

BAB I

PENDAHULUAN

1.1 Latar Belakang

Cuaca merupakan kondisi pada atmosfer yang terjadi karena berbagai faktor, antara lain radiasi matahari, suhu udara, kecepatan dan arah angin. Jenis-jenis cuaca meliputi cuaca cerah, berawan, panas, dingin, berangin, dan hujan [1]. Cuaca sangat memengaruhi aktivitas kehidupan manusia. Klasifikasi unsur-unsur cuaca dapat berfungsi sebagai pedoman untuk keadaan cuaca di suatu lokasi. Klasifikasi unsur-unsur cuaca sering digunakan dalam peramalan cuaca dan pedoman *deep learning* dalam proses pengambilan keputusan. Sistem komputer memiliki kemampuan untuk mengamati kondisi cuaca dan menilainya berdasarkan kondisi cuaca yang sedang terjadi. Oleh karena itu, untuk memberikan hasil yang lebih akurat dan mengurangi kesalahan pengamat cuaca, sistem harus mengklasifikasikan cuaca menurut kategorinya.

Perubahan cuaca bergantung pada banyak faktor seperti suhu, kelembaban, angin, waktu, lokasi dan lainnya. Dari faktor ini dapat diperoleh jenis cuaca seperti cuaca cerah, berawan, hujan, hujan petir. Cuaca ekstrim berpotensi menjadi bencana seperti banjir, tanah longsor, kebakaran, penyebaran penyakit yang mempengaruhi daya tahan tubuh manusia. Dengan teknik klasifikasi cuaca yang baik dapat diprediksi kemungkinan perubahan cuaca yang terjadi dengan lebih akurat [2].

Pada tahun 2020, penelitian klasifikasi curah hujan dilakukan menggunakan metode *Convolutional Neural Network (CNN)* dengan arsitektur VGG16Net. Penelitian ini menggunakan 4 skenario pengujian. Hasil dari performa sistem yang diperoleh pengklasifikasi cuaca tidak mengalami *overfitting* maupun *overshooting* atau sistem mampu mengenali kondisi dari data citra latih dan data citra validasi dengan akurasi sebesar 87% [1]. Pada tahun yang sama di tahun 2020, penelitian klasifikasi untuk prediksi cuaca menggunakan metode *Esemble Learning*. Penelitian ini menggunakan dataset klasifikasi cuaca di Australia yang terdiri dari 142.193 *record* dan 24 atribut, dengan data selama 10 tahun, hasil akurasi dan MSE dari metode *esemble learning* adalah 81.21% sebagai akurasi dan 18.79% untuk

MSE [3]. Pada tahun 2022, penelitian klasifikasi cuaca provinsi DKI JAKARTA menggunakan algoritma *Random Forest* dengan teknik *oversampling*, eksperimen diperoleh akurasi model *Random Forest* mencapai 70% dengan teknik SMOTE. Hasil klasifikasi juga mampu memperbaiki prediksi setiap kelas minoritas 1, 2, 5, 6 dan 7 dengan rata-rata kenaikan prediksi benar mencapai 50% [2].

Pada penelitian ini terdapat perbedaan yaitu menggunakan Arsitektur *MobileNetV2*, serta Kaggle sebagai dataset yang akan digunakan. Salah satu metode yang populer saat ini adalah *MobileNet*. *MobileNet* memiliki kapabilitas yang terbaik untuk mengenali objek dengan akurasi yang tinggi dan kecepatan waktu deteksi [4].

1.2 Rumusan Masalah

Berdasarkan deskripsi latar belakang dan penelitian terkait, maka dapat dirumuskan beberapa masalah dalam Tugas Akhir ini yaitu:

1. Bagaimana merancang sebuah system untuk mengklasifikasikan cuaca menggunakan *CNN* dengan arsitektur *MOBILENETV2*
2. Bagaimana mengukur parameter kinerja dari system klasifikasi cuaca yang dirancang menggunakan *CNN* dengan arsitektur *MOBILENETV2*
3. Menganalisis performansi dari metode *CNN* dengan arsitektur *MOBILENETV2* untuk klasifikasi cuaca

1.3 Tujuan Manfaat

Adapun tujuan dari penulisan Tugas Akhir ini adalah.

1. Merancang dan mengimplementasikan sistem yang dapat mendeteksi cuaca menggunakan metode *CNN* dengan arsitektur *MOBILENETV2*.
2. Mengetahui parameter yang mempengaruhi hasil dari performansi sistem pengukuran klasifikasi cuaca menggunakan metode *CNN* dengan arsitektur *MOBILENETV2*.
3. Menganalisis performansi metode *CNN* dengan arsitektur *MOBILENETV2*.

Manfaat dari penelitian ini yaitu dapat mendeteksi cuaca secara efektif dan

efisien sehingga dapat dilakukan klasifikasi cuaca lebih lanjut dengan tepat dan cepat.

1.4 Batasan Masalah

Pada Tugas Akhir ini terdapat beberapa batasan masalah antara lain:

1. Penelitian ini menggunakan dataset sekunder yang dapat diakses secara online di Kaggle, dengan total citra cuaca 768.
2. Penelitian ini mengklasifikasi cuaca pada tiga kelas yaitu cerah, berawan dan hujan sehingga citra cuaca yang digunakan hanya yang berkaitan dengan cuaca tersebut, dengan persentase 80% data pelatihan model dan 20% data pengujian.
3. Penelitian ini menggunakan metode *Convolutional Neural Network (CNN)*
4. Penelitian ini menggunakan arsitektur *MOBILENETV2*.
5. Penelitian ini membahas mengenai akurasi pembelajaran model pada citra cuaca pada kelas yang sudah ditentukan.
6. Format File citra yang digunakan adalah *.jpg.

1.5 Metode Penelitian

Dalam mengerjakan Tugas Akhir ini digunakan metode eksperimental dengan tahapan-tahapan sebagai berikut:

1. Studi Literatur

Pada tahap ini dilakukan pembelajaran, pengumpulan materi serta informasi mengenai hal terkait melalui referensi yang menunjang pada penelitian, seperti jurnal, artikel, ebook mengenai cuaca dan *CNN*.

2. Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data digunakan sekunder dan dapat diperoleh dari Kaggle yang sudah tersedia secara online.

3. Perancangan Sistem

Pada tahap ini dilakukan perancangan dan analisa kebutuhan dalam penelitian terkait, serta mengetahui parameter apa saja yang digunakan untuk

klasifikasi pada cuaca. Kemudian dilakukan perancangan program dan menganalisis hasil perancangan sesuai dengan metode yang digunakan.

4. Implementasi Sistem

Pada tahap ini dilakukan simulasi dan implementasi metode *CNN* dengan arsitektur *MOBILENETV2* pada sistem ke dalam program (coding).

5. Pengujian dan Analisis

Hasil Pada tahap ini dilakukan validasi coba dan analisis kinerja dari implementasi metode *CNN* dengan arsitektur *MOBILENETV2* pada sistem untuk klasifikasi cuaca yang telah diimplementasikan untuk mengetahui performansi sistem.

6. Dokumentasi dan Penyusunan Laporan

Pada tahap ini dilakukan dokumentasi dan penyusunan laporan akhir serta pengambilan kesimpulan penelitian yang telah dilakukan terhadap penerapan *CNN* dengan arsitektur *MOBILENETV2* pada sistem untuk klasifikasi cuaca.