BAB 1
PENDAHULUAN

1.1 Latar Belakang Masalah

Berkembang pesatnya bidang *artificial intelligence* telah mendorong penerapan edge AI secara masif, di mana model machine learning dijalankan langsung pada perangkat pengguna, bukan pada server cloud terpusat. Tren ini dipicu oleh semakin maraknya penggunaan perangkat mobile, sistem Internet of Things (IoT), serta infrastruktur edge computing yang menghasilkan dan memproses data langsung pada perangkat [1]. Akibatnya, semakin banyak model AI yang diimplementasikan secara on-device untuk mendukung aplikasi real-time pada perangkat dengan keterbatasan latensi dan privasi [2].

Peralihan menuju edge AI didorong oleh beberapa faktor. Menjalankan inferensi machine learning secara on-device dapat mengurangi latensi inferensi secara signifikan, sehingga memungkinkan aplikasi real-time dapat berjalan lebih responsif dan meningkatkan pengalaman pengguna. Selain itu, inferensi on-device dapat meningkatkan privasi dan keamanan data karena data sensitif pengguna tidak perlu dikirim ke server cloud [3]. Studi sebelumnya menunjukkan bahwa inferensi berbasis cloud dapat menimbulkan latensi signifikan, biaya bandwidth yang tinggi, serta risiko keamanan terhadap data privat seperti rekam medis dan data biometrik selama proses transmisi dan penyimpanan [4]. Edge AI juga menawarkan keuntungan dari sisi efisiensi energi sistem secara keseluruhan dengan menghindari konsumsi daya tinggi pada pusat data cloud. Meskipun demikian, eksekusi algoritma machine learning yang intensif komputasi pada perangkat mobile tetap menghadapi tantangan signifikan akibat keterbatasan kapasitas baterai [5].

Untuk menjalankan AI pada perangkat, platform mobile umumnya menyediakan framework untuk menyederhanakan eksekusi AI dengan cara mengotomatisasi manajemen hardware. Framework tersebut memudahkan proses pengembangan karena detail perangkat keras tingkat rendah disembunyikan dari pengembang, sehingga pengembang tidak perlu bersinggungan langsung dengan perangkat keras. Namun mekanisme internal framework tersebut sering kali spesifik terhadap platform tertentu. Akibatnya, model machine learning yang identik dapat menunjukkan karakteristik runtime yang berbeda secara signifikan tergantung pada platform perangkat keras, maupun framework yang digunakan [6, 7].

Sebagai salah satu platform mobile yang paling banyak digunakan, Apple menyediakan Core ML sebagai framework native untuk menjalankan machine learning secara on-device. Core ML dirancang untuk mengoptimalkan inferensi dengan memanfaatkan arsitektur perangkat keras pada perangkat iOS, termasuk CPU, GPU, dan Apple Neural Engine (ANE). Core ML memungkinkan pengembang untuk mengimplementasikan model machine learning secara efisien tanpa harus mengelola perangkat keras secara langsung. Namun, framework ini tidak menyediakan dokumentasi rinci mengenai bagaimana beban kerja inferensi dijadwalkan dan didistribusikan ke unit komputasi yang tersedia [8], sehingga pengembang harus mengandalkan pengukuran empiris menggunakan alat profiling untuk mengamati perilaku runtime. Akibatnya, latensi sulit diprediksi secara analitis [9]. Pola konsumsi energi dan perilaku termal juga hanya dapat diamati melalui pengukuran empiris pada perangkat nyata, khususnya pada eksekusi berdurasi panjang di mana efek termal dan energi berkembang seiring waktu [10, 11].

Beberapa studi sebelumnya telah mengukur kinerja inferensi machine learning pada ekosistem Apple, namun umumnya terbatas pada evaluasi kinerja sesaat dan belum mencakup perilaku runtime pada beban kerja berkelanjutan. Bazso dan Ahremark telah melakukan studi benchmark terhadap konversi model PyTorch ke Core ML dan menunjukkan bahwa Core ML mampu memanfaatkan ANE untuk mempercepat inferensi secara signifikan dibandingkan eksekusi pada CPU [12]. Namun, studi mereka membatasi pengukuran pada latensi rata-rata dari 500 iterasi yang dijalankan secara berurutan pada kondisi termal normal, sehingga karakteristik runtime yang muncul pada beban kerja berkelanjutan seperti stabilitas latensi, konsumsi energi, dan perilaku termal belum dikaji. Studi tersebut secara eksplisit mengidentifikasi pengukuran konsumsi daya dan eksplorasi pemanfaatan ANE pada lingkungan empiris yang lebih komprehensif sebagai arah penelitian lanjutan [12]. Demikian pula penelitian Jacques et al. yang melakukan studi komparatif terhadap framework machine learning pada perangkat iOS dan menunjukkan bahwa pemilihan framework dapat berpengaruh signifikan terhadap penggunaan baterai [13]. Meskipun relevan, penelitian tersebut menitikberatkan pada perbandingan antar framework dan tidak menginvestigasi dampak konfigurasi eksekusi perangkat keras yang berbeda di dalam Core ML terhadap karakteristik runtime.

Padahal, dalam skenario edge AI, inferensi umumnya dijalankan secara berkelanjutan pada perangkat dengan *resource* terbatas seperti pada *voice assistant*

dan *computer vision* [14]. Pada aplikasi *computer vision*, *Convolutional Neural Network* (CNN) merupakan arsitektur yang dominan digunakan, dengan MobileNetV2 sebagai salah satu arsitektur CNN yang banyak diterapkan pada perangkat mobile karena dirancang khusus untuk efisiensi komputasi [15, 16]. Dalam kondisi ini, karakteristik runtime seperti latensi yang stabil, konsumsi energi, dan perilaku termal secara langsung memengaruhi keberlanjutan sistem, umur baterai, serta pengalaman pengguna [10, 17]. Pada kondisi demikian, setiap konfigurasi compute unit Core ML berpotensi menghasilkan profil latensi, konsumsi energi, dan perilaku termal yang berbeda [17, 18, 19]. Perbedaan tersebut tidak dapat diprediksi secara analitis dan hanya dapat diidentifikasi melalui pengukuran empiris langsung pada perangkat [8, 9].

Berdasarkan kondisi tersebut, penelitian ini berfokus pada analisis empiris terhadap karakteristik runtime inferensi model *Convolutional Neural Network* (CNN) secara on-device menggunakan Core ML. Penelitian ini mengkaji bagaimana perbedaan konfigurasi eksekusi perangkat keras, yang diatur melalui parameter `MLComputeUnits` dalam Core ML, memengaruhi latensi inferensi, stabilitas, perilaku termal, dan konsumsi energi selama operasi berkelanjutan. Parameter ini memungkinkan pembatasan penggunaan unit komputasi tertentu, sehingga CPU, GPU, dan Apple Neural Engine dapat dievaluasi secara terpisah dalam kondisi yang terkontrol. Dengan mengalihkan fokus dari pengukuran latensi rata-rata pada kondisi sesaat menjadi perilaku runtime selama eksekusi berkelanjutan, penelitian ini secara khusus dapat memberikan wawasan praktis yang mendukung deployment model machine learning yang lebih efisien dan terinformasi pada platform mobile modern.

1.2 Rumusan Masalah

1. Bagaimana menganalisis karakteristik runtime inferensi model Convolutional Neural Network secara on-device menggunakan Core ML?
2. Bagaimana perbedaan konfigurasi compute unit (CPU, GPU, dan ANE) memengaruhi kinerja runtime berdasarkan metrik latensi, stabilitas kinerja, konsumsi energi, dan perilaku termal?

1.3 Batasan Masalah

1. Penelitian hanya berfokus pada tiga konfigurasi eksekusi Core ML, yaitu `.cpuOnly` (CPU), `.cpuAndGPU` (CPU dan GPU), dan `.all` (CPU, GPU, dan Apple Neural Engine), tanpa mempertimbangkan konfigurasi lain `MLComputeUnits`.
2. Penelitian menggunakan model MobileNetV2 dengan format *specification version* 1 (NeuralNetwork).
3. Penelitian hanya mencakup fase inferensi (*inference*) dan tidak membahas tahap pelatihan (*training*) maupun konversi model dari *framework* eksternal.
4. Penelitian hanya dilakukan pada satu perangkat iOS, sehingga hasil tidak digeneralisasi ke perangkat, chip, atau versi sistem operasi lain.
5. Penelitian hanya mengevaluasi karakteristik runtime berupa latensi, stabilitas kinerja, konsumsi energi, dan perilaku termal, tanpa melibatkan metrik akurasi model karena seluruh konfigurasi menjalankan model yang identik.
6. Perilaku termal hanya diukur sebagai status termal diskret (`nominal`, `fair`, `serious`, `critical`) melalui `ProcessInfo.thermalState`, bukan sebagai suhu fisik dalam satuan derajat, karena iOS tidak mengekspos sensor suhu kepada aplikasi.

1.4 Tujuan Penelitian

1. Menganalisis karakteristik runtime inferensi model Convolutional Neural Network secara on-device menggunakan Core ML.
2. Mengkaji bagaimana perbedaan konfigurasi compute unit (CPU, GPU, dan ANE) memengaruhi kinerja runtime berdasarkan metrik latensi, stabilitas kinerja, konsumsi energi, dan perilaku termal.

1.5 Manfaat Penelitian

1. Memberikan analisis empiris mengenai dampak konfigurasi eksekusi Core ML terhadap kinerja inferensi machine learning secara on-device, khususnya dalam aspek latensi, stabilitas kinerja, konsumsi energi, dan perilaku termal.

2. Menjadi referensi bagi pengembang dan peneliti dalam memahami karakteristik runtime Core ML pada perangkat iOS sebagai dasar untuk pengambilan keputusan yang lebih terinformasi dan efisien dalam deployment machine learning secara on-device.

1.6 Sistematika Penulisan

Sistematika penulisan laporan adalah sebagai berikut:

- Bab 1 PENDAHULUAN
  Bab ini berisi uraian mengenai latar belakang penelitian, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta sistematika penulisan yang digunakan dalam penyusunan skripsi ini.

- Bab 2 LANDASAN TEORI
  Bab ini membahas teori-teori yang mendasari penelitian, meliputi konsep Edge AI dan on-device machine learning, inferensi machine learning, *Convolutional Neural Network* (CNN) beserta arsitektur MobileNetV2, arsitektur perangkat mobile (CPU, GPU, dan Apple Neural Engine), serta framework Core ML. Selain itu, dibahas pula karakteristik runtime seperti latensi, stabilitas, konsumsi energi, dan perilaku termal yang menjadi dasar analisis dalam penelitian ini.

- Bab 3 METODOLOGI PENELITIAN
  Bab ini menjelaskan tahapan penelitian secara sistematis, meliputi perancangan eksperimen, pemilihan dan persiapan model machine learning, konfigurasi compute unit pada Core ML, skenario pengujian, serta metode pengukuran dan pengumpulan data terkait karakteristik runtime.

- Bab 4 HASIL DAN DISKUSI
  Bab ini menyajikan hasil eksperimen yang telah dilakukan, berupa analisis performa runtime berdasarkan berbagai konfigurasi compute unit. Pembahasan difokuskan pada perbandingan latensi, stabilitas kinerja, konsumsi energi, serta perilaku termal selama proses inferensi berlangsung.

- Bab 5 SIMPULAN DAN SARAN
  Bab ini memuat kesimpulan dari hasil penelitian yang telah dilakukan berdasarkan rumusan masalah yang diajukan, serta memberikan saran untuk pengembangan penelitian selanjutnya.