

Fall Detection With Motion Tracking

Jonathan Edmund Kusnadi
Departemen Ilmu Komputer dan Elektronika
Universitas Gadjah Mada
Sleman, Indonesia
jonathan.edmund1202@mail.ugm.ac.id

Abstract—Deteksi jatuh merupakan area penelitian yang penting bagi masyarakat lanjut usia karena jatuh dapat menyebabkan cedera serius bahkan kematian. Dalam paper ini, kami mengusulkan sebuah program yang mengimplementasikan sistem deteksi jatuh untuk individu lanjut usia menggunakan teknik tracking.

Index Terms—COVID-19, Deep Learning, CNN, Detection, Multi-layered CNN, X-Ray

I. INTRODUCTION

Deteksi jatuh merupakan area penelitian penting bagi masyarakat lanjut usia karena jatuh dapat menyebabkan cedera serius bahkan kematian. Dalam paper ini, kami mengusulkan sebuah program yang mengimplementasikan sistem deteksi jatuh untuk individu lanjut usia menggunakan teknik tracking.

Selama beberapa dekade terakhir, banyak solusi yang telah dikembangkan untuk mendeteksi insiden jatuh secara otomatis. Teknologi pendeteksi jatuh biasanya mengandalkan tiga jenis perangkat: perangkat yang dapat dikenakan, sensor sekitar, dan kamera penglihatan. Perangkat yang dapat dikenakan bersifat mengganggu, dan perangkat tersebut harus dipasang dalam hubungan yang relatif dekat dengan subjek manusia. Sangat berat bagi para lansia untuk memakai perangkat fisik untuk jangka waktu yang lama. Perangkat berbasis suasana, seperti sensor audio dan getaran, telah digunakan dalam beberapa sistem pendeteksi jatuh. Perangkat berbasis suasana tidak terlalu mengganggu; namun, sifatnya yang kurang visibel membawa beberapa tantangan dalam membedakan jatuhnya benda mati dengan jatuhnya manusia. Pendekatan berbasis penglihatan memberikan solusi non-invasif dan andal untuk deteksi jatuh. Selain itu, kamera video telah banyak dipasang di berbagai bangunan umum dan fasilitas kesehatan untuk tujuan pengawasan. Ini adalah solusi berbiaya rendah untuk sistem deteksi jatuh yang praktis [1].

Program yang diusulkan ini menggunakan algoritma deteksi objek [fill] untuk mengidentifikasi dan melacak individu dalam suatu ruangan. Kemudian, program menganalisis gerakan dari individu yang dilacak untuk menentukan apakah terjadi jatuh atau tidak. Tujuan dari program ini adalah memberikan sistem deteksi jatuh yang efektif dan akurat yang dapat dengan mudah diimplementasikan di berbagai lingkungan, termasuk panti jompo dan fasilitas perawatan lanjut usia. Sistem yang diusulkan juga dapat digunakan untuk memberikan peringatan secara real-time kepada pengasuh atau layanan darurat dalam kejadian jatuh. Efektivitas program yang diusulkan dievaluasi menggunakan dataset jatuh di dunia nyata, dan hasilnya

menunjukkan bahwa sistem yang diusulkan mencapai tingkat akurasi dan ketahanan yang tinggi dalam mendeteksi jatuh. Paper ini menyajikan kontribusi berharga bagi bidang deteksi jatuh, dan program yang diusulkan berpotensi untuk meningkatkan keselamatan dan kesejahteraan individu lanjut usia.

II. METHODOLOGY

A. Dataset

Pada penelitian ini awalnya menggunakan dataset yang digunakan adalah dataset dari Joseph Paul Cohen, Paul Morrison dan Lan Dao yang berisi 156 gambar X-Ray paru-paru. Namun untuk membangun model CNN yang lebih baik, dataset tersebut ditambahkan dengan dataset dari Muhammad E. H. Chowdhury et al. yang berisi 13964 gambar[3][4]. Namun dataset tersebut memiliki kelas yang tidak seimbang sehingga dilakukan proses undersampling untuk mendapatkan dataset yang seimbang. Dataset yang digunakan pada penelitian ini adalah dataset yang sudah dilakukan undersampling sehingga dataset yang digunakan memiliki jumlah 8819 gambar dengan distribusi data pada Fig 1 dan sampel dari dataset pada Fig 2 dan Fig 3.

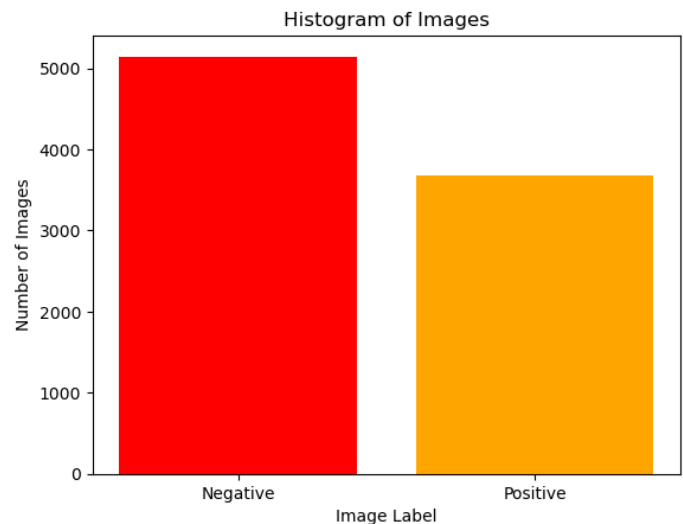


Fig. 1. Distribusi kelas pada dataset

Dataset kemudian di pisah menjadi tiga bagian yaitu training set, validation set, dan test set. Training set dan validation

Dataset Positive Samples

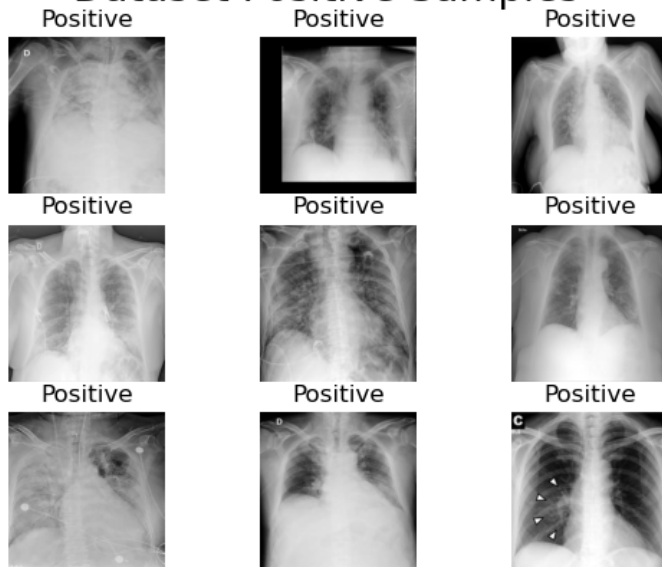


Fig. 2. Sampel positif dataset

Dataset Negative Samples

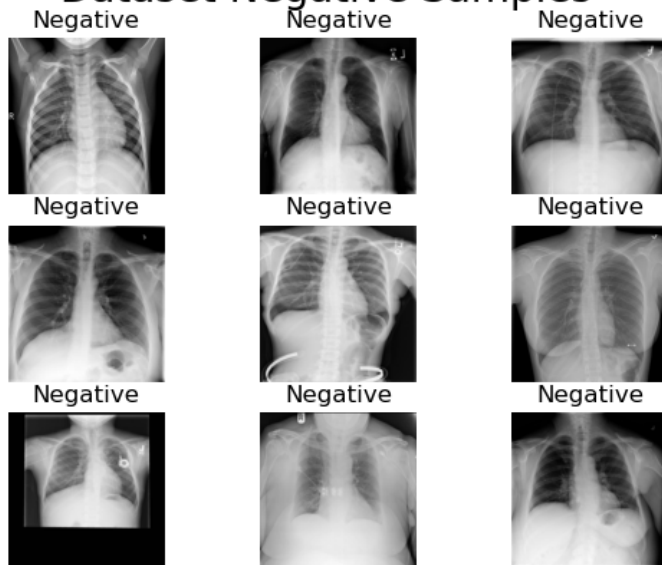


Fig. 3. Sampel negatif dataset

set digunakan untuk melatih model CNN, sedangkan test set digunakan untuk mengevaluasi performa dari model CNN. Training set sejumlah 60% dari dataset, validation set sejumlah 20% dari dataset, dan test set sejumlah 20% dari dataset.

B. Preprocessing

Pada tahap preprocessing, gambar-gambar pada dataset diubah ukurannya agar seragam. Gambar-gambar pada dataset juga diubah menjadi grayscale karena gambar X-Ray pada dataset tersebut masih dalam bentuk RGB. Gambar-gambar

pada dataset juga di normalisasi dengan membagi setiap pixel dengan 255. Hal ini dilakukan untuk mengurangi waktu yang dibutuhkan untuk melatih model CNN.

Normalisasi pada gambar (image normalization) sangat penting untuk dilakukan dalam pelatihan model Convolutional Neural Network (CNN) karena dapat meningkatkan performa model dengan mengurangi variabilitas dalam data masukan. Menurut literatur, normalisasi pada gambar dalam CNN dapat didefinisikan sebagai *"the process of transforming input images so that they have similar statistical properties, which can improve the performance of the model"* [5].

C. CNN Architecture

Arsitektur dari model CNN yang diusulkan terdiri dari lima belas lapisan. Arsitektur diawali dengan lapisan konvolusi dua dimensi dengan input gambar 200 x 200 piksel yang kemudian dilanjutkan dengan lapisan *max pooling*. Kemudian dilanjutkan dengan lapisan konvolusi, lapisan normalisasi, dan *max pooling*, ketiga lapisan tersebut diulang sebanyak tiga kali. Hasil kemudian di *flatten* dan dimasukkan dalam dua lapisan *dense* atau *neural network* yang memiliki 128 *neuron*. Kemudian dilanjutkan dengan lapisan *dropout* yaitu penghapusan neuron secara random sebagai upaya untuk memperkecil kemungkinan *overfitting*. Lapisan *dropout* diikuti dengan lapisan *dense* yang memiliki satu *neuron* yang merupakan output dari model CNN. Arsitektur dari model CNN dapat dilihat pada Fig 3.

Parameter yang digunakan pada model CNN adalah sebagai berikut:

- **Optimizer:** Adam
- **Learning rate:** 0.001
- **Loss function:** Binary Crossentropy
- **Batch size:** 32
- **Epoch:** 50

Untuk mengoptimalkan efisiensi dari tahap pelatihan model CNN, digunakan beberapa *callback* yaitu *early stopping* dan *ReduceLROnPlateau*.

• Early Stopping

Early stopping digunakan untuk menghentikan proses pelatihan model CNN ketika model sudah tidak mengalami peningkatan performa dalam 5 epoch. Callback Early Stopping adalah teknik yang digunakan dalam pelatihan model Convolutional Neural Network (CNN) untuk menghentikan pelatihan saat performa model tidak lagi meningkat. Teknik ini membantu mencegah *overfitting* dan mempercepat proses pelatihan. Menurut literatur, callback early stopping pada CNN dapat didefinisikan sebagai *"a technique in deep learning that monitors the validation loss of a CNN model during training and stops the training process when the validation loss no longer improves over a certain number of epochs"* [6].

• ReduceLROnPlateau

ReduceLROnPlateau adalah teknik yang digunakan

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 196, 196, 32)	832
max_pooling2d_1 (MaxPooling2D)	(None, 98, 98, 32)	0
conv2d_2 (Conv2D)	(None, 98, 98, 64)	18496
batch_normalization (Batch Normalization)	(None, 98, 98, 64)	256
max_pooling2d_2 (MaxPooling2D)	(None, 49, 49, 64)	0
conv2d_3 (Conv2D)	(None, 49, 49, 128)	73856
batch_normalization_1 (Batch Normalization)	(None, 49, 49, 128)	512
conv2d_4 (Conv2D)	(None, 49, 49, 128)	147584
batch_normalization_2 (Batch Normalization)	(None, 49, 49, 128)	512
max_pooling2d_3 (MaxPooling2D)	(None, 24, 24, 128)	0
flatten (Flatten)	(None, 73728)	0
dense (Dense)	(None, 128)	9437312
dense_1 (Dense)	(None, 128)	16512
dropout (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 1)	129

=====
 Total params: 9,696,001
 Trainable params: 9,695,361
 Non-trainable params: 640

Fig. 4. Arsitektur model CNN

dalam pelatihan model Convolutional Neural Network (CNN) untuk menurunkan laju belajar (learning rate) saat performa model tidak lagi meningkat. Teknik ini membantu mencegah terjebak di dalam lokal minimum dan mempercepat proses konvergensi. Menurut literatur, callback ReduceLROnPlateau pada CNN dapat didefinisikan sebagai *"a callback function in deep learning that monitors the validation loss of a CNN model during training and reduces the learning rate of the optimizer when the validation loss stops improving for a certain number of epochs"* [7].

Parameter yang digunakan pada *callback* yang digunakan tertera dalam Table 1.

III. RESULTS AND DISCUSSION

Pada penelitian ini, digunakan perangkat komputasi GPU NVIDIA GeForce GTX 1650. Pelatihan model selesai pada epoch ke-27 disebabkan *callback early stopping*. Grafik *loss* dan akurasi dari model CNN dapat dilihat pada Fig 5 dan Fig 6.

TABLE I
CALLBACK PARAMETER

Callback	Parameter	Value
Early Stopping	min_delta	0.01
	patience	10
	restore_best_weights	True
ReduceLROnPlateau	factor	0.1
	patience	5
	min_delta	0.01

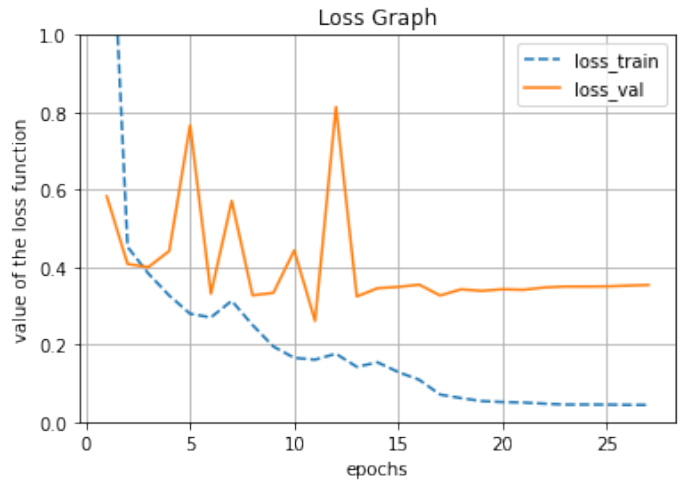


Fig. 5. Grafik *loss* dari model CNN

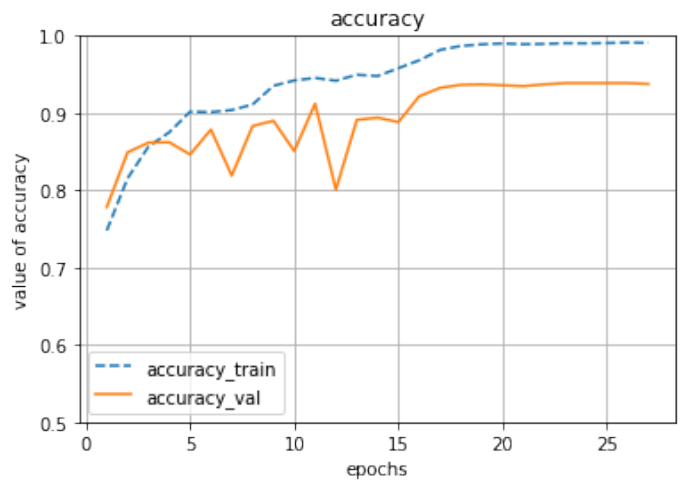


Fig. 6. Grafik akurasi dari model CNN

Grafik *loss* menunjukkan bahwa model CNN yang diusulkan memiliki *loss* yang semakin menurun seiring dengan bertambahnya epoch walaupun terjadi fluktuasi yang signifikan pada awal. Hal tersebut menunjukkan bahwa model CNN yang diusulkan semakin baik dalam melakukan klasifikasi gambar X-ray paru-paru.

Grafik akurasi menunjukkan bahwa model CNN yang diusulkan memiliki akurasi yang semakin meningkat seiring dengan bertambahnya epoch. Hal tersebut menunjukkan bahwa model CNN yang diusulkan semakin baik dalam melakukan klasifikasi gambar X-ray paru-paru.

Dari hasil pelatihan model CNN, didapatkan akurasi sebesar 94% dan *loss* sebesar 23%. Hasil tersebut menunjukkan bahwa model CNN yang diusulkan memiliki performa yang baik dalam melakukan klasifikasi gambar.

IV. CONCLUSION

Pada penelitian ini, model Convolutional Neural Network (CNN) terbukti dapat melakukan klasifikasi Covid-19 berdasarkan gambar X-ray paru-paru. Model CNN yang diusulkan memiliki akurasi sebesar 94% dan *loss* sebesar 23%. Hasil tersebut menunjukkan bahwa model CNN yang diusulkan memiliki performa yang baik dalam melakukan klasifikasi gambar X-ray paru-paru.

Untuk penelitian selanjutnya, dapat dilakukan penelitian dengan menggunakan dataset yang lebih besar dan beragam. Selain itu, dapat dilakukan penelitian dengan melakukan pengolahan citra seperti *histogram equalization* beserta metode *preprocessing* lainnya pada dataset. Untuk menghasilkan akurasi yang lebih tinggi, dapat dilakukan penelitian dengan menggunakan model CNN yang lebih kompleks seperti ResNet, DenseNet, dan EfficientNet. Dapat juga dilakukan penelitian dengan menggunakan *transfer learning* untuk meningkatkan akurasi dari model CNN yang diusulkan.

V. ADDITIONAL INFORMATION

File *Google Colab* atau *Jupyter Notebook* dapat diakses pada tautan berikut: <https://colab.research.google.com/drive/1dv7vCyRvFQis5hKDv8Sr2cJx3V8V1lo5>

REFERENCES

- [1] D. Ros and R. Dai, "A Flexible Fall Detection Framework Based on Object Detection and Motion Analysis," 2023 International Conference on Artificial Intelligence in Information and Communication (ICAIC), Bali, Indonesia, 2023, pp. 063-068, doi: 10.1109/ICAIC57133.2023.10066990.
- [2] Y. LeCun, Y. Bengio and G. Hinton, "Deep learning," in *Nature*, vol. 521, no. 7553, pp. 436-444, May 2015, doi: 10.1038/nature14539.
- [3] M.E.H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M.A. Kadir, Z.B. Mahbub, K.R. Islam, M.S. Khan, A. Iqbal, N. Al-Emadi, M.B.I. Reaz, M. T. Islam, "Can AI help in screening Viral and COVID-19 pneumonia?" *IEEE Access*, Vol. 8, 2020, pp. 132665 - 132676. Paper link
- [4] Rahman, T., Khandakar, A., Qiblawey, Y., Tahir, A., Kiranyaz, S., Kashem, S.B.A., Islam, M.T., Maadeed, S.A., Zughair, S.M., Khan, M.S. and Chowdhury, M.E., 2020. Exploring the Effect of Image Enhancement Techniques on COVID-19 Detection using Chest X-ray Images. Paper Link
- [5] Y. Chen, Z. Shi, B. Zhou, and J. Zhu, "Deep Learning for Computer Vision: Theory, Algorithms, and Applications," Springer, 2019, pp. 103-105.

- [6] Y. Chen, Z. Shi, B. Zhou, and J. Zhu, "Deep Learning for Computer Vision: Theory, Algorithms, and Applications," Springer, 2019, pp. 153-154.
- [7] S. Huang, "Deep Learning for Image Recognition," CRC Press, 2019, pp. 133-134.