

Klasifikasi Retak Ban Kendaraan Menggunakan Arsitektur ResNet50

Iwansyah Edo Hendrawan¹, Riza Ibnu Adam², Chaerur Rozikin³

¹ Universitas Singaperbangsa Karawang, Iwansyah.edo18106@student.unsika.ac.id, Karawang, Indonesia

² Universitas Singaperbangsa Karawang, Riza.adam@staff.unsika.ac.id, Karawang, Indonesia

³ Universitas Singaperbangsa Karawang, Chaerur.rozikin@staff.unsika.ac.id, Karawang, Indonesia

Informasi Makalah

Submit : Oktober 30, 2022
Revisi : November 23, 2022
Diterima : Juni 1, 2023

Kata Kunci :

Ban
Deep Learning
ResNet50

Abstrak

Ban adalah salah satu komponen penting yang hampir ada pada semua kendaraan. Ban berfungsi untuk meneruskan fungsi pengereman serta memberikan kenyamanan kepada pengguna. Seiring usia ban yang sering digunakan dapat mengalami kerusakan yang disebabkan oleh berbagai faktor seperti jalan yang dilalui ataupun cuaca yang diterima oleh ban kendaraan. Retak pada ban merupakan salah satu kerusakan yang dapat terjadi pada ban dimana hal ini dapat menjadi sebuah tanda dimana ban yang digunakan telah kehilangan sifat lenturnya. Ban yang telah mengalami penurunan pada kelenturannya menandakan bahwa traksi yang dimiliki ban berkurang sehingga jika diabaikan dapat menjadi sumber potensi kecelakaan pengendara. Perawatan yang tepat pada ban sangat diperlukan sehingga ban yang digunakan dapat dalam keadaan optimal sehingga tidak akan melukai pengendara. Dalam proses perawatan ban kendaraan saat ini masih dilakukan secara manual yang bergantung ke mata telanjang manusia sehingga dapat terjadi kemungkinan human error. Pada penelitian ini dilakukan pengembangan model dimana untuk melakukan klasifikasi terhadap ban kendaraan yang retak sebagai upaya mengurangi kemungkinan human error. Model yang dibangun menggunakan arsitektur ResNet50 yang merupakan salah satu arsitektur CNN yang merupakan algoritma dalam deep learning. Arsitektur ResNet50 menggunakan model Transfer Learning dimana model tidak memerlukan pelatihan atau penyesuaian layer. Dari 6 skenario parameter tuning yang dilakukan model mendapat nilai terbaik untuk akurasi 94%, presisi 94% dan recall 94%.

Abstract

Tires are one of the important components that exist in almost all vehicles. The tire's purpose is to continue the braking function while also providing comfort to the user. With age, tires that are often used can experience damage caused by various factors, such as the road being traversed or the weather received by the vehicle's tires. Cracks in tires are one of the damages that can occur to tires, and this can be a sign that the tires used have lost their flexible properties. Tires that have decreased

in flexibility indicate that the traction they have is reduced, so if they are neglected, they can be a source of potential accidents for motorists. Proper tire maintenance is essential so that the tires used are in optimal condition and do not injure the rider. In the current vehicle tire maintenance process, it is still done manually, which depends on the human eye, so there is a possibility of human error. In this research, a model was developed to classify cracked tires in an effort to reduce the possibility of human error. The model built uses the ResNet50 architecture, which is one of the CNN architectures, which is an algorithm in deep learning. The ResNet50 architecture uses a transfer learning model where the model does not require training or layer adjustments. Of the 6 parameter tuning scenarios, the model got the best score for 94% accuracy, 94% precision, and 94% recall.

1. Pendahuluan

Ban adalah komponen yang hampir dimiliki oleh setiap kendaraan. Ban berfungsi sebagai penerus fungsi rem dan meneruskan traksi dari mesin kendaraan ke permukaan jalan selain itu ban juga berfungsi memberikan kenyamanan pada pengguna kendaraan sehingga dapat nyaman dalam berkendara (Anshori, Mursadin, & Siswanto, 2018; Setiawan & Midyanti, 2018). Sebagai komponen yang sering digunakan ban kendaraan dapat mengalami rusak yang dipicu oleh berbagai faktor dimana buruknya kondisi jalan yang dilalui atau cuaca yang ban terima sehingga mengikis ketahanan ban (Arthono & Permana, 2022). Kerusakan pada ban kendaraan ada berbagai jenis dan salah satunya adalah retak. Retak yang muncul pada ban kendaraan sendiri merupakan sebuah tanda dimana ban telah mulai kehilangan kelenturan atau elastisitasnya (Nurjannah, 2020). Ban yang telah kehilangan kemampuannya dapat menjadi faktor kecelakaan pengemudi, hal ini disebabkan ban yang mulai kehilangan kelenturannya mempunyai daya cengkram atau traksi yang rendah sehingga jika digunakan dapat terjadi selip dimana dalam kondisi selip kendaraan sangat sulit untuk dikendalikan (Ruusen, 2021). Untuk menghindari kemungkinan faktor kecelakaan maka ban perlu mendapat perawatan dan penanganan yang tepat sehingga ban dapat digunakan dengan maksimal sebagaimana

funingsinya. Salah satu cara yang dapat dilakukan untuk perawatan ban adalah dengan melakukan pengecekan rutin. Pengecekan rutin berfungsi untuk menentukan apakah ban kendaraan masih dalam kondisi baik ataupun tidak untuk digunakan.

Dalam melakukan perawatan ban dengan cara mengecek kondisi ban saat ini masih dilakukan secara konvensional dimana cara yang dilakukan adalah dengan melihat kondisi ban kendaraan secara langsung menggunakan mata telanjang. Dalam hal ini ada beberapa kekurangan yang berupa human error dimana jika yang melakukan pengecekan pada ban sedang tidak sehat maka akan terjadi kesalahan dalam melakukan diagnosa pada ban kendaraan. Diagnosa yang salah pada ban tentu dapat menyebabkan kerugian dimana biaya perawatan kendaraan yang naik atau menjadi faktor kecelakaan kendaraan. Perkembangan pada teknologi yang digunakan oleh manusia telah berkembang sangat pesat dimana segala hal pada permasalahan yang dimiliki oleh manusia diselesaikan dengan teknologi yang telah dikembangkan (Arsal, Wardijono, & Anggraini, 2020). Pengolahan citra merupakan salah satu bidang yang berkembang dimana pengolahan citra diterapkan pada kasus deteksi objek untuk menyelesaikan permasalahan yang ada (K. B. V. Putra, Bayupati, & Arsa, 2021). Klasifikasi merupakan sebuah tindakan dalam melakukan pengenalan terhadap suatu

objek berdasarkan informasi objek sebelumnya (Ary & Rismiati, 2019; Nasution, Khotimah, & Chamidah, 2019). Klasifikasi dapat dilakukan secara konvensional ataupun dengan menggunakan algoritma yang diterapkan pada model untuk melakukan klasifikasi (Wibawa, Purnama, Akbar, & Dwiyanto, 2018). Oleh karena pada penelitian akan dilakukan pengembangan pada model yang dapat melakukan klasifikasi terhadap retak yang muncul pada ban kendaraan dengan menerapkan pengolahan citra digital sehingga kemungkinan human error dapat dikurangi.

Pada penelitian yang dilakukan oleh Sarah Lasniari (2022) yang berjudul Klasifikasi Citra Daging Babi dan Daging Sapi Menggunakan Deep Learning Arsitektur ResNet-50 dengan Augmentasi Citra mendapatkan akurasi sebesar 87%. Penelitian serupa juga dilakukan oleh Faiz Nashrullah (2020) dengan judul Investigasi Parameter Epoch Pada Arsitektur ResNet-50 Untuk Klasifikasi Pornografi mendapatkan akurasi terbaik 91%. Lalu pada penelitian lainnya yang dilakukan oleh Ivan Pratama Putra (2022) mendapat akurasi 98%.

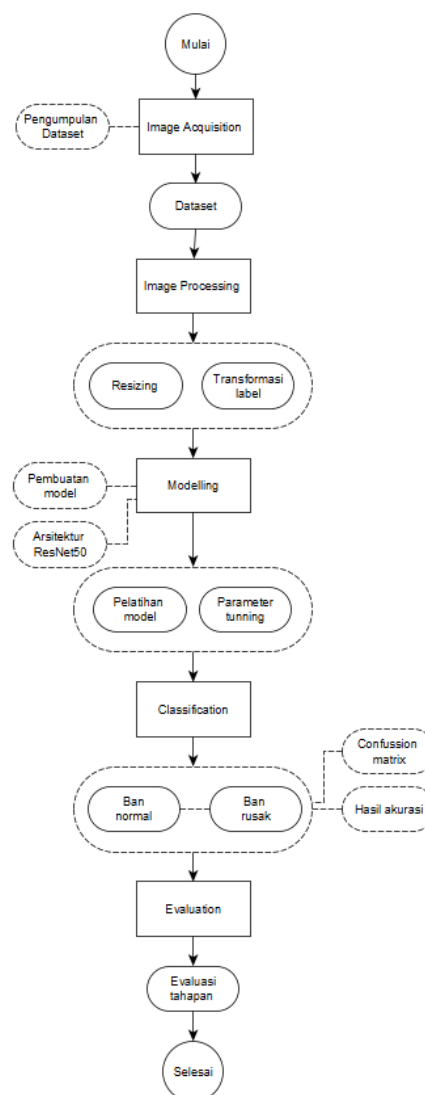
Deep learning merupakan salah satu cabang dari bidang machine learning dimana deep learning bekerja layaknya syaraf biologis dalam menentukan keputusan (Effendi, 2018; Hermawan, Lianata, & Maranto, 2022; Peryanto, Yudhana, & Umar, 2020). Algoritma deep learning melakukan klasifikasi dengan data yang dipecah menjadi dua training dan testing (Azizah, Umayah, & Fajar, 2018). ResNet merupakan arsitektur Convolution Neural Network (CNN) yang merupakan salah satu algoritma deep learning (Ridhovan & Suharso, 2022). Algoritma CNN sendiri termasuk dalam algoritma supervised learning (Wulandari, Yasin, & Widiharih, 2020).

ResNet50 merupakan salah satu model pre trained atau sebuah model yang sudah dilatih sebelumnya selain itu tidak perlu lagi

penyesuaian pada layer yang akan digunakan (Ridhovan & Suharso, 2022). ResNet50 terdiri dari 50 layer yang digunakan dalam proses konvolusinya. ResNet50 memiliki konsep skip connections dimana berfungsi untuk mencegah hilangnya fitur dalam proses konvolusi (Lasniari et al., 2022; Nashrullah et al., 2020).

2. Metode Penelitian

2.1. Metode



Gambar 1. Rancangan Penelitian

Pada penelitian ini metode yang digunakan dalam rancangan penelitian adalah computer vision dimana metode ini biasa digunakan dalam pengembangan pengolahan citra (Lumabiang, Manaha, Liem, &

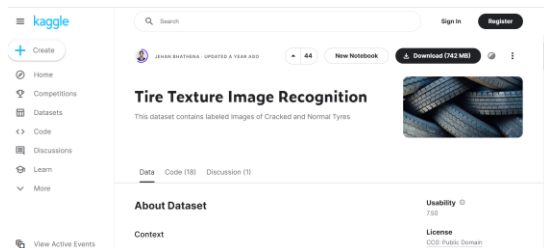
Tombeng, 2019; R. R. Putra & Antony, 2018). Kelebihan computer vision juga dapat menyesuaikan tahapan yang dibutuhkan oleh pengguna. Gambar 1 merupakan gambaran rancangan penelitian yang akan dilakukan. Pada penelitian ini dimulai dengan tahap image acquisition.

2.1.1. Image Acquisition

Pada tahap image acquisition dilakukan pengumpulan dataset yang akan digunakan pada penelitian yang dilakukan. Penelitian ini memakai dataset kaggle yang bersifat publik dan dapat di unduh siapapun. Dataset berisi 1028 gambar yang terdiri dari ban normal dan ban retak. Untuk melihat pembagian dataset dapat dilihat pada tabel 1.

Tabel 1. Dataset

Jenis Data	Data Latih	Data Uji
Ban Normal	376	115
Ban Retak	327	210
Total:	703	325



Gambar 2. Kaggle



Gambar 3. Ban Normal



Gambar 4. Ban Retak

2.1.2. Image Processing

Pada tahap image processing dilakukan penyesuaian pada gambar yang akan digunakan sebagai input model. Tahap ini dilakukan resizing atau proses pengubahan ukuran pada gambar yang diatur menjadi 100x100 pixel. Selanjutnya adalah transformasi label dimana dilakukan pengubahan nilai label pada citra menjadi 0 dan 1 untuk mempermudah dalam klasifikasi.

2.1.3. Modelling

Pada tahap ini model yang akan digunakan dalam klasifikasi akan dibuat. Pada penelitian ini menggunakan arsitektur ResNet50.

Pada perancangan model ResNet50 dilakukan parameter tuning dimana parameter yang ada pada model diubah dan disesuaikan untuk mendapatkan parameter terbaik yang bisa digunakan oleh model. Pada penelitian ini parameter yang akan disesuaikan adalah nilai pada epochs dan nilai layer dropout.

2.1.4. Classification

Pada tahap classification dilakukan pengujian model terhadap objek ingin di klasifikasikan. Dalam penelitian ini akan dilakukan 6 skenario parameter tuning untuk mendapatkan parameter terbaik untuk model. Skenario parameter tuning pada model dapat dilihat pada tabel 2 dimana dilakukan pengujian model dengan nilai parameter yang berbeda.

Tabel 2. Skenario

Skenario	Epochs	Steps Epoch	Batch Size	Dropout
1	60	5	10	0.25
2	100	5	10	0.25
3	150	5	10	0.25
4	60	5	10	0.5
5	100	5	10	0.5
6	150	5	10	0.5

2.1.5. Evaluation

Dalam tahap evaluation akan dilakukan penilaian terhadap model yang sudah dibuat apakah sudah sesuai dengan tujuan penelitian ataupun tidak. Dalam penelitian ini evaluasi model menggunakan confusion matriks dimana data dari setiap model yang telah diuji dinilai.

3. Hasil dan Pembahasan

3.1. Perancangan model

1. Pengecekan dataset

Pada langkah ini dataset yang akan digunakan perlu dilakukan pengecekan untuk mengetahui folder yang terdapat pada dataset. Untuk program dapat dilihat pada gambar 5.

```
# list folder data

# Main Dataset
main_folder="Tire Textures/training_data"
class_names=os.listdir(main_folder)
print(class_names)

# Validation Dataset
validation_folder="Tire Textures/testing_data"
val_class_names=os.listdir(validation_folder)
print(val_class_names)

['normal', 'cracked']
['normal', 'cracked']
```

Gambar 5. Listing Folder Dataset

2. Preprocessing model

Pada tahap ini dilakukan proses resizing gambar dan transformasi label. Hal ini dilakukan agar model lebih mudah dalam membaca data citra.

```
# Declaring variables
x=[] # list kosong untuk menyimpan array gambar
y=[] # list kosong untuk menyimpan label

for folder in os.listdir(main_folder):
    image_list=os.listdir(main_folder+"/"+folder)
    for img_name in image_list:
        # Loading images
        img=image.load_img(main_folder+"/"+folder+"/"+img_name,target_size=(100,100))

        # Converting to array
        img=image.img_to_array(img)

        # Transfer Learning: this is to apply preprocess of resnet
        img=resnet50.preprocess_input(img) # Optional step mengubah RGB ke BGR

        # Appending the arrays
        x.append(img) # appending image array
        y.append(class_names.index(folder)) # appending class index to the array

print("Preparing Training Dataset Completed.")
```

Gambar 6. Resizing

```
# Training Dataset
print("Training Dataset")

x=np.array(x) # Converting data ke numpy array
print(x.shape)

y=to_categorical(y) # onehot encoding label
# print(y)
print(y.shape)

# Validation Dataset
print("Validation Dataset")

val_images=np.array(val_images)
print(val_images.shape)

val_image_label=to_categorical(val_image_label) # onehot encoding label
print(val_image_label.shape)
```

Gambar 7. Transformasi Label

3. Pembagian Dataset

Pada penelitian ini model menggunakan pembagian dengan rasio 80:20 dimana 80% data training dan 20% data validasi.

4. Modelling

Pada tahap ini model yang dibuat menggunakan arsitektur ResNet50 yang terdiri dari 50 layer. Pada tahap ini dilakukan penambahan layer dropout yang dapat dilihat pada gambar 8. Layer dropout yang digunakan dalam skenario adalah 0.5 dan 0.25.

```
from tensorflow.keras.applications import ResNet50
from tensorflow.keras import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Dropout

num_classes = 2

my_new_model = Sequential()
my_new_model.add(ResNet50(include_top=False, pooling='avg', weights='imagenet'))
my_new_model.add(Dropout(0.5))
my_new_model.add(Dense(num_classes, activation='softmax'))

# Mengatur layer resnet apakah trainable atau tidak
# my_new_model.layers[0].trainable = False

my_new_model.summary()
```

Gambar 8. ResNet50

Model: "sequential_1"

Layer (type)	Output Shape	Param #
resnet50 (Functional)	(None, 2048)	23587712
dropout_1 (Dropout)	(None, 2048)	0
dense_1 (Dense)	(None, 2)	4098

Total params: 23,591,810
 Trainable params: 23,538,690
 Non-trainable params: 53,120

Gambar 9. Arsitektur ResNet50

Setelahnya adalah melakukan fit model. Dalam tahap ini nilai epochs diubah sesuai rancangan skenario yang telah direncanakan. Rancangan skenario dapat dilihat pada tabel 2.

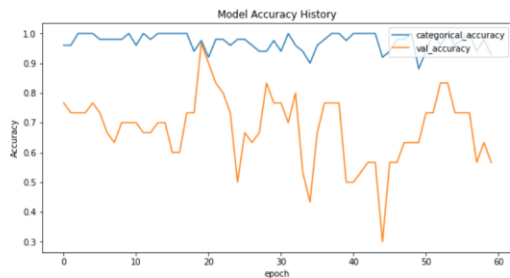
```
# Fit the Model
history = my_new_model.fit(xtrain,
    ytrain,
    epochs=100,
    steps_per_epoch=5,
    batch_size=10,
    verbose=True,
    validation_data=(val_images, val_image_label),
    validation_steps=3,
    callbacks=[callbacks, model_checkpoint_callback])
print("Fitting the model completed.")
```

Gambar 10. Fit Model

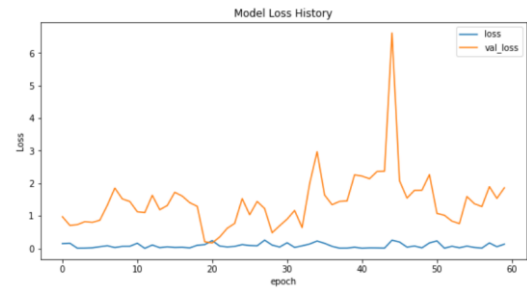
3.2. Pelatihan Model

1. Skenario 1

Pada skenario pertama, model yang menggunakan nilai epochs 60 dan nilai dropout 0.25 menghasilkan loss 13% dan akurasi 92%. Namun pada nilai validasi loss mendapat nilai 186% dan validasi akurasi 56% sehingga model yang dibuat tergolong overfitting.



Gambar 11. Akurasi Skenario 1



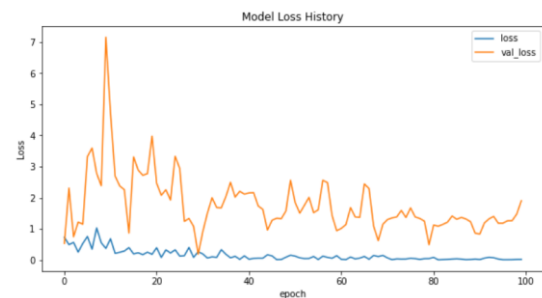
Gambar 12. Loss Skenario 1

2. Skenario 2

Pada skenario ke dua, model yang menggunakan nilai epochs 100 dan nilai dropout 0.25 menghasilkan loss 1% dan akurasi 100%. Namun pada nilai validasi loss mendapat nilai 190% dan validasi akurasi 53% sehingga model yang dibuat tergolong overfitting.



Gambar 13. Akurasi Skenario 2



Gambar 14. Loss Skenario 2

3. Skenario 3

Pada skenario ke tiga, model yang menggunakan nilai epochs 150 dan nilai dropout 0.5 menghasilkan loss 4% dan akurasi 97%. Namun pada nilai validasi loss mendapat nilai 72% dan validasi akurasi 70% sehingga model yang dibuat tergolong overfitting.



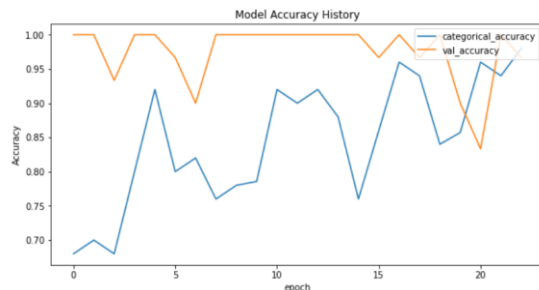
Gambar 15. Akurasi Skenario 3



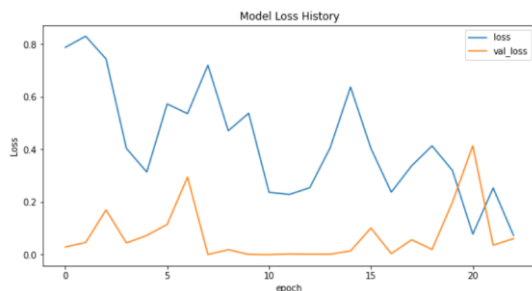
Gambar 16. Loss Skenario 3

4. Skenario 4

Pada skenario ke empat, model yang menggunakan nilai epochs 60 dan nilai dropout 0.5 menghasilkan loss 7% dan akurasi 97%. Namun pada nilai validasi loss mendapat nilai 6% dan validasi akurasi 96% sehingga model yang dibuat tergolong Good model.



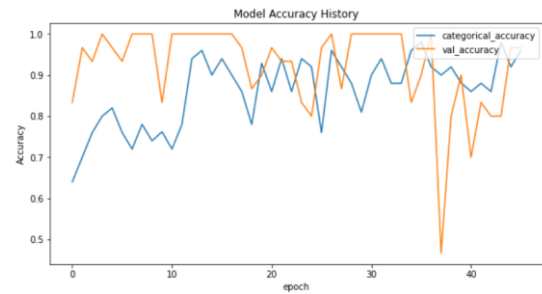
Gambar 17. Akurasi SKenario 4



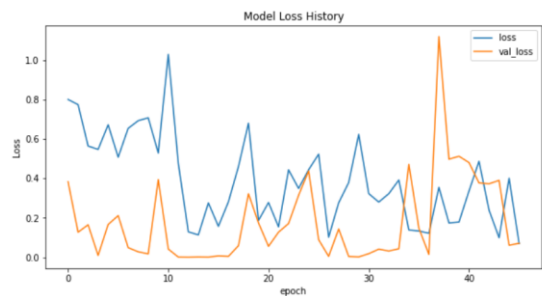
Gambar 18. Loss Skenario 4

5. Skenario 5

Pada skenario ke lima, model yang menggunakan nilai epochs 100 dan nilai dropout 0.5 menghasilkan loss 6% dan akurasi 96%. Namun pada nilai validasi loss mendapat nilai 7% dan validasi akurasi 96% sehingga model yang dibuat tergolong Good model.



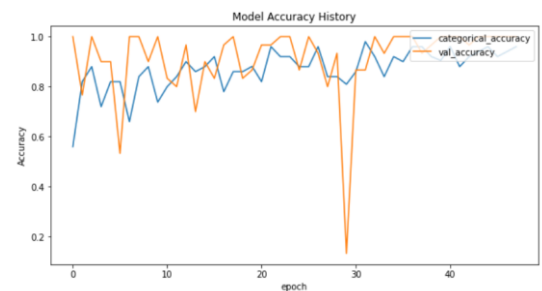
Gambar 19. Akurasi Skenario 5



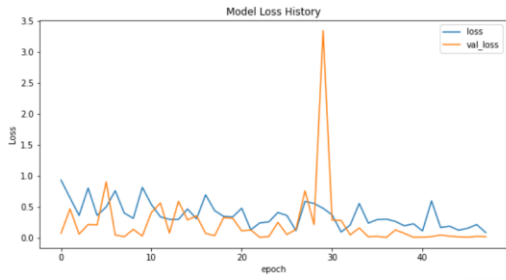
Gambar 20. Loss Skenario 5

6. Skenario 6

Pada skenario ke enam, model yang menggunakan nilai epochs 150 dan nilai dropout 0.5 menghasilkan loss 8% dan akurasi 96%. Namun pada nilai validasi loss mendapat nilai 14% dan validasi akurasi 100% sehingga model yang dibuat tergolong Good model.



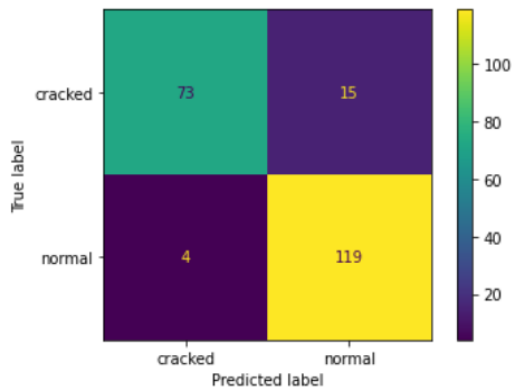
Gambar 21. Akurasi Skenario 6



Gambar 22. Loss Skenario 6

3.3. Pengujian Model

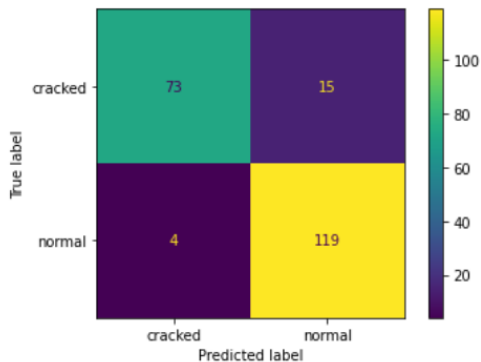
1. Skenario 1



Gambar 23. Confussion Matrix Skenario 1

Pada skenario 1 hasil pengujian yang dilakukan pada model mendapatkan akurasi sebesar 90%, presisi 92% dan recall 91% yang didapat dari confusion matrix. Pada pengujian ini model mendapat hasil yang baik. Namun pelatihan model pada skenario 1 model yang dihasilkan termasuk overfitting sehingga meski memiliki hasil yang baik pada pengujian model masih tergolong buruk.

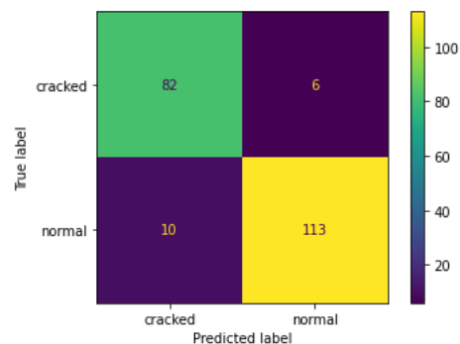
2. Skenario 2



Gambar 24. Confussion Matrix Skenario 2

Pada skenario 2 hasil pengujian yang dilakukan pada model mendapatkan akurasi sebesar 93%, presisi 93% dan recall 93% yang didapat dari confusion matrix. Pada pengujian ini model mendapat hasil yang baik. Namun pelatihan model pada skenario 2 model yang dihasilkan termasuk overfitting sehingga meski memiliki hasil yang baik pada pengujian model masih tergolong buruk.

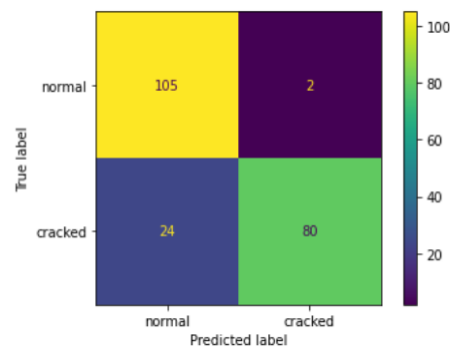
3. Skenario 3



Gambar 25. Confussion Matrix Skenario 3

Pada skenario 3 hasil pengujian yang dilakukan pada model mendapatkan akurasi sebesar 92%, presisi 92% dan recall 93% yang didapat dari confusion matrix. Pada pengujian ini model mendapat hasil yang baik. Namun pelatihan model pada skenario 3 model yang dihasilkan termasuk overfitting sehingga meski memiliki hasil yang baik pada pengujian model masih tergolong buruk.

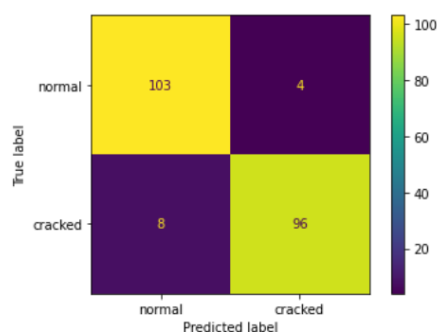
4. Skenario 4



Gambar 26. Confussion Matrix Skenario 4

Pada skenario 4 hasil pengujian yang dilakukan pada model mendapatkan akurasi sebesar 88%, presisi 89% dan recall 88% yang didapat dari confusion matrix. Pada pengujian ini model mendapat hasil yang baik. Serta saat pelatihan model pada skenario 4 model yang dihasilkan baik sehingga dengan hasil yang didapatkan pada pengujian model yang didapatkan tergolong baik.

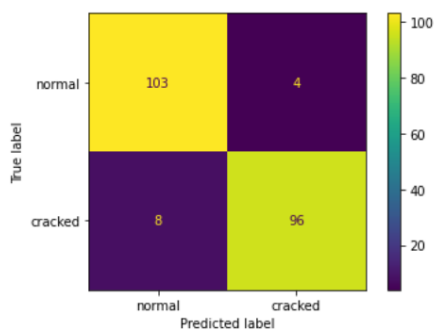
5. Skenario 5



Gambar 27. Confussion Matrix Skenario 4

Pada skenario 5 hasil pengujian yang dilakukan pada model mendapatkan akurasi sebesar 94%, presisi 94% dan recall 94% yang didapat dari confusion matrix. Pada pengujian ini model mendapat hasil yang baik. Serta saat pelatihan model pada skenario 3 model yang dihasilkan baik sehingga dengan hasil yang didapatkan pada pengujian model yang didapatkan tergolong baik.

6. Skenario 6

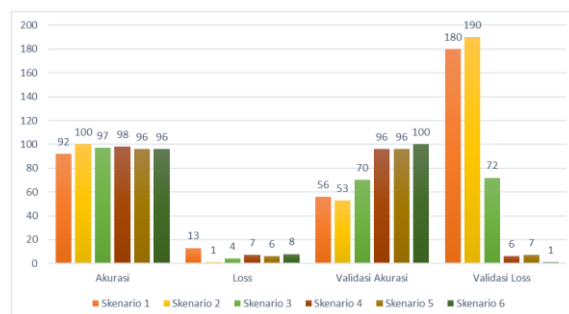


Gambar 28. Confussion Matrix Skenario 6

Pada skenario 6 hasil pengujian yang dilakukan pada model mendapatkan akurasi sebesar 84%, presisi 84% dan recall 84% yang didapat dari confusion matrix. Pada pengujian ini model mendapat hasil yang baik. Serta saat pelatihan model pada skenario 6 model yang dihasilkan baik sehingga dengan hasil yang didapatkan pada pengujian model yang didapatkan tergolong baik.

3.4. Perbandingan hasil

Pada hasil pelatihan model performa yang dihasilkan oleh skenario 1-4 mendapatkan hasil yang buruk dimana kedua model yang dibuat mengalami overfitting. Sedangkan untuk model 3-6 mendapat hasil yang baik dimana tidak terdapat overfitting ataupun underfitting pada model. Untuk perbandingan hasil dapat dilihat pada gambar 23 dan tabel 2.



Gambar 29. Pelatihan Model Skenario 1-4

Tabel 3. Pelatihan Model Skenario 1-4

Skenario	Akurasi	Loss	Validasi Akurasi	Validasi Loss
1	92%	13%	56%	180%
2	100%	1%	53%	190%
3	97%	4%	70%	72%
4	98%	7%	96%	6%
5	96%	6%	96%	7%
6	96%	8%	100%	1%

Dari pelatihan model yang dilakukan skenario 5 mendapatkan hasil terbaik pada nilai validasi.

4. Simpulan

Berdasarkan perancangan model yang telah dilakukan pada penelitian ini dengan menggunakan arsitektur ResNet50 adalah 6 skenario parameter tuning yang telah dilakukan. Skenario 1-3 yang menggunakan 3 epochs 60, 100, 150 dan nilai layer dropout 0.25 mendapatkan hasil yang buruk pada saat pelatihan model dimana semua model mengalami overfitting. Sedangkan untuk skenario 4-6 yang menggunakan epochs 60, 100, 150 dengan nilai layer dropout 0.5 mendapat nilai yang baik dimana semua model tidak mengalami overfitting atau underfitting.

Lalu pada pengujian yang telah dilakukan skenario ke lima mendapat nilai terbaik dimana model mendapatkan akurasi 94%, recall 94% dan presisi 94%. Hasil yang didapatkan pada skenario ke lima menandakan bahwa model yang dibuat telah sesuai dan sangat baik. Pada penelitian ini membuktikan bahwa arsitektur ResNet yang digunakan dalam melakukan klasifikasi terhadap ban retak dan normal mampu bekerja dengan baik sehingga dapat menjadi acuan dalam perancangan model klasifikasi dengan masalah serupa.

5. Referensi

- Anshori, M. R., Mursadin, A., & Siswanto, R. (2018). Analisis Faktor-Faktor yang Mempengaruhi Kerusakan Ban Pada Unit Dump Truck Di Pt X. *Scientific Journal of Mechanical Engineering Kinematika*, 3(2), 107–118.
- Arsal, M., Wardijono, B. A., & Anggraini, D. (2020). Face Recognition untuk Akses Pegawai Bank Menggunakan Deep Learning dengan Metode CNN. *J. Nas. Teknol. Dan Sist. Inf*, 6(1), 55–63.
- Arthono, A., & Permana, V. A. (2022). Perencanaan Perkerasan Lentur Jalan Raya Menggunakan Metode Analisa Komponen SNI 1732-1989-F Ruas Jalan Raya Mulya Sari Kecamatan Pamanukan Sampai Kecamatan Binong Kabupaten Subang Propinsi Jawa Barat. *Jurnal Komposit*, 6(1), 41–51.
- Ary, M., & Rismiati, D. A. F. (2019). Ukuran Akurasi Klasifikasi Penyakit Mesothelioma Menggunakan Algoritma K-Nearest Neighbor dan Backward Elimination. *Sains Dan Teknologi Informasi*, 5(1), 11–18.
- Azizah, L. M., Umayah, S. F., & Fajar, F. (2018). Deteksi Kecacatan Permukaan Buah Manggis Menggunakan Metode Deep Learning dengan Konvolusi Multilayer. *Semesta Teknika*, 21(2), 230–236.
- Effendi, N. (2018). Penerapan Jaringan Syaraf Tiruan Untuk Memprediksi Efektifitas Pembelajaran Dengan E-Learning. *Sains Dan Teknologi Informasi*, 4(1), 1–10.
- Hermawan, A., Lianata, L., & Maranto, A. R. K. (2022). Implementasi Machine Learning Sebagai Pengenal Nominal Uang Rupiah dengan Metode YOLOv3. *SATIN-Sains Dan Teknologi Informasi*, 8(1), 12–22.
- Lasniari, S., Jasril, J., Sanjaya, S., Yanto, F., & Affandes, M. (2022). Klasifikasi Citra Daging Babi dan Daging Sapi Menggunakan Deep Learning Arsitektur ResNet-50 dengan Augmentasi Citra. *Jurnal Sistem Komputer Dan Informatika (JSON)*, 3(4), 450–457.
- Lumabiang, V. M., Manaha, C. A. L., Liem, A. T., & Tombeng, M. (2019). Prototipe Pendeteksi Object Menggunakan Computer Vision dan Raspberry Pi. *SENSITIF: Seminar Nasional Sistem Informasi Dan Teknologi Informasi*, 1341–1351.
- Nashrullah, F., Wibowo, S. A., & Budiman, G. (2020). Investigasi Parameter Epoch Pada Arsitektur ResNet-50 untuk Klasifikasi Pornografi. *Journal of Computer, Electronic, and Telecommunication*, 1(1), 1–8.
- Nasution, D. A., Khotimah, H. H., & Chamidah, N. (2019). Perbandingan Normalisasi Data untuk Klasifikasi Wine Menggunakan Algoritma K-NN. *CESS (Journal of Computer Engineering, System and Science)*, 4(1), 78–82.
- Nurjannah, N. (2020). Kkomparasi Perpaduan SIR20/SBR dan SIR

- 3CV/BR Sebagai Base Elastomer Terhadap Karakteristik Komposit Karet untuk Telapak Ban Pejal Vulkanisir. *Jurnal Penelitian Karet*, 197–208.
- Peryanto, A., Yudhana, A., & Umar, R. (2020). Rancang Bangun Klasifikasi Citra dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network. *Format J. Ilm. Tek. Inform*, 8(2), 138.
- Putra, I. P., Rusbandi, R., & Alamsyah, D. (2022). Klasifikasi Penyakit Daun Jagung Menggunakan Metode Convolutional Neural Network. *Jurnal Algoritme*, 2(2), 102–112.
- Putra, K. B. V., Bayupati, I. P. A., & Arsa, D. M. S. (2021). Klasifikasi Citra Daging Menggunakan Deep Learning dengan Optimisasi Hard Voting. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(4), 656–662.
- Putra, R. R., & Antony, F. (2018). Sistem Computer Vision Pengenalan Pola Angka dan Operator Matematika Pada Permainan Kartu Angka Berbasis Jaringan Syaraf Tiruan Perceptron. *Jurnal Informatika Global*, 9(1).
- Ridhovan, A., & Suharso, A. (2022). Penerapan Metode Residual Network (ResNet) dalam Klasifikasi Penyakit Pada Daun Gandum. *JUPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 7(1), 58–65.
- Ruusen, A. S. (2021). Penegakan Hukum Pidana Karena Kelalaian Pengemudi Kendaraan Yang Mengakibatkan Kecelakaan Lalu Lintas. *Lex Crimen*, 10(2).
- Setiawan, R. A., & Midyanti, D. M. (2018). Rancang Bangun Alat Monitoring Tekanan Angin Ban Secara Real Time Menggunakan Metode Tsukamoto Pada Kendaraan Roda Empat. *Coding Jurnal Komputer Dan Aplikasi*, 6(3).
- Wibawa, A. P., Purnama, M. G. A., Akbar, M. F., & Dwiyanto, F. A. (2018). Metode-metode Klasifikasi. *Prosiding SAKTI (Seminar Ilmu Komputer Dan Teknologi Informasi)*, 3(1), 134–138.
- Wulandari, I., Yasin, H., & Widiharih, T. (2020). Klasifikasi citra digital bumbu dan rempah dengan algoritma convolutional neural network (cnn). *Jurnal Gaussian*, 9(3), 273–282.