

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil pelatihan dan pengujian model, dapat disimpulkan bahwa arsitektur CNN dengan menggunakan bantuan model dasar *Xception* mencapai kinerja yang sangat baik. Pada set pelatihan, model mencapai akurasi tertinggi yaitu 0.9677 atau **96.77%** dan loss 0.2829 atau sebesar **28.29%**, sedangkan pada set pengujian, model mencapai akurasi sebesar **95.24%**, dan mendapati nilai loss 0.3048 atau **30.48%**. Hal ini menunjukkan bahwa model mampu dengan baik untuk memahami dan memetakan hubungan kompleks dalam dataset citra abjad tunggal BISINDO.

Dengan melihat metrik evaluasi, dapat diamati bahwa model berhasil mengklasifikasikan sebagian besar kelas dengan tingkat presisi, *recall*, dan *f1-score* yang tinggi. Namun, perlu diperhatikan bahwa terdapat beberapa kelas seperti ‘O’, ‘Q’, dan ‘V’ yang memiliki performa rendah. Penyebab potensial dapat diidentifikasi melalui analisis lebih lanjut, dan pembaruan pada data atau arsitektur model mungkin diperlukan.

Beberapa faktor utama yang dapat mempengaruhi akurasi klasifikasi mencakup keterbatasan dataset, jumlah lapisan dalam arsitektur model CNN, dan penggunaan model pra-latih. Ketersediaan data yang cukup untuk setiap huruf sangat penting untuk melatih model yang akurat. Kurangnya data untuk huruf tertentu dapat menyebabkan performa model menjadi tidak konsisten. Penggunaan model pra-latih juga memainkan peran penting dalam meningkatkan akurasi klasifikasi karena model pra-latih telah dilatih pada dataset besar dan dapat mengekstrak fitur yang lebih bermakna dari data baru. Integrasi model pra-latih dapat mempercepat proses pelatihan dan meningkatkan performa, terutama ketika dataset yang tersedia terbatas. Dengan mempertimbangkan faktor-faktor ini, pengembangan model CNN yang efektif untuk pengenalan citra BISINDO dapat dioptimalkan.

5.2 Saran

Penulis telah merangkum beberapa saran untuk perbaikan dan pengembangan lebih lanjut seperti berikut:

1. Lakukan analisis lebih lanjut terhadap kelas-kelas dengan performa rendah seperti 'C', 'D', 'F', 'H', 'N', 'P', 'S', 'V' dan 'Y'. Mungkin diperlukan perbaikan pada representasi data atau penyesuaian pada arsitektur model.
2. Pertimbangkan untuk menggunakan augmentasi data tambahan untuk kelas-kelas yang kurang mewakili. Hal ini dapat membantu meningkatkan keragaman data dan kinerja model.
3. Lakukan eksperimen lebih lanjut dengan hyperparameter untuk mencari kombinasi yang optimal. Terutama, perhatikan learning rate, jumlah lapisan, dan dropout rates.
4. Melakukan penelitian lebih lanjut dengan variasi arsitektur model, seperti menggunakan model lain atau menambahkan lapisan-lapisan spesifik untuk meningkatkan performa.