

BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Penyusup adalah orang yang masuk secara diam-diam ke dalam daerah yang tidak seharusnya dimasuki [4]. Tindakan penyusupan ini sendiri seringkali dilakukan guna mengambil barang kepemilikan korban yang berada di dalam rumah korban tanpa sepengetahuan korban pada saat rumah sedang kosong (pencurian). Menurut Badan Pusat Statistik, kejadian kejahatan yang paling banyak dialami pada tahun 2014, 2018, dan 2021 adalah kejahatan pencurian dengan presentase berturut-turut 41.05%, 45.01%, dan 26.50% [5]. Untuk mengatasi tindakan pencurian, memperkerjakan satpam seringkali digunakan sebagai jalan keluar. Namun penggunaan satpam tidak sepenuhnya efektif karena penjagaan satpam tidak dapat mencakup seluruh daerah yang rawan disusupi selama 24 jam. Penggunaan CCTV juga seringkali digunakan sebagai jalan keluar. Namun hingga sekarang, penggunaan CCTV seringkali dirasa kurang efektif karena *visual* CCTV harus terus dipantau oleh seseorang tanpa ada tanda bahwa sedang terjadi pencurian atau perampokan. Oleh karena itu, diperlukan pembelajaran mesin yang dapat mendeteksi adanya penyusup secara *live* dengan menggunakan *deep learning*.

Deep learning adalah cabang *machine learning* yang memungkinkan komputer agar dapat belajar secara mandiri dan memahami dunia dalam hal konsep hierarki [6]. *Deep learning* dapat digunakan untuk mengidentifikasi suatu objek berdasarkan citra dengan cara mempelajari representasi data objek secara otomatis. Salah satu metode pengidentifikasian yang dapat digunakan dalam *deep learning* adalah metode *Convolutional Neural Network* atau biasa disingkat CNN. CNN adalah jenis jaringan saraf khusus untuk memproses data yang memiliki topologi grid [6]. CNN memiliki proses kovolusi dengan tujuan memperoleh fitur yang terdapat pada citra yang diperlukan untuk mengidentifikasi citra tersebut.

You Only Look Once atau bisa disingkat YOLO adalah sebuah *framework machine learning* yang menggunakan algoritmanya sendiri untuk melakukan pengidentifikasian objek secara *real-time*. Algoritma pada *framework* YOLO sendiri merupakan metode CNN yang *layer-layer*-nya telah di-*predefined* terlebih dahulu sehingga YOLO dapat melakukan pengidentifikasian objek secara sangat cepat [7]. Meskipun dengan pengidentifikasian objeknya dapat dilakukan secara

cepat, YOLO memiliki tingkat akurasi lebih tinggi hingga 2x dari rata-rata akurasi dari sistem *real-time* lainnya [7]. Sehingga, hingga sekarang YOLO masih terus dikembangkan dari yang mulanya berawal dari YOLO v1 hingga sekarang YOLO v7 karena kecepatannya yang sangat cepat tanpa mengurangi tingkat keakuratan pengidentifikasian objeknya [8].

Berdasarkan kajian di atas, dengan keunggulannya yang dapat melakukan pendeteksian dan pengidentifikasian objek secara cepat, YOLO dapat dimanfaatkan untuk melakukan pendeteksian aplikasi *real-time* seperti aplikasi pengawasan dan mobil dengan fitur *self-driving* dimana kecepatan pengidentifikasian objek sangat krusial dan penting. YOLO sendiri pernah digunakan pada penelitian pembuatan aplikasi pendeteksian manusia secara *real-time* melalui kamera pengawasan. Pada penelitian tersebut didapati hasil bahwa tingkat akurasinya mencapai 95.05% dan 96.81% [9]. Dengan adanya penelitian tersebut, maka memungkinkan pengembangan lebih lanjut pemanfaatan YOLO yang dapat digunakan dalam implementasi sistem keamanan rumah untuk mendeteksi adanya intrusi ke dalam rumah secara *real-time*.

1.2 Rumusan Masalah

Berdasarkan latar belakang, permasalahan yang hendak diselesaikan dapat dirumuskan sebagai berikut:

1. Bagaimana mengimplementasikan algoritma *You Only Look Once (YOLO)* versi 7 untuk mendeteksi penyusup dan mengirimkan notifikasi *alert* secara *real-time* ?
2. Berapa tingkat keakuratan dari YOLOv7 untuk mendeteksi penyusup berdasarkan citra gambar ?

1.3 Batasan Permasalahan

Batasan masalah dalam penelitian ini dapat diuraikan menjadi beberapa poin sebagai berikut:

1. Sistem dibuat menggunakan *framework* YOLO versi 7.
2. Hasil *output* dari deteksi penyusup berupa notifikasi *alert* yang akan dikirimkan menuju *e-mail* dan telegram.

3. Penyusup yang dimaksud adalah manusia yang masuk ke dalam rumah pada saat rumah sedang kosong.
4. Penggunaan algoritma YOLO hanya mendeteksi penyusup dengan *dataset* yang berasal dari Roboflow.

1.4 Tujuan Penelitian

Tujuan penelitian yang ingin dicapai dapat diuraikan menjadi beberapa poin sebagai berikut:

1. Mengimplementasikan algoritma YOLO untuk mendeteksi penyusup *real-time* berdasarkan citra gambar.
2. Melihat tingkat keakuratan dari YOLOv7 untuk mendeteksi penyusup berdasarkan citra gambar.

1.5 Manfaat Penelitian

Manfaat dari penelitian dapat diuraikan menjadi beberapa poin sebagai berikut:

1. Menghasilkan sebuah sistem notifikasi *alert* secara *real-time* kepada pengguna jika terdeteksi adanya penyusup yang tertangkap pada kamera.

1.6 Sistematika Penulisan

Berisikan uraian singkat mengenai struktur isi penulisan laporan penelitian, dimulai dari Pendahuluan hingga Simpulan dan Saran.

Sistematika penulisan laporan adalah sebagai berikut:

- Bab 1 PENDAHULUAN
Berisikan latar belakang dari penelitian yang berlangsung. Latar belakang mencakup penjelasan mengenai YOLOv7, masalah yang ingin diatasi dengan adanya penelitian ini, dan hasil yang ingin dicapai dari penelitian.
- Bab 2 LANDASAN TEORI
Berisikan tinjauan-tinjauan teori berdasarkan literatur yang mendukung penelitian berlangsung. Literatur pendukung terkait *Machine learning*, CNN, dan YOLO.

- Bab 3 METODOLOGI PENELITIAN

Berisikan metodologi penelitian yang dilakukan selama penelitian berlangsung dimulai dengan studi literatur, analisa kebutuhan, pengumpulan *dataset*, perancangan sistem, pengembangan sistem, evaluasi sistem, dan penulisan laporan.

- Bab 4 HASIL DAN DISKUSI

Berisikan hasil penelitian dan diskusi terkait hasil yang dicapai dari penelitian selama penelitian berlangsung.

- Bab 5 KESIMPULAN DAN SARAN

Berisikan kesimpulan dari hasil penelitian yang didapan dan pemberian saran terhadap peneliti selanjutnya jika ingin menggunakan penelitian ini sebagai referensi penelitian selanjutnya.

