

BAB I PENDAHULUAN

1.1 Latar Belakang

Rambu lalu lintas memegang peranan vital dalam instrumen keselamatan berkendara, sebab mereka memberikan informasi terkait kondisi jalan yang akan dilalui, kondisi lingkungan sekitar, serta menjadi acuan bagi pengguna jalan lainnya dalam membuat keputusan selama perjalanan [1], [2]. Dengan kemajuan teknologi, rambu lalu lintas mempunyai peran krusial, seperti pada pengembangan sistem *Advanced Driver Assistance Systems (ADAS)* [3], [4]. Sistem ini umumnya diterapkan pada mobil biasa dan *self-driving car*, yaitu teknologi yang memungkinkan mobil untuk berkendara tanpa pengemudi manusia [5]. Sistem-sistem ini tentunya sangat bergantung pada kemampuan mengenali dan mengartikan rambu lalu lintas pada berbagai tantangan termasuk keragaman desain rambu, letak geografis, dan keterbatasan daya komputasi [6], [7]. Maka dari itu, diperlukan adanya model yang dapat melakukan deteksi rambu lalu lintas pada sistem dengan daya komputasi terbatas namun tetap memiliki performa akurasi dan kecepatan yang baik. Menjawab kebutuhan untuk permasalahan ini, model YOLOv5 dengan penerapan metode *knowledge distillation* akan diimplementasikan dalam penelitian ini.

Penggunaan model objek deteksi dari keluarga You Only Look Once (YOLO) telah banyak digunakan dalam penelitian terkait, khususnya untuk *traffic recognition* dan *traffic sign detection* karena reputasinya dalam tingkat akurasi dan kecepatan deteksi [5], [8], [9]. Selain karena kecepatan dan akurasinya, model-model YOLO juga dianggap "*user-friendly*" karena dokumentasinya terstruktur dengan rapi, serta dukungan komunitas yang masih aktif. Dua faktor ini menciptakan kemudahan bagi siapapun yang hendak mempelajari, mendalami, atau mengembangkan model objek deteksi.

Dalam konteks objek deteksi pada sistem dengan daya komputasi terbatas, kebutuhan akan model yang ringan dan akurat menjadi dasar dan tujuan banyak

penelitian menggunakan model YOLO dengan berbagai pendekatan [10], [11]. Dengan demikian, dapat dikatakan penelitian-penelitian sebelumnya berfokus pada peningkatan performa pada tiap eksperimennya, sehingga tantangan dalam efisiensi komputasi untuk kondisi *real-time* masih terus membuka peluang eksplorasi.

Metode kompresi model merupakan pendekatan yang beberapa tahun terakhir aktif diteliti. Umumnya, metode ini bekerja dengan cara mengurangi kompleksitas model dan ukuran model bertujuan agar model menjadi lebih ringan secara komputasi. Salah satu metode kompresi yang umum digunakan adalah teknik *knowledge distillation* (KD) [12]. KD secara mudah dapat diartikan sebagai teknik melakukan distilasi atau transfer pengetahuan dari model jaringan besar (*teacher*) ke model jaringan yang lebih kecil (*student*), cara ini memungkinkan model yang lebih kecil untuk belajar dengan panduan dari model besar. Sehingga model yang lebih kecil mampu mencapai performa yang lebih baik berkat bantuan dari model yang lebih besar. “*Knowledge*” atau pengetahuan yang menjadi acuan oleh model student memiliki beberapa jenis, dan tergantung pada jenis dasar pengetahuannya, teknik-teknik KD juga memiliki pendekatan berbeda [12], [13].

Sebagai metode kompresi, penerapan teknik *knowledge distillation* pada model YOLOv5 memungkinkan pelatihan yang lebih baik untuk model kecil sehingga berdampak pada peningkatan kecepatan deteksi dan tingkat akurasi. Dengan demikian, pendekatan ini menawarkan solusi untuk kebutuhan deteksi objek pada lingkungan dengan daya komputasi terbatas. Hasil penelitian ini diharapkan dapat berkontribusi pada literatur terkait optimasi model *deep learning* untuk diaplikasikan pada perangkat dengan kapasitas komputasi terbatas.

1.2 Rumusan Masalah

Berdasarkan penjelasan pada latar belakang, berikut merupakan rumusan masalah pada penelitian ini:

1. Bagaimana efektivitas penerapan teknik *knowledge distillation* pada YOLOv5 untuk mendeteksi rambu lalu lintas?
2. Bagaimana perbandingan performa model YOLOv5 yang didistilasi dengan

varian model setara tanpa distilasi?

1.3 Batasan Masalah

Penelitian ini berada pada batasan ruang lingkup sebagai berikut:

1. Jenis rambu lalu lintas yang dideteksi merupakan rambu yang spesifik digunakan di Indonesia. Tidak termasuk rambu-rambu lain yang sifatnya lokal dan non-standar.
2. Teknik kompresi yang digunakan terbatas pada *knowledge distillation*, teknik lain seperti *pruning* dan *quantization* tidak dibahas dalam penelitian ini.
3. Model deteksi rambu lalu lintas penelitian ini dilatih dengan asumsi kondisi cuaca dan pencahayaan yang terang atau jelas. Tidak mencakup kondisi kompleks atau ekstrem seperti kabut, hujan lebat, atau malam hari.
4. Evaluasi kinerja model menggunakan metrik-metrik umum seperti *mean Average Precision*, *Precision*, dan *Recall*. Serta tambahan perhitungan konsumsi memori dan waktu inferensi, tidak termasuk penghitungan *Frame Per Second (FPS)* dan metrik evaluasi lain.
5. Implementasi dan pengujian deteksi dilakukan pada lingkungan Google Colab, tidak termasuk pengujian pada kendaraan di jalan raya.

1.4 Tujuan Penelitian

Tujuan yang ingin dicapai dengan penelitian ini adalah sebagai berikut:

1. Menerapkan teknik *knowledge distillation* pada model YOLOv5 untuk deteksi rambu lalu lintas.
2. Mengetahui kinerja model YOLOv5 dengan penerapan teknik *knowledge distillation*.

1.5 Manfaat Penelitian

Penelitian yang dilakukan diharapkan mampu memberikan manfaat sebagai berikut:

1. Dapat memberikan kontribusi dalam pengembangan lebih lanjut dibidang *computer vision*, khususnya pada penerapan metode kompresi model objek deteksi.
2. Memberikan referensi dan wawasan tentang penerapan teknik *knowledge distillation* pada model YOLOv5.

1.6 Sistematika Penulisan

BAB I PENDAHULUAN, bab ini menguraikan latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA, bab ini membahas literatur yang menjadi tinjauan dan dasar-dasar teori yang digunakan penelitian ini. Seperti deteksi objek, YOLOv5, CNN, *model compression*, dan *knowledge distillation*.

BAB III METODE PENELITIAN, bab ini membahas objek penelitian, alur penelitian yang mencakup pengumpulan data, *preprocessing* data, augmentasi data, *training* dan evaluasi model, dan pengujian. Terdapat pula penjelasan terkait *mockup* antarmuka aplikasi web.

BAB IV HASIL DAN PEMBAHASAN, bab ini berisi penjelasan terkait *training* model, analisis dan evaluasi hasil *training* model, pengujian model yang telah dilatih, dan uji coba deteksi menggunakan aplikasi web.

BAB V PENUTUP, bab terakhir berisi kesimpulan yang didapatkan setelah penelitian, dan saran untuk penelitian berikutnya.