

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil eksperimen dan analisis yang telah dilakukan dalam membandingkan arsitektur MobileNetV2 dan VGG16 untuk klasifikasi alfabet BISINDO pada kondisi dataset terbatas, dapat ditarik kesimpulan sebagai berikut:

- Implementasi metode *Transfer Learning* menggunakan arsitektur MobileNetV2 terbukti sangat efektif dalam mengklasifikasikan alfabet BISINDO meskipun dalam kondisi keterbatasan data. Hal ini ditunjukkan pada skenario *Baseline*, di mana model mampu mengenali fitur alfabet dengan sangat baik dan mencapai tingkat akurasi pengujian sebesar 94% secara cepat. Keberhasilan ini mengonfirmasi bahwa bobot *pre-trained* dari ImageNet memberikan dasar ekstraksi fitur yang kuat dalam pengenalan alfabet BISINDO. Namun, skenario ini memiliki kekurangan karena kurangnya variasi data pada dataset yang terbatas, sehingga model memiliki risiko sekadar menghafal (*memorization*) pola data latih yang statis dan seragam. Kondisi tersebut membuat model pada skenario ini kurang tangguh (*robust*) jika dihadapkan pada variasi kondisi dunia nyata.
- Penerapan teknik Augmentasi Data terbukti secara signifikan meningkatkan ketangguhan (*robustness*) model dan kemampuan generalisasi dibandingkan penggunaan data mentah. Skenario *Proposed* membuktikan performa yang jauh lebih tangguh saat dihadapkan pada data yang divariasikan, berhasil mencapai titik konvergensi terbaik dengan *validation loss* terendah sebesar 0.3075. Meskipun secara numerik akurasi sedikit menurun menjadi 92,31%, teknik ini berhasil memperkaya representasi visual dan memaksa model mempelajari fitur invarian gestur tangan. Hal ini efektif mengatasi risiko model sekadar menghafal pola data latih, sehingga model tetap akurat saat menghadapi variasi orientasi dan posisi objek yang kompleks di dunia nyata. Sementara itu,

mekanisme *Early Stopping* berfungsi secara presisi untuk menghentikan pelatihan pada titik optimal di *epoch* ke-186, sehingga mencegah pemborosan waktu komputasi serta menghindari degradasi akurasi akibat *overtraining*.

- c. Arsitektur MobileNetV2 mampu menghasilkan keseimbangan yang sangat optimal antara akurasi tinggi dan efisiensi waktu komputasi. Melalui skenario *Benchmark* sebagai pembandingan dengan konfigurasi eksperimen yang identik, MobileNetV2 terbukti jauh lebih unggul dibandingkan arsitektur VGG16, yang dikenal sebagai arsitektur deep dengan kompleksitas dan jumlah parameter yang besar. MobileNetV2 mencapai akurasi 92,31% dengan waktu pelatihan hanya ~35,01 menit, sedangkan VGG16 yang memiliki parameter lebih masif justru gagal konvergen secara stabil dengan akurasi tertahan di angka 81% dan membutuhkan waktu pelatihan jauh lebih lama, yaitu ~82,77 menit. Hasil ini memvalidasi bahwa penggunaan arsitektur *lightweight* merupakan solusi yang lebih tepat untuk sistem pengenalan isyarat yang responsif, terutama pada kondisi keterbatasan dataset dan sumber daya komputasi.

5.2 Saran

Berdasarkan kendala yang dihadapi selama proses penelitian serta hasil evaluasi yang diperoleh, berikut adalah beberapa saran untuk pengembangan penelitian selanjutnya:

- a. Perluasan dataset dan variasi lingkungan sangat disarankan karena penelitian ini masih terbatas pada dataset statis dengan jumlah sampel yang minim. Peneliti selanjutnya perlu menambah jumlah sampel data dan memperbanyak variasi kondisi pengambilan gambar, seperti tingkat pencahayaan yang berbeda serta penggunaan latar belakang yang kompleks (*complex background*), untuk menguji ketangguhan (*robustness*) model dalam skenario dunia nyata.

- b. Implementasi pada perangkat bergerak (*mobile*) menjadi langkah pengembangan yang strategis mengingat efisiensi arsitektur MobileNetV2 yang terbukti ringan dan cepat dibandingkan arsitektur pembanding. Fokus penelitian berikutnya dapat diarahkan pada tahap *deployment* model ke dalam aplikasi berbasis Android atau iOS secara *real-time* guna memberikan dampak praktis yang lebih besar bagi komunitas Tuli dalam berkomunikasi sehari-hari.
- c. Pengembangan ke objek dinamis berupa kata atau kalimat merupakan langkah lanjut yang penting karena penelitian ini hanya berfokus pada klasifikasi alfabet statis. Peneliti dapat mengombinasikan algoritma CNN sebagai pengekstraksi fitur dengan arsitektur seperti LSTM (*Long Short-Term Memory*) atau GRU (*Gated Recurrent Unit*) untuk menangkap ketergantungan urutan waktu (*temporal dependencies*) pada gerakan isyarat yang lebih kompleks.

