

BAB 5 PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan maka diperoleh kesimpulan sebagai berikut

1. Perancangan model CNN pada arsitektur MobileNet dan MobileNetV2 serta proses pelatihan dilakukan pada infrastruktur Visual Studio Code dengan menggunakan framework Keras dan Tensorflow.
2. Nilai akurasi pelatihan dan akurasi validasi pada epoch terakhir arsitektur MobileNet yaitu 1.0000 dan 0.9474, sedangkan pada arsitektur MobileNetV2 yaitu 1.0000 dan 0.9561.
3. Nilai loss training dan loss validation pada epoch terakhir arsitektur MobileNet yaitu 0.000804 dan 0.1584, sedangkan pada arsitektur MobileNetV2 yaitu 0.0070 dan 0.1372.
4. Persentase akurasi dari confusion matrix dengan menggunakan dataset test pada model arsitektur MobileNet adalah 93.999%, sama halnya dengan model arsitektur MobileNetv2 yang juga mendapatkan nilai akurasi 93.999%.
5. Persentase loss dari confusion matrix dengan menggunakan dataset test pada model arsitektur MobileNet adalah 0.1319%, sama halnya dengan model arsitektur MobileNetv2 yang juga mendapatkan nilai akurasi 0.1319%.

6. Perancangan aplikasi mobile dilakukan pada infrastruktur Visual Studio Code dengan menggunakan framework UI Flutter dan menggunakan bahasa pemrograman dart.
7. Pengujian aplikasi berjalan dengan sempurna dengan persentase akurasi dari pengujian aplikasi mencapai angka 100%, sedangkan presentase loss dari pengujian aplikasi mencapai angka 0%.

5.2 Saran

Saran yang perlu dikembangkan untuk tugas akhir ini agar lebih baik sebagai berikut.

1. Perlu adanya penambahan jumlah dataset train agar hasil prediksi dari model meningkat
2. Perlu adanya variasi dataset seperti penambahan mata dari ras eropa, latin, dan lain lain, hal ini bertujuan agar model lebih bisa membaca lebih baik dan meningkatkan nilai akurasi
3. Perlu adanya penambahan kelas penyakit mata, seperti glaucoma, diabetic renopathy, degenerasi macula, dan lain lain. Hal ini bertujuan agar model dan aplikasi dapat digunakan dengan lebih universal dan tidak terpaku pada penyakit mata katarak saja.
4. Arsitektur transfer learning pada model pendeteksian mata katarak ini dapat diganti menggunakan arsitektur transfer learning lainnya seperti, VGG-16, ResNet50, atau InceptionV3
5. Modul deteksi mata katarak yang digunakan pada pengujian ini dapat diganti menggunakan modul lain seperti YOLO, MTCNN, dan lain lain.