *ResNet50* selalu menghasilkan nilai akurasi yang rendah jika dibandingkan dengan model CNN yang lain pada saat pengujian. Kedua model tersebut kurang cocok apabila diimplementasikan ke dalam sistem deteksi penyakit TB dimana sistem ini umumnya sangat memerlukan nilai akurasi yang tinggi. Sebaliknya, model *DenseNet121* dengan teknik *dense connection* ternyata mampu menghasilkan performa terbaik untuk mendeteksi TB jika dibandingkan model CNN lainnya terutama bila gambar yang digunakan berukuran besar yaitu 224x224 dan 299x299. Untuk sistem yang menggunakan ukuran gambar masukan yang lebih kecil, model *Xception* dengan teknik *Depthwise Separable Convolutions* dapat menjadi pilihan



Gambar 5. Hasil prediksi dari tahap pengujian



Gambar 6. Perbandingan nilai akurasi masing-masing model CNN dengan berbagai ukuran gambar

pertama karena nilai akurasi yang dihasilkan model ini paling tinggi daripada model CNN lainnya.

4. KESIMPULAN

Metode *Convolutional Neural Network* (CNN) dapat digunakan untuk mendeteksi penyakit TB dengan data masukan berupa gambar foto rontgen dada. Penelitian ini menggunakan beberapa model CNN yaitu *ResNet50*, *DenseNet121*, *MobileNet*, *InceptionV3*, *Xception* dan *InceptionResNetV2* untuk mendapatkan model CNN yang mampu menghasilkan performa terbaik dalam memprediksi kelas normal maupun TB.

Pengujian dilakukan dengan menggunakan ukuran dataset yang bervariasi untuk mengetahui pengaruh ukuran gambar masukan terhadap performa yang dihasilkan masing-masing model CNN dalam mendeteksi penyakit TB. Nilai akurasi digunakan untuk mengevaluasi performa model CNN yang telah dilatih dan waktu komputasi dihitung untuk mengetahui waktu yang dibutuhkan masing-masing model CNN untuk melakukan proses pelatihan.

Hasil pengujian menunjukkan bahwa semakin besar ukuran dataset maka nilai akurasi yang didapatkan juga akan semakin tinggi namun sebaliknya waktu komputasi yang dibutuhkan juga akan semakin lama. Model *DenseNet121* sangat cocok digunakan pada sistem yang mampu mendeteksi foto rontgen dengan penyakit TB dimana sistem pasti membutuhkan akurasi yang tinggi. Nilai akurasi yang didapatkan dengan *DenseNet121* lebih tinggi jika dibandingkan dengan model CNN lainnya yaitu sebesar 91,57%. Sedangkan model *MobileNet* membutuhkan waktu paling sedikit untuk menyelesaikan proses pelatihan terutama bila ukuran gambar yang digunakan 128x128 maupun 150x150.

DAFTAR PUSTAKA

- BISWAS, B., GHOSH, S. K., BHATTACHARYYA, S., PLATOS, J., SNASEL, V., & CHAKRABARTI, A. 2020. Chest X-ray enhancement to interpret pneumonia malformation based on fuzzy soft set and Dempster-Shafer theory of evidence. *Applied Soft Computing Journal*, 86, 105889.
- CHAN, Y. H., ZENG, Y. Z., WU, H. C., WU, M. C., & SUN, H. M. 2018. Effective pneumothorax detection for chest X-ray images using local binary pattern and support vector machine. *Journal of Healthcare Engineering*, 2018.
- CHAVES, E., GONÇALVES, C. B., ALBERTINI, M. K., LEE, S., JEON, G., & FERNANDES, H. C. 2020. Evaluation of transfer learning of pre-trained CNNs applied to breast cancer detection on infrared images. *Applied Optics*, 59(17), E23.
- CHOLLET, F. 2017. Xception: Deep learning with depthwise separable convolutions. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017-Janua*, 1800–1807.
- GOODFELLOW, I., BENGIO, Y., dan COURVILLE, A. 2016. *Deep Learning*. MIT Press.
- HAN, J. dan KAMBER, M. 2001. *Data Mining: Concepts and Techniques Tutorial*. Morgan Kaufman Publisher. San Francisco.

- HE, K., ZHANG, X., REN, S., & SUN, J. 2016. Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem, 770–778.
- HOWARD, A. G., ZHU, M., CHEN, B., KALENICHENKO, D., WANG, W., WEYAND, T., ADAM, H. 2017. *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*.
- HUANG, G., LIU, Z., VAN DER MAATEN, L., & WEINBERGER, K. Q. 2017. Densely connected convolutional networks. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-Janua, 2261–2269.
- JAEGER, S., KARARGYRIS, A., CANDEMIR, S., FOLIO, L., SIEGELMAN, J., CALLAGHAN, F., & MCDONALD, C. J. 2013. Automatic Tuberculosis Screening Using Chest Radiographs. *IEEE Trans Med Imaging*. 2014 Feb;33(2):233-45.
- JAEGER, S., CANDEMIR, S., ANTANI, S., WANG, Y. J., LU, P., & THOMA, G. 2014. Two public chest X-ray datasets for computer-aided screening of pulmonary diseases. *Quantitative Imaging in Medicine and Surgery*, 4(6), 475–477.
- KAUSHAL, C., BHAT, S., KOUNDAL, D., & SINGLA, A. 2019. Recent Trends in Computer Assisted Diagnosis (CAD) System for Breast Cancer Diagnosis Using Histopathological Images. *Irbm*, 40(4), 211–227.
- MAHMUD, K. H., ADIWIJAYA, & FARABY, S. AL. 2019. Klasifikasi Citra Multi-Kelas Menggunakan Convolutional Neural Network. *E-Proceeding of Engineering*, 6(1), 2127–2136.
- NITHYA, A., APPATHURAI, A., VENKATADRI, N., RAMJI, D. R., & ANNA PALAGAN, C. 2020. Kidney disease detection and segmentation using artificial neural network and multi-kernel k-means clustering for ultrasound images. *Measurement: Journal of the International Measurement Confederation*, 149, 106952.
- PEREZ, L., & WANG, J. 2017. The Effectiveness of Data Augmentation in Image Classification using Deep Learning.
- PURNAMA, I. K. E., HERNANDA, A. K., RATNA, A. A. P., NURTANIO, I., HIDAYATI, A. N., PURNOMO, M. H., RACHMADI, R. F. 2019. Disease Classification based on Dermoscopic Skin Images Using Convolutional Neural Network in Teledermatology System. 2019 International Conference on Computer Engineering, Network, and Intelligent Multimedia, CENIM 2019 - Proceeding, 2019-November, 1–5.
- RAJAN, P. G., & SUNDAR, C. 2019. Brain Tumor Detection and Segmentation by Intensity Adjustment. *Journal of Medical Systems*, 43(8), 1–13.
- RAJPURKAR, P., IRVIN, J., ZHU, K., YANG, B., MEHTA, H., DUAN, T., & Ng, A. Y. 2017. CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. 3–9.
- SANTOSH, K. C., VAJDA, S., ANTANI, S., & THOMA, G. R. 2016. Edge map analysis in chest X-rays for automatic pulmonary abnormality screening. *International Journal of Computer Assisted Radiology and Surgery*, 11(9), 1637–1646.
- SANTOSO, A., & ARIYANTO, G. 2018. Implementasi Deep Learning Berbasis Keras untuk Pengenalan Wajah. *Emitor: Jurnal Teknik Elektro*, 18(01), 15–21.
- SAVERA, T. R., SURYAWAN, W. H., & SETIAWAN, A. W. 2020. Deteksi Dini Kanker Kulit Menggunakan K-NN dan Convolutional Neural Network. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)* Vol. 7, No. 2, April 2020, hlm. 373-378.
- SIMONYAN, K., & ZISSERMAN, A. 2015. Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–14.
- SAXENA, P., MAHESHWARI, A., TAYAL, S., & MAHESHWARI, S. 2019. Predictive modeling of brain tumor: A Deep learning approach.
- SZEGEDY, C., VANHOUCKE, V., IOFFE, S., SHLENS, J., & WOJNA, Z. 2016. Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem, 2818–2826.
- SZEGEDY, C., IOFFE, S., VANHOUCKE, V., & ALEMI, A. A. 2017. Inception-v4, inception-ResNet and the impact of residual connections on learning. *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, 4278–4284.
- TALO, M. 2019. Convolutional Neural Networks for Multi-class Histopathology Image Classification.
- VARSHNI, D., THAKRAL, K., AGARWAL, L., NIJHAWAN, R., & MITTAL, A. 2019.

Pneumonia Detection Using CNN based Feature Extraction. Proceedings of 2019 3rd IEEE International Conference on Electrical, Computer and Communication Technologies, ICECCT 2019, 1–7.

WHO. 2019. Global Tuberculosis report country profile 2019. Publication, 63(10), 476.

ZHANG, C., SARGENT, I., PAN, X., GARDINER, A., HARE, J., & ATKINSON, P. M. 2018. VPRS-Based regional decision fusion of CNN and MRF classifications for very fine resolution remotely sensed images. IEEE Transactions on Geoscience and Remote Sensing, 56(8), 4507–4521.

Halaman ini sengaja dikosongkan