

**IMPLEMENTASI METODE SSD (SINGLE SHOT MULTIBOX
DETECTOR) UNTUK MENDETEKSI PELANGGARAN JALUR
BUSWAY MENGGUNAKAN MASUKAN CITRA DIGITAL**

NASKAH PUBLIKASI TUGAS AKHIR



Disusun oleh :

**AGUS SUSILO
5150711022**

**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO
UNIVERSITAS TEKNOLOGI YOGYAKARTA
2019**

**HALAMAN PENGESAHAN
NASKAH PUBLIKASI TUGAS AKHIR MAHASISWA**

Judul Tugas Akhir:

**RANCANG BANGUN SISTEM PENDETEKSI PELANGGARAN PADA JALUR
BUSWAY BERBASIS CITRA DIGITAL MENGGUNAKAN METODE SSD (SINGLE
SHOT MULTIBOX DETECTOR)**

Judul Naskah Publikasi:

**IMPLEMENTASI METODE SSD (SINGLE SHOT MULTIBOX DETECTOR)
UNTUK MENDETEKSI PELANGGARAN JALUR BUSWAY MENGGUNAKAN
MASUKAN CITRA DIGITAL**

Disusun oleh:

AGUS SUSILO

5150711022

Mengetahui,

Nama	Jabatan	Tanda Tangan	Tanggal
M.S. Hendriyawan A, S.T., M.Eng	Pembimbing

Naskah Publikasi Tugas Akhir ini telah diterima sebagai salah satu persyaratan untuk memperoleh gelar Sarjana pada Program Studi Teknik Elektro

Yogyakarta,2019

Ketua Program Studi Teknik Elektro

Fakultas Teknologi Informasi dan Elektro, Universitas Teknologi Yogyakarta

M.S Hendriyawan Achmad,S.T.,M.Eng

NIK. 110810056

PERNYATAAN PUBLIKASI

Yang bertanda tangan di bawah ini, Saya:

Nama : Agus Susilo
NIM : 5150711022
Program Studi : Teknik Elektro
Fakultas : Teknologi Informasi dan Elektro

**“IMPLEMENTASI METODE SSD (SINGLE SHOT MULTIBOX DETECTOR)
UNTUK MENDETEKSI PELANGGARAN JALUR BUSWAY MENGGUNAKAN
MASUKAN CITRA DIGITAL”**

Menyatakan bahwa Naskah Publikasi ini hanya akan dipublikasikan di JURNAL FTIE UTY dan tidak akan dipublikasikan di jurnal orang lain.

Demikian surat pernyataan ini dibuat dengan sebenar-benarnya

Yogyakarta, 17 Juni 2019

Penulis,

Agus Susilo
5150711022

IMPLEMENTASI METODE SSD (SINGLE SHOT MULTIBOX DETECTOR) UNTUK MENDETEKSI PELANGGARAN JALUR BUSWAY MENGGUNAKAN MASUKAN CITRA DIGITAL

Agus Susilo, M.S.Hendriyawan A

*Program Studi Teknik Elektro, Fakultas Teknologi Informasi dan Elektro
Universitas Teknologi Yogyakarta
Jl. Ringroad Utara Jombor Sleman Yogyakarta
E-mail : agussusilo7697@gmail.com*

ABSTRAK

Pada zaman sekarang ini masih banyak sebuah pelanggaran lalu lintas yang terjadi dimana saja. Banyak pelanggaran yang dilakukan oleh pengendara baik mobil maupun motor yang tidak memperhatikan rambu-rambu lalu lintas serta memasuki jalur busway khusus bus Transjakarta. Adapun permasalahan yang dihadapi pada penelitian ini adalah bagaimana sistem dapat membedakan antara bus Transjakarta dengan kendaraan lain, yaitu mobil dan motor berbasis citra digital. Penelitian ini menggunakan metode SSD (Single Shot MultiBox Detector) untuk proses Training dataset serta Running sistem, komponen yang digunakan pada penelitian ini berupa laptop, python IDLE sebagai aplikasi untuk membuat koding dari keseluruhan penelitian, library tensorflow yang terdapat sebuah deep learning sebagai proses ekstraksi fitur, kamera webcam untuk mendeteksi objek kendaraan. Kemudian dilakukan pengujian empat skenario secara langsung pada prototype, apabila terdeteksi kendaraan berupa mobil dan motor pada jalur busway maka telah terjadi pelanggaran, ketika bus Transjakarta yang melaju pada jalur busway maka tidak terdeteksi adanya sebuah pelanggaran atau bisa disebut aman. Adapun output dari pendeteksian ini berupa pemberitahuan sebuah text yang bertuliskan "Pelanggaran" dan "Aman" pada display lcd laptop pemantau. Hasil dari penelitian ini dilakukan sebuah pengujian akurasi deteksi objek kendaraan dari dataset yang sudah ada berupa foto bus, mobil, dan motor yang didapatkan rata-rata persentasenya adalah 98,2%. Sehingga akurasi dataset kendaraan tersebut mencapai target yaitu > 70%.

Kata kunci : Citra Digital, Dataset, Deep Learning, Tensorflow, SSD (Single Shot MultiBox Detector)

1. PENDAHULUAN

Pada zaman sekarang ini para pengendara kendaraan beroda dua maupun beroda empat sudah jarang memperhatikan rambu-rambu lalu lintas yang ada. Saat ini banyak pengendara melakukan cara pintas agar dapat cepat sampai di tujuan. Cara pintas yang dimaksud disini adalah menerobos, melanggar lalu lintas yang sudah ditetapkan oleh UU Lalu Lintas nomer 22 tahun 2009 pasal 287 ayat 1 mengenai rambu-rambu lalu lintas, akan tetapi hal tersebut secara tidak disadari oleh pengendara ketika melakukan pelanggaran dapat menimbulkan sebuah kerugian baik untuk pribadi maupun orang lain.

Dengan salah satu alasan tersebut diatas, maka penelitian ini dimaksudkan untuk merancang sebuah sistem pendeteksi pelanggaran pada jalur Transjakarta, menggunakan kamera webcam untuk proses pemantauan dan input data objek gambar, dengan judul Rancang Bangun Sistem Pendeteksi Pelanggaran Pada Jalur Busway Berbasis Citra Digital Menggunakan Metode SSD (Single Shot

MultiBox Detector). Sistem deteksi ini dipasang pada perpotongan separator jalur busway yang dianggap merupakan tempat yang sangat strategis karena kebanyakan kendaraan non-Transjakarta memasuki jalur khusus busway dari tempat tersebut. Sistem ini dapat membedakan antara bus Transjakarta dengan kendaraan lain berupa mobil, dan motor yang melaju pada jalur busway. Dengan dasar pemikiran tersebut diatas, peneliti berkeyakinan dengan adanya suatu sistem pendeteksi pelanggaran, maka tingkat pelanggaran dan kecelakaan pada jalur busway Transjakarta akan semakin berkurang.

2. METODOLOGI

2.1 Alat dan Bahan

Dalam penelitian ini, penulis menggunakan beberapa komponen alat dan bahan untuk membuat alat proyek tugas akhir. Berikut komponen alat dan bahan yang digunakan:

1. Laptop

Dalam melakukan training dataset diperlukan sebuah tensorflow yang diinstall pada GPU dengan

compute capability minimal adalah 3.5 (Kepler, Maxwell, Pascal, Volta). Oleh karena itu laptop digunakan sebagai alat untuk melakukan *training dataset*.

2. Kamera Webcam

Pada proyek tugas akhir ini menggunakan kamera webcam yang berfungsi untuk mendeteksi objek berupa mobil dan motor serta bus Transjakarta dengan keadaan secara *real-time* serta untuk memudahkan kita dalam mengolah pesan cepat seperti chat melalui video atau bertatap muka melalui video secara langsung. Webcam juga berfungsi sebagai alat untuk men-transfer sebuah media secara langsung.

3. Python IDLE

Python merupakan bahasa pemrograman dengan tujuan umum yang dikembangkan secara khusus untuk membuat *source code* mudah dibaca. Python juga memiliki library yang lengkap sehingga memungkinkan programmer untuk membuat aplikasi yang mutakhir dengan menggunakan *source code* yang tampak sederhana. Jadi aplikasi software yang digunakan untuk memprogram atau untuk membuat koding dari keseluruhan tugas akhir yang penulis buat ini adalah Python IDLE (*Integrated DeveLopment Environment*) editor bawaan Python yang bisa digunakan untuk menuliskan kode Python. IDLE memiliki mode shell (mode interaktif) yang bisa digunakan menjalankan perintah-perintah dengan hasil keluarannya langsung ditampilkan.

4. Tensorflow

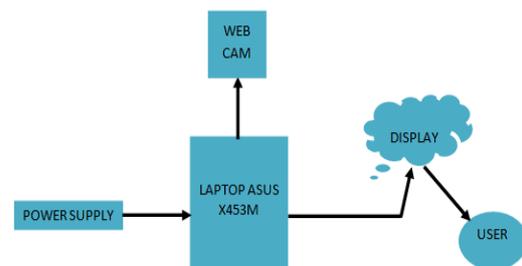
Merupakan kerangka dasar yang digunakan dalam proses *machine learning* serta sebagai *library* khusus untuk *machine learning* yang dikembangkan oleh google. Arsitektur yang fleksibel memungkinkan penerapan komputasi yang mudah di terapkan di berbagai platform seperti (CPU,GPU,TPU) dengan ekstraksi CUDA untuk komputasi tujuan umum pada unit pemrosesan grafis. Tensorflow ini membuat penelitian secara umum dan terbuka.

2.2 Deskripsi Sistem

Menghadapi masalah pelanggaran pada jalur *busway* ini dengan cara melakukan pendeteksian objek kendaraan selain bus transjakarta yang melintas pada jalur *busway* melainkan yang melintas ialah kendaraan lain. Pemasangan sistem ini dilakukan pada *separator* atau perpotongan jalan menuju jalur *busway*, sistem ini terdiri dari Laptop dan juga kamera webcam. Kemudian sistem akan muncul sebuah pemberitahuan berupa text yang bertuliskan “Pelanggaran” dan “Aman” pada display Lcd laptop, sehingga petugas dapat segera melakukan penindakan. Adapun sebuah diagram blok dan flowchart pada sistem ini untuk merancang sebuah program tersebut antara lain:

1. Diagram Blok

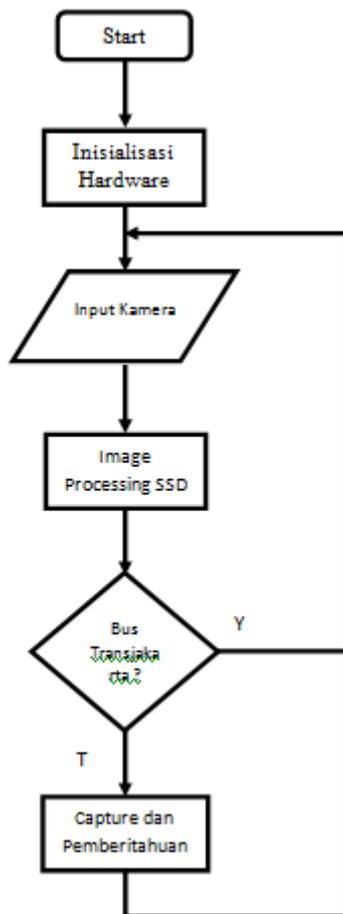
Adapun komponen yang saling berelasi antara lain yaitu, *power supply* atau sumber ini menghidupkan sebuah laptop, kemudian kamera webcam terhubung pada *port usb* dari laptop yang berfungsi sebagai input yang akan menangkap gambar pada jalur *busway*, kemudian gambar yang telah ditangkap akan di program pada laptop dan dideteksi objek apakah bus, mobil atau motor yang melaju pada jalur *busway* tersebut. Outputnya berupa sebuah *display* pada LCD Laptop atau PC yang akan menampilkan sebuah hasil pemrosesan input dari kamera webcam. Kemudian pada *display* tersebut akan ditampilkan berupa jenis objek kendaraan yang terdeteksi serta text bertuliskan “Pelanggaran” jika terjadi sebuah pelanggaran dan akan bertuliskan “Aman” ketika tidak terjadi pelanggaran. User tersebut adalah seorang petugas yang dapat memantau jalur *busway* pada sebuah laptop, apabila terjadi pelanggaran dapat langsung dilakukan penindakan secara langsung. Hubungan antara komponen ini dinyatakan dalam bentuk diagram blok seperti Gambar 2.1.



Gambar 2.1 Diagram Blok Sistem

2. Flowchart Sistem

Pada bagian ini akan dibahas secara umum mengenai konsep dan cara kerja dari sistem pendeteksi bus Transjakarta dan kendaraan lain berdasarkan bentuk kendaraan menggunakan deteksi objek berbasis *deep learning* menggunakan metode SSD (*Single Shot MultiBox Detector*). Sistem bekerja sesuai dengan kendaraan yang melintas pada jalur *busway*, apabila pada jalur *busway* ada kendaraan lain selain bus Transjakarta maka sistem akan mendeteksi adanya sebuah pelanggaran dan kemudian pada hasil *capture image* akan muncul text yang bertuliskan sebuah “Pelanggaran”, jika kendaraan yang melaju pada jalur *busway* itu bus Transjakarta kemudian pada hasil *capture image* akan muncul text yang bertuliskan sebuah “Aman”. Berikut *flowchart* dari cara kerja sistem dapat dilihat pada Gambar 2.2:



Gambar 2.2 Flowchart Pengujian Sistem

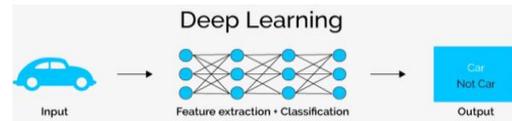
2.3 Open CV

OpenCV (*Open Source Computer Vision Library*) adalah sebuah pustaka perangkat lunak yang ditujukan untuk pengolahan citra dinamis secara real-time, yang pengembangannya diawali oleh Intel, dan sekarang didukung oleh Willow Garage dan Itseez. OpenCV dirilis dibawah lisensi permisif BSD yang lebih bebas dari pada GPL, dan memberikan kebebasan sepenuhnya untuk dimanfaatkan secara komersil tanpa perlu mengungkapkan kode sumbernya. Open CV juga memiliki antar muka yang mendukung bahasa pemrograman C++, C, Python dan Java, termasuk untuk sistem operasi Windows, Linux, Mac OS, iOS dan Android. OpenCV didesain untuk efisiensi dalam komputasi dan difokuskan pada aplikasi real-time.

2.4 Deep Learning

Deep Learning (Pembelajaran Dalam) atau sering dikenal dengan istilah Pembelajaran Struktural Mendalam (*Deep Structured Learning*) atau Pembelajaran Hierarki (*Hierarchical Learning*) adalah salah satu cabang dari ilmu pembelajaran mesin (*Machine Learning*) yang terdiri dari algoritma pemodelan abstraksi tingkat tinggi pada data menggunakan sekumpulan fungsi transformasi non-linear yang ditata berlapis-lapis dan

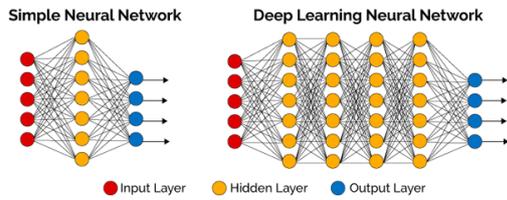
mendalam. Teknik dan algoritma dalam pembelajaran dapat digunakan baik untuk kebutuhan pembelajaran terarah (*supervised learning*), pembelajaran tak terarah (*unsupervised learning*) dan semi terarah (*semi-supervised learning*) dalam berbagai aplikasi seperti pengenalan citra, pengenalan suara, klasifikasi teks, dan sebagainya. *Deep Learning* disebut sebagai *Deep* (dalam) karena struktur dan jumlah jaringan saraf pada algoritmanya sangat banyak bisa mencapai hingga ratusan lapisan. Pemodelan jaringan *deep learning* ditunjukkan oleh Gambar 2.3.



Gambar 2.3 Pemodelan Jaringan Deep Learning

Deep Learning adalah salah satu jenis algoritma jaringan saraf tiruan yang menggunakan metadata sebagai input dan mengolahnya menggunakan dengan sejumlah lapisan tersembunyi (*hidden layer*) transformasi non linier dari data masukan untuk menghitung nilai output. Algoritma pada *Deep Learning* memiliki fitur yang unik yaitu sebuah fitur yang mampu mengekstraksi secara otomatis. Hal ini berarti algoritma yang dimilikinya secara otomatis dapat menangkap fitur yang relevan sebagai keperluan dalam pemecahan suatu masalah. Dalam jaringan saraf tiruan tipe *Deep Learning* setiap lapisan tersembunyi bertanggung jawab untuk melatih serangkaian fitur unik berdasarkan output dari jaringan sebelumnya.

Adapun perbedaan dari Jaringan Saraf Tiruan dengan *Deep Learning* antara lain yaitu, pada Jaringan Saraf Tiruan arsitektur jaringan yang dimilikinya kurang kompleks dan membutuhkan lebih banyak informasi tentang data input sehingga dapat menentukan algoritma mana yang dapat digunakan. Dalam jaringan saraf tiruan terdiri dari beberapa algoritma yaitu Model Hebb, Perceptron, Adaline, Propagasi Maju. Sedangkan pada algoritma jaringan saraf *Deep Learning* tidak memerlukan informasi apapun terhadap data yang akan dipelajarinya, dan algoritma dapat secara mandiri melakukan *tuning* (penyetelan) dan memilih model yang paling optimal. Perbedaan antara lapisan layer pada Jaringan Saraf Tiruan dengan Jaringan *Deep Learning* ditunjukkan oleh Gambar 2.4.



Gambar 2.4 Perbedaan antara lapisan layer pada Jaringan Saraf Tiruan dengan Jaringan Deep Learning

2.5 Skenario Pengujian

Dalam proyek tugas akhir ini ada 4 kondisi pengujian alat dengan menggunakan prototype secara langsung maupun dari laptop. Adapun hasil dari pengujian nantinya output berupa sebuah *display* pada LCD Laptop yang akan menampilkan sebuah hasil pemrosesan input dari kamera webcam. Adapun pengujian kondisinya dapat dilihat pada Tabel 2.1 antara lain sebagai berikut:

Tabel 2.1 Skenario Pengujian Alat

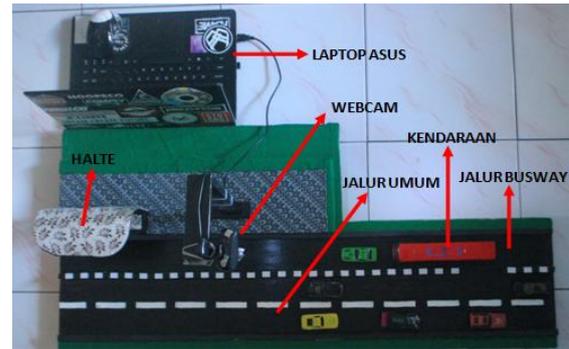
NO	INPUT		OUTPUT
	Transjakarta	Mobil / Motor	
1	Tidak Ada	Tidak Ada	Aman
2	Tidak Ada	Ada	Pelanggaran
3	Ada	Tidak Ada	Aman
4	Ada	Ada	Pelanggaran

3. HASIL dan PEMBAHASAN

3.1 Test Bed

1. Purwarupa

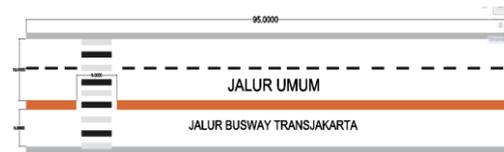
Dalam tugas akhir ini penulis membuat sebuah rancang bangun yang bisa disebut purwarupa, purwarupa ini adalah sebagai alat pengujian secara langsung dengan diameter yang kecil dengan menggunakan papan dan properti yang lain dengan hasil pengujian yang sesuai dengan apa yang dibuat dalam menyelesaikan sebuah masalah yang ada serta sebagai solusi untuk kedepannya. Purwarupa yang penulis buat ini berfungsi sebagai alat pendeteksi pelanggaran pada jalur *busway*, jadi apabila pada jalur *busway* terdapat kendaraan selain *busway* yang melintas, maka akan terindikasi telah terjadinya sebuah pelanggaran, yang kemudian akan muncul pemberitahuan berupa text “Pelanggaran” atau “Aman” pada *image capture* yang ada pada display LCD laptop pemantau. Berikut purwarupa lalu lintas jalur *busway* yang ditunjukkan pada Gambar 3.1.



Gambar 3.1 Lalu Lintas Jalur Busway

2. Rancangan Lebar Jalur Busway

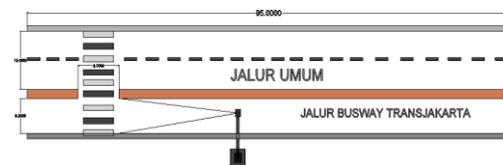
Pada rancangan lebar jalan pada jalur *busway* ini perbandingannya 1 : 100. Lebar jalan yang saya buat dengan ukuran 6 cm untuk jalur *busway*, dan 10 cm untuk jalan umum. Adapula sebuah *separator* atau bisa disebut perpotongan jalan ini dengan ukuran 8 cm, *separator* ini yang biasanya untuk masuk para pelanggar jalur *busway*. Rancangan mekanik jalur *busway* dapat dilihat pada Gambar 3.2.



Gambar 3.2 Rancangan Jalur Busway

3. Penempatan USB Webcam

Penempatan kamera pemantau atau bisa disebut webcam pada proyek akhir ini berada setelah *separator* atau perpotongan jalan dari jalur utama menuju jalur *busway*, kamera webcam ini tepat diatas dan posisinya di tengah-tengah agar mudah dalam proses penangkapan sebuah pelanggaran kendaraan mobil dan motor yang melaju pada jalur *busway*. Berikut penempatan USB Webcam dapat dilihat pada Gambar 3.3.



Gambar 3.3 Rancangan USB Webcam

3.2 Hasil Training Datasets

Training dataset ini dilakukan menggunakan tensorflow versi GPU yang diinstal pada laptop ASUS A456UR dengan kartu grafis NVIDIA GeForce 930MX. Dalam pengujian perangkat lunak pada proyek tugas akhir ini mencakup sebuah dataset kendaraan serta akurasi deteksi objek kendaraan. Penulis membuat sebuah dataset kendaraan ini secara manual dengan foto asli berupa mobil, bus, dan motor mainan, jadi dataset kendaraan ini bukan mengambil dari internet, menurut Ren, S. dkk., (2016) dalam proses training ini harus didapatkan total loss atau nilai error yang sangat kecil kisaran 0,05, karena sangat berpengaruh terhadap nilai keakuratan pendeteksian. Total loss dapat dilihat pada Gambar 3.4.

```
Anaconda Prompt - python train.py --logstoderr --train_dir=training/ --pipeline_config_
INFO:tensorflow:global step 51836: loss = 0.1403 (1.252 sec/step)
INFO:tensorflow:global step 51837: loss = 0.0442 (1.253 sec/step)
INFO:tensorflow:global step 51837: loss = 0.0442 (1.253 sec/step)
```

Gambar 3.4 Akhir Pelatihan Dataset

Kemudian setelah dataset di *training* langkah selanjutnya yaitu pengujian dataset. Pengujian dataset ini dilakukan sebagai acuan dalam proses pelatihan data atau *training* dataset, dalam proses *training* ini menggunakan 151 gambar atau foto mainan secara langsung dari posisi depan, samping kanan-kiri, dan atas, dari tiap fitur dalam satu mainan. Jadi dengan jumlah gambar sebanyak 151 ini, dataset yang dihasilkan sudah cukup untuk proses pendeteksian objek kendaraan dengan cukup akurat. Sumber dataset yang digunakan dalam *training* dataset adalah berupa hasil foto mainan yang berbentuk kendaraan, yaitu bus, mobil, dan motor mainan. Adapun beberapa daftar sumber gambar dataset kendaraan yang penulis gunakan untuk *training* dataset dapat ditunjukkan pada Tabel 3.1.

Tabel 3.1 Sumber Dataset Kendaraan

Jenis	Gambar Kendaraan
Bus Transjakarta	
Mobil	

Tabel 3.1 Sumber Dataset Kendaraan(Lanjutan)

Jenis	Gambar Kendaraan
Mobil	
Motor	

3.3 Uji Akurasi

1. Pengujian Akurasi Deteksi Objek

Ketika sumber dataset kendaraan sudah di *training*, kemudian dilakukan pengujian terhadap tingkat akurasi dari dataset tersebut, dari pengujian tersebut dapat dilihat apakah sistem pendeteksi ini dapat mendeteksi objek kendaraan berupa mainan yang ada pada suatu gambar secara akurat atau tidak akurat. Pengujian akurasi deteksi objek ini dilakukan terhadap 10 sampel gambar yang berbeda-beda yang dimana terdapat objek kendaraan didalamnya. Kemudian hasil dari pengujian akan diketahui persentase akurasi kecocokan terhadap objek deteksi dengan dataset yang sudah ada. Berikut hasil pengujian akurasi deteksi objek yang ditunjukkan pada Tabel 3.2.

Tabel 3.2 Pengujian Akurasi Deteksi Objek

Gambar Asli	Hasil Deteksi Objek	Akurasi %
		99%
		99%
		99%
		99%
		99%
		97%
		99%

Tabel 3.2 Pengujian Akurasi Deteksi Objek(Lanjutan)

Gambar Asli	Hasil Deteksi Objek	Akurasi %
		93%
		99%
		99%
Rata-rata Akurasi Deteksi (%)		98.2%

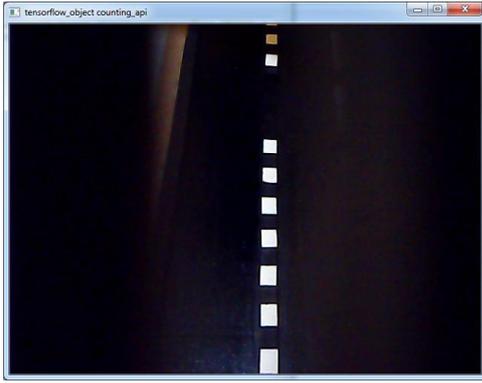
Berdasarkan hasil dari Tabel 3.2 pengujian deteksi kendaraan diatas terdapat pengujian beberapa objek kendaraan yaitu 10 sampel gambar, hasil pengujian didapatkan rata-rata persentase akurasi adalah 98.2%. Sehingga akurasi dataset kendaraan tersebut mencapai target yaitu >70%.

3.4 Uji Deteksi Pelanggaran

Setelah dilakukan proses pengujian perangkat lunak berupa pengujian terhadap deteksi akurasi objek, selanjutnya dilakukan pengujian terhadap alat yang berupa simulasi pada prototype jalur *busway*. Pengujian ini ada 4 kondisi sekenario alat yang mendeteksi objek pada jalur *busway*.

1. Kondisi Pertama

Kondisi sekenario alat ini ketika pada jalur *busway* tidak terdapat Bus Transjakarta, mobil dan motor yang melaju. Sehingga tidak terdeteksi adanya sebuah pelanggaran serta output berupa sebuah text pada *image capture* yang ada pada display LCD Laptop pemantau tidak akan muncul. Berikut hasil dari pengujian ditunjukkan pada Gambar 3.4 dari hasil *running*, Gambar 3.5 tampak atas, Gambar 3.6 tampak samping kiri, dan Gambar 3.7 tampak samping kanan.



Gambar 3.4 Tanpa Objek Kendaraan



Gambar 3.5 Tampak Depan Atas



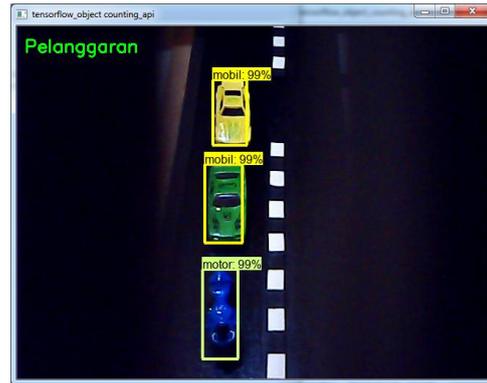
Gambar 3.6 Tampak Samping Kiri



Gambar 3.7 Tampak Samping Kanan

2. Kondisi Kedua

Kondisi sekenario alat ini ketika pada jalur *busway* tidak terdapat Bus Transjakarta melainkan terdapat Mobil atau Motor yang melaju pada jalur *busway*. Sehingga terdeteksi adanya sebuah pelanggaran, kemudian akan muncul berupa sebuah text “**Pelanggaran**” pada *image capture* yang ada pada display LCD Laptop pemantau. Berikut hasil dari pengujian ditunjukkan pada Gambar 3.8 dari hasil *running*, Gambar 3.9 tampak atas, Gambar 3.10 tampak samping kiri, dan Gambar 3.11 tampak samping kanan.



Gambar 3.8 Objek Mobil dan Motor



Gambar 3.9 Tampak Atas



Gambar 3.10 Tampak Samping Kiri



Gambar 3.11 Tampak Samping Kanan



Gambar 3.14 Tampak Samping Kiri

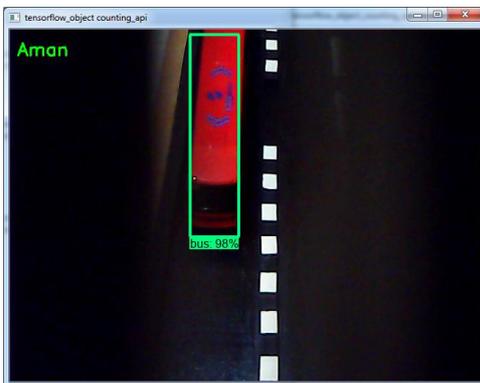
3. Kondisi Ketiga

Pada kondisi ketiga ini ada 2 penerapan, yang diantaranya sebagai berikut:

1. Kondisi skenario alat ini ketika pada jalur busway terdapat Bus Transjakarta melainkan tidak terdapat Mobil atau Motor yang melaju pada jalur busway. Sehingga tidak terdeteksi adanya sebuah pelanggaran, yang kemudian akan muncul berupa sebuah text “Aman” pada image capture yang ada pada display LCD Laptop pemantau. Berikut hasil dari pengujian ditunjukkan pada Gambar 3.12 dari hasil running, Gambar 3.13 tampak atas, Gambar 3.14 tampak samping kiri, dan Gambar 3.15 tampak samping kanan.



Gambar 3.15 Tampak Samping Kanan



Gambar 3.12 Objek BusTransjakarta

2. Pada kondisi skenario alat ini terdapat sebuah Bus Pariwisata yang melaju pada jalur busway, tetapi tidak terindikasi sebuah pelanggaran, karena pada jalur busway ini untuk Bus Pariwisata diperbolehkan melaju pada jalur busway Transjakarta tersebut. Namun pada hasil pendeteksian ini Bus Pariwisata tidak terdeteksi sebagai objek kendaraan Bus Pariwisata, melainkan terdeteksi (N/A) atau bisa disebut objek itu tidak diketahui, karena awalnya hanya mendeteksi Bus Transjakarta saja, jadi pada kondisi ini pendeteksian dapat membedakan antara Bus Transjakarta dengan Bus Pariwisata berdasarkan kelas kategori (name class) pada objek bus tersebut. Hasil pendeteksian berupa Bus Pariwisata yang kemudian akan muncul berupa sebuah text “Aman” pada image capture yang ada pada display LCD Laptop pemantau. Berikut hasil dari pengujian ditunjukkan pada Gambar 3.16 dari hasil running, Gambar 3.17 tampak atas, Gambar 3.18 tampak samping kiri, dan Gambar 3.19 tampak samping kanan.



Gambar 3.13 Tampak Atas



Gambar 3.16 Objek BusPariwisata



Gambar 3.17 Tampak Atas



Gambar 3.18 Tampak Samping Kiri

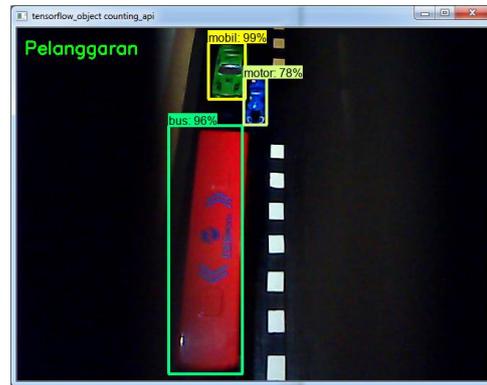


Gambar 3.19 Tampak Samping Kanan

4. Kondisi Keempat

Pada kondisi keempat ini ada 2 penerapan, yang diantaranya sebagai berikut:

1. Pada kondisi skenario alat ini ketika pada jalur busway terdapat Bus Transjakarta serta terdapat Mobil atau Motor yang melaju di belakang bus pada jalur busway. Sehingga terdeteksi adanya sebuah pelanggaran, kemudian akan muncul berupa sebuah text “Pelanggaran” pada image capture yang ada pada display LCD Laptop pemantau. Berikut hasil dari pengujian ditunjukkan pada Gambar 3.20 hasil running, Gambar 3.21 tampak atas, Gambar 3.22 tampak samping kiri, Gambar 3.23 tampak samping kanan.



Gambar 3.20 Objek Bus, Motor dan Mobil



Gambar 3.21 Tampak Atas



Gambar 3.22 Tampak Samping Kiri



Gambar 3.23 Tampak Samping Kanan

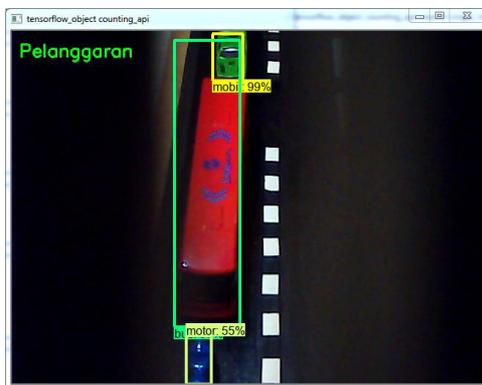


Gambar 3.26 Tampak Samping Kiri

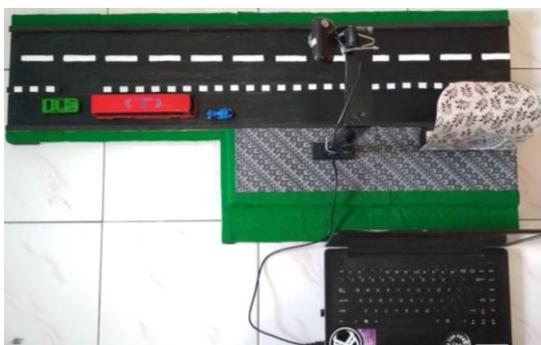
2. Pada kondisi skenario alat ini ketika pada jalur busway terdapat Motor yang melaju di depan Bus Transjakarta serta terdapat Mobil a yang melaju di belakang bus pada jalur busway ini. Sehingga terdeteksi adanya sebuah pelanggaran, kemudian akan muncul berupa sebuah text “Pelanggaran” pada image capture yang ada pada display LCD Laptop pemantau. dan Gambar 3.24 hasil running, Gambar 3.25 tampak atas, Gambar 3.26 tampak samping kiri, dan Gambar 3.27 tampak samping kanan.



Gambar3.27 Tampak Samping Kanan



Gambar 3.24 Objek Motor, Bus dan Mobil



Gambar 3.25 Tampak Atas

4. PENUTUP

4.1 Kesimpulan

Kesimpulan dari laporan tugas akhir ini adalah sistem dapat mendeteksi adanya sebuah pelanggaran yang ada pada jalur busway Transjakarta. Didapatkan hasil persentase pengujian akurasi deteksi objek sebesar 98.2% dalam 10 sampel foto, sehingga sistem dapat mendeteksi jenis objek kendaraan yang diinginkan berdasarkan masukan citra digital serta akurasi dataset kendaraan tersebut melebihi target >70%. Dalam skenario pengujian alat ini ada 4 kondisi yaitu ketika ada sebuah kendaraan yang melaju di jalur busway selain bus Transjakarta dan bus Pariwisata maka terdeteksi “PELANGGARAN”, apabila di jalur busway tersebut yang melintas adalah bus Transjakarta maupun bus Pariwisata maka terdeteksi “AMAN”. Hasil dari pendeteksian tersebut berupa pemberitahuan pada capture serta text yang bertuliskan “PELANGGARAN” dan “AMAN” pada display lcd laptop pemantau.

4.2 Saran

Adapun saran untuk penelitian yang selanjutnya adalah sebagai berikut:

1. Dalam proses pelatihan dataset, akan lebih cepat apabila menggunakan GPU dengan tingkat *compute capability* lebih dari 6.0.
2. Menggunakan lebih banyak foto sebagai sumber pelatihan dataset, agar didapatkan hasil yang lebih akurat dalam mendeteksi objek.
3. Pada proses Training dataset harus cukup lama agar total loss semakin kecil, karena total loss ini sangat berpengaruh terhadap keakuratan pendeteksian objek.

PERSEMBAHAN

Puji syukur dipanjatkan atas kehadiran Allah SWT, karena dengan limpahan karunia-Nya penulis dapat menyelesaikan penelitian ini.

Terima kasih kepada bapak M.S. Hendriyawan Achmad.,S.T.,M.Eng selaku Ketua Program Studi Teknik Elektro Fakultas Teknologi Informasi dan Elektro dan selaku dosen pembimbing, serta seluruh dosen Program Studi Teknik Elektro, Universitas Teknologi Yogyakarta.

DAFTAR PUSTAKA

- [1] Amelia, Sarwoko, M. dan Riza, A.T. (2013), *Perancangan Sistem Deteksi Jenis Kendaraan Bermotor Berbasis Mikrokontroler (Studi Kasus: Jalur Transjakarta)*, *Jurnal Teknologi*, Vol 6, 90–97.
- [2] Hernawan dan Arief (2006), *Jaringan Saraf Tiruan Teori dan Aplikasi*, Yogyakarta: ANDI.
- [3] Irfan, M. (2017), *Klasifikasi Kendaraan Berbasis Pengolahan Citra Digital dengan Metode Multilayer Perceptron*, *Indonesian Journal of Electronics and Instrumentations Systems*, Vol 7 (2), 139–148.
- [4] Lecun, Y., Bengio, Y. dan Hinton, G. (2015), *Deep Learning*, .
- [5] Li, D. (2014), *Deep Learning: Methods and Applications, Foundations and Trends® in Signal Processing*, Vol 7(3–4), 197–387.
- [6] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. dan Berg, A.C. (2016), *SSD: Single Shot Multibox Detector*, *International Journal of European Conference on Computer Vision*, Vol 5.
- [7] Murni (2004), *Pengolahan Citra*, Bandung.
- [8] Nurfita, R.D. (2018), *Implementasi Deep Learning Berbasis Tensorflow Untuk Pengenalan Sidik Jari*, Universitas Muhammadiyah Surakarta.
- [9] Ren, S., He, K., Girshick, R. dan Sun, J. (2016), *Towards Real-Time Object Detection with Region Proposal Networks*, , Volume 3, 1–9.
- [10] Samuel, A. (1959), *Machine Learning*, .
- [11] Sogen, M.D.T. (2015), *Rancang Bangun Purwarupa Sistem Pendeteksi Kendaraan Menggunakan Pustaka OpenCV*, Politeknik Katolik Saint Paul.