

## DETEKSI OBJEK MENGGUNAKAN DASHBOARD CAMERA UNTUK SISTEM PERINGATAN PENCEGAH KECELAKAAN PADA MOBIL

Raymond Sutjiadi<sup>\*1</sup>, Timothy John Pattiasina<sup>2</sup>

<sup>1</sup>Program Studi Teknik Informatika, Institut Informatika Indonesia Surabaya

<sup>2</sup>Program Studi Sistem Informasi, Institut Informatika Indonesia Surabaya

Email: <sup>1</sup>raymond@ikado.ac.id, <sup>2</sup>temmy@ikado.ac.id

\*Penulis Korespondensi

(Naskah masuk: 01 Oktober 2019, diterima untuk diterbitkan: 11 Februari 2020)

### Abstrak

Saat ini penggunaan dashboard camera marak digunakan pada mobil untuk merekam kondisi sekitar kendaraan ketika berkendara. Dashboard camera adalah semacam kamera yang ditempatkan pada bagian dashboard mobil dengan kamera menyorot ke arah depan kendaraan yang berfungsi untuk merekam kondisi jalan. Di lain pihak, pada mobil premium saat ini sudah disematkan beberapa teknologi canggih untuk mencegah terjadinya kecelakaan atau tabrakan yang biasa disebut dengan Forward Collision Warning System. Teknologi ini pada dasarnya berfungsi untuk mencegah terjadinya tabrakan dari arah depan, baik dengan cara aktif ataupun pasif. Pada penelitian ini akan dibuat sebuah sistem terintegrasi dimana dashboard camera, yang diimplementasikan menggunakan kamera smartphone berbasis Android, tidak hanya digunakan untuk perekaman secara statis, tetapi juga digunakan untuk membuat sistem pencegah kecelakaan secara pasif. Adapun aplikasi ini dibuat dengan menggunakan metode Object Detection Model yang ada pada Tensorflow Open Source Machine Learning Library. Tahapan riset yang dilakukan dimulai dari proses analisis permasalahan, studi pustaka untuk mencari pembandingan dari penelitian sebelumnya, serta dilakukan implementasi pembuatan perangkat lunak dan diakhiri dengan pengujian untuk mengukur kinerja sistem. Dari hasil pengujian tampak bahwa aplikasi ini mampu mendeteksi objek kendaraan berupa mobil penumpang, bus, dan truk, serta dapat memberikan peringatan baik secara visual maupun alarm apabila kendaraan di depan sudah berada pada jarak yang cukup dekat untuk memperingatkan pengemudi akan bahaya tabrakan.

**Kata kunci:** dashboard camera, deteksi objek, forward collision warning system, pengolahan citra digital, tensorflow

## OBJECT DETECTION USING DASHBOARD CAMERA FOR CAR ACCIDENT PREVENTION SYSTEM

### Abstract

*Nowadays dashboard camera becomes familiar to be used in a car to record the condition around the vehicle while driving. Dashboard camera is a video camera placed in car's dashboard faces in front of the vehicle to record the road condition. In the other side, premium cars now equipped with advanced technology to prevent collision called Forward Collision Warning System. This technology basically acts to prevent front collision, either in active or passive ways. In this research was built integrated system where dashboard camera, which implemented by camera of Android based smartphone, not only used as static recording, but also as passive collision avoidance system. This application was developed using Object Detection Method in Tensorflow Open Source Machine Learning Library. The research stage was started from problem analysis, literature study to search comparison from previous research, also software development and finalized with testing to measure system performance. From the testing result, this application was able to detect vehicle objects in form of passenger car, bus, and truck, also could provide both visual and alarm warning when there was a vehicle come closely from in front, to warn the driver about the danger of collision.*

**Keywords:** dashboard camera, object detection, forward collision warning system, digital image processing, tensorflow

## 1. PENDAHULUAN

Saat ini penggunaan *dashboard camera* marak digunakan pada mobil untuk merekam kondisi sekitar kendaraan ketika berkendara. *Dashboard camera* adalah semacam kamera yang ditempatkan pada bagian *dashboard* mobil dengan kamera menyorot ke arah depan kendaraan yang berfungsi untuk merekam kondisi jalan. Selayaknya kamera CCTV pada gedung, penggunaan *dashboard camera* ini menjadi penting manakala terjadi insiden di jalan, dimana rekaman yang ada bisa dijadikan alat bukti yang meyakinkan guna pembuktian pihak-pihak yang terlibat dalam insiden kecelakaan (Hnoohom & Thanapattherakul, 2016).

Di lain pihak, pada mobil premium saat ini sudah disematkan beberapa teknologi canggih untuk mencegah kecelakaan atau tabrakan yang biasa disebut dengan *Forward Collision Warning System* (Yadav, Kancharla & Nair, 2011). Teknologi ini pada dasarnya berfungsi untuk mencegah terjadinya tabrakan dari arah depan, baik dengan cara aktif ataupun pasif. Cara aktif adalah sistem mampu melakukan manipulasi pada gerak mobil, misal melakukan pengereman secara otomatis apabila mobil terlalu dekat dengan objek di depan atau memanipulasi gerak setir apabila mobil terdeteksi keluar jalur. Sedangkan cara pasif adalah sistem akan memperingatkan pengemudi, baik melalui suara alarm, indikator lampu peringatan, getaran pada setir atau cara lain, yang bertujuan untuk menyadarkan pengemudi untuk segera melakukan pengereman kendaraan.

Teknologi *Forward Collision Warning System* ini dibuat dengan memasang sensor-sensor di bagian depan mobil. Sensor tersebut bisa menggunakan gelombang radio elektromagnetik (radar), sinar laser, kamera, atau gabungan dari beberapa jenis sensor (Adla, Al-Holou, Murad & Bazzi, 2013). Dimana tugas dari sensor tersebut adalah melakukan pendeteksian objek yang ada di depan mobil dan oleh sistem komputer dihitung jarak objek tersebut dengan mobil. Jika jarak objek dianggap terlalu dekat dengan mobil, maka sistem akan memicu output sesuai dengan yang dikehendaki, misal mengaktifkan alarm peringatan atau mengaktifkan pengereman. Penggunaan sensor dan sistem komputer ini membutuhkan biaya yang cukup mahal dan tidak dapat diterapkan pada semua mobil, khususnya pada mobil yang belum memiliki sistem komputer yang canggih (*drive-by-wire system*).

Pada penelitian ini akan dibuat sebuah sistem terintegrasi dimana *dashboard camera*, yang diimplementasikan menggunakan kamera *smartphone* berbasis sistem operasi Android, tidak hanya digunakan untuk keperluan perekaman secara statis, tetapi juga digunakan untuk membuat sistem peringatan pencegah kecelakaan secara pasif dengan memanfaatkan metode pengolahan citra digital untuk mendeteksi keberadaan objek di depan mobil. Proses pengolahan data dilakukan dengan menggunakan

*Tensorflow Open Source Machine Learning Library*. Selain pendeteksian dan pengklasifikasian objek, jarak objek juga diukur untuk kemudian ditentukan apakah jarak objek tersebut sudah cukup dekat dengan mobil sehingga perlu mengaktifkan alarm dan/atau tampilan peringatan kepada pengemudi untuk segera melakukan aksi pengereman.

Harapannya sistem peringatan pencegah kecelakaan ini juga bisa diterapkan pada mobil-mobil non premium yang belum mendapatkan fitur *Forward Collision Warning System* bawaan dari pabrik. Selain itu *dashboard camera* yang diimplementasikan menggunakan kamera *smartphone* berbasis Android sebagai sistem pencegah kecelakaan ini juga akan lebih fungsional dan ekonomis karena menyatukan fungsi perekaman sekaligus pendeteksian objek di depan mobil.

## 2. METODE PENELITIAN

### 2.1. *Forward Collision Warning System*

Pada tahun 2002, perusahaan Mercedes-Benz yang berada di Amerika Serikat, pertama kali memperkenalkan *Forward Collision Warning System* (FCW) (Cicchino, 2017), hal tersebut menjadi titik tolak kemajuan teknologi di bidang keselamatan transportasi. Sebenarnya teknologi ini telah lama digunakan di navigasi pesawat udara untuk mendeteksi keberadaan objek di depan guna mencegah terjadinya tabrakan dari arah depan. Teknologi yang serupa kini banyak dijumpai di mobil-mobil premium dengan tujuan yang sama, yaitu mencegah terjadinya tabrakan dengan objek di depan mobil. Sistem FCW ini juga sering disebut dengan nama lain seperti *Pre-safe Braking System*, *Collision Warning with Auto-Brake System*, *Pre-Crash Warning Systems*, dan *Collision Mitigation Braking System*, dimana fungsi dan kapasitas dari masing-masing sistem tersebut dapat bervariasi, meskipun memiliki tujuan umum yang sama yaitu untuk mencegah tabrakan dari arah depan.

Secara umum teknologi FCW sendiri dapat dibagi menjadi 3 (tiga) bagian utama, yang terdiri atas suatu sensor yang memiliki tugas untuk mendeteksi keberadaan objek di depan mobil, bagian *computer system* yang bertugas untuk membaca data hasil pendeteksian sensor dan menterjemahkannya menjadi *output*, serta bagian aktuator yang bertugas untuk melakukan tindakan aktif berdasarkan perintah komputer. Khusus pada bagian sensor dapat dibagi dalam beberapa jenis, antara lain *Vehicle Speed Sensor* (VSS) dan *Throttle Position Sensor* (TPS). VSS memiliki kegunaan untuk sensor yang mampu mendeteksi kecepatan kendaraan dimana sensor tersebut dipasang di blok transmisi. Sedangkan TPS dipakai untuk melakukan pendeteksian pada saat proses pembukaan pedal gas (*accelerator valve*) yang dikontrol langsung oleh pengemudi kendaraan.

Salah satu metode yang digunakan dalam menganalisa sistem FCW adalah *time-to-contact*

(TTC) (Horn, Fang & Masaki, 2009) untuk memunculkan peringatan pada saat nilai TTC lebih rendah dari ambang waktu tertentu, misalnya 2 detik. Momentum TTC ( $T_m$ ) dapat dihitung dengan menggunakan rumus:

$$T_m = -Z/V \quad (1)$$

dimana  $Z$  adalah jarak relatif ke target dan  $V$  adalah kecepatan relatif. Akan tetapi karena jarak dan kecepatan kendaraan relatif bukanlah sesuatu yang dapat diukur dengan tepat, mengingat kendaraan merupakan objek yang bergerak, maka dengan memanfaatkan kamera dapat diberikan perhitungan momentum TTC yang lebih tepat untuk mengeluarkan model proyeksi perspektif dari objek kendaraan dengan rumus:

$$W_t = -fW/Z_t \quad (2)$$

dimana  $W_t$  adalah lebar target pada gambar yang muncul di kamera pada waktu ( $t$ ),  $Z_t$  adalah jarak ke target,  $W$  adalah lebar kendaraan dan  $f$  adalah panjang fokus kamera.

## 2.2. Citral Digital & Motion Detection

Jika ditinjau dari pengertiannya, suatu citra (*image*) adalah bentuk representasi, imitasi atau dapat dikatakan sebagai kemiripan dari suatu objek. Pada hakekatnya representasi suatu citra dapat dilakukan melalui cara analog maupun digital. Suatu citra analog dapat diartikan sebagai suatu bentuk citra yang memiliki sifat kontinu dan tidak digambarkan pada perangkat komputer, dimana hal tersebut mengakibatkan pemrosesan citra ini tidak dapat dilakukan secara langsung. Sedangkan pengertian dari citra digital sendiri ialah suatu bentuk citra yang memiliki kemampuan untuk diolah menggunakan perangkat komputer. Pada dasarnya suatu bentuk citra (gambar) digital dapat direpresentasikan ke dalam bentuk matriks yang memiliki bagian  $X$  baris serta  $Y$  kolom. Perpotongan antara baris dan kolom yang terjadi pada citra dalam bentuk matriks ini dinamakan *picture element (pixel)*. *Pixel* dapat diartikan sebagai suatu satuan pemetaan gambar terkecil yang dapat dilihat pada layar monitor, dimana untuk masing-masing *node* yang tampak dalam komputer grafik, diwakilkan oleh satu atau lebih pixel yang digambarkan pada suatu layar monitor. Dalam pelaksanaan penelitian yang dilakukan ini, bentuk representasi citra yang dipakai adalah citra digital. Adapun representasi dari bentuk citra digital dengan ukuran  $N$  baris serta  $M$  kolom dapat dilihat dalam persamaan dibawah ini:

$$f_{(x,y)} = \begin{bmatrix} f_{(0,0)} & f_{(0,1)} & \dots & f_{(0,M-1)} \\ f_{(1,0)} & \dots & \dots & f_{(1,M-1)} \\ \dots & \dots & \dots & \dots \\ f_{(N-1,0)} & f_{(N-1,1)} & \dots & f_{(N-1,M-1)} \end{bmatrix} \quad (3)$$

dimana  $f(x,y)$  adalah nilai intensitas suatu *pixel* dalam koordinat  $(x,y)$ .

*Motion detection* memiliki prinsip dasar yaitu dengan cara melakukan perbandingan antara 2 (dua) buah citra yang dinyatakan dalam koordinat  $f(x,y,t_1)$  dan  $f(x,y,t_2)$ , yang menghasilkan suatu citra baru  $r(x,y)$ , dimana citra tersebut akan memiliki nilai 0 (hitam) atau 1 (putih) dengan kriteria yang telah ditetapkan seperti yang dapat dilihat berikut ini:

$$\begin{cases} 1 & \text{jika } |f(x,y,t_1) - f(x,y,t_2)| < T \\ 0 & \text{jika } |f(x,y,t_1) - f(x,y,t_2)| \geq T \end{cases} \quad (4)$$

dimana  $T$  adalah nilai *threshold*, dengan besar yang tergantung pada kepekaan terhadap perubahan yang diinginkan (Ihzan, Widyawan & Edi, 2012). Adapun citra baru yang dihasilkan dari perbandingan tersebut akan mengandung suatu bentuk objek yang dapat bergerak.

## 2.3. Computer Vision

Sebagai salah satu bagian dari panca indera yang terdapat pada tubuh manusia, mata dapat diibaratkan sebagai suatu sistem yang canggih dengan kemampuan untuk melakukan serta menerima suatu respon atas suatu bentuk *visual stimulus*. Jika dilihat dari fungsinya, *computer vision* dan *human sense of sight* dapat dikatakan memiliki kesamaan dalam mencapai tujuan untuk melakukan terjemahan suatu bentuk data spasial, yang adalah bentuk data dengan indeks lebih dari satu dimensi. Akan tetapi *computer vision* sendiri tidak memiliki kemampuan dalam hal melakukan replika yang sama persis seperti yang dapat dilakukan oleh mata seorang manusia. Hal tersebut dapat terjadi karena pengetahuan yang dimiliki oleh *computer vision* terhadap mekanisme atau cara kerja sistem mata dan otak manusia belum dapat diterjemahkan secara maksimal, sehingga mengakibatkan kesulitan dalam hal merancang sebuah sistem untuk melakukan replikasi yang tepat sesuai fungsi dari mata itu sendiri. Adapun yang menjadi inti dari penerapan teknologi *computer vision* adalah kemampuan suatu sistem komputer untuk melakukan duplikasi dari penglihatan manusia kedalam suatu perangkat elektronik, dimana pada akhirnya perangkat elektronik tersebut akan memiliki pemahaman serta pengertian terhadap inputan dari suatu objek berupa gambar yang disajikan.

*Computer Vision System (CVS)* dapat melakukan replikasi, serta dalam penerapannya di beberapa kasus, CVS juga memiliki kemampuan untuk memperbaiki kelemahan dari sistem penglihatan seorang manusia. Dalam penerapannya CVS memiliki bagian yang terdiri atas komputer sebagai bentuk analog dari otak seorang manusia, dan sensor atau kamera digital analog sebagai bentuk lain dari indera penglihatan manusia, serta suatu sistem

penerangan yang merupakan sumber cahaya sebagai fasilitator dari akuisisi suatu citra digital. 3 (tiga) operasi utama yang terdapat dalam CVS secara umum terdiri atas akuisisi citra (*image acquisition*), pengolahan citra (*image processing*), serta penerjemahan atau pemahaman citra (*image understanding*) (Susmita & Malviya, 2013).

Dalam bentuk yang lebih rinci, tahapan-tahapan yang dilalui pada proses analisis citra dimulai dari tahap awal yang merupakan tahapan pembentukan suatu citra, dimana citra akan diambil untuk kemudian disimpan ke dalam komputer. Selanjutnya pada tahapan pengolahan awal citra, kualitas citra digital diperbaiki untuk meningkatkan detail citra. Tahapan berikutnya adalah segmentasi citra, dimana citra objek diidentifikasi dan dipisahkan dari latar belakangnya, kemudian dilanjutkan dengan tahapan pengukuran citra, yaitu tahapan dimana beberapa fitur-fitur yang signifikan dikuantitaskan. Pada tahapan terakhir adalah interpretasi dari citra, dimana fitur-fitur yang sebelumnya diekstrak akan diinterpretasikan lebih lanjut.

## 2.4. Tensorflow

*Tensorflow* adalah *open source library* untuk *machine learning* yang diperkenalkan oleh Google dan mendukung beberapa bahasa pemrograman (Devikar, 2016). Saat ini *tensorflow* sangat berperan pada penelitian skala besar dalam *machine learning* dan penelitian *deep neural network* yang membutuhkan komputasi paralel secara masif. Dengan cara melakukan penggabungan atas suatu bentuk aljabar komputasi dan teknik kompilasi optimal, pemanfaatan *tensorflow* akan memberikan kemudahan dalam proses penghitungan ekspresi matematika yang majemuk, dimana permasalahan yang kerap terjadi adalah lamanya waktu proses yang dibutuhkan dalam menyelesaikan bentuk perhitungan matematika. *Tensorflow* sendiri dapat digunakan pada semua sistem operasi.

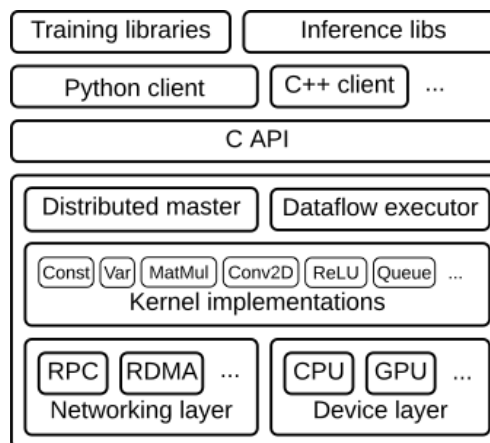
Fitur utama yang dimiliki oleh *tensorflow* meliputi:

1. Pendefinisian, pengoptimalan serta proses perhitungan yang dilakukan secara efisien dalam bentuk ekspresi matematis yang juga melibatkan bentuk *multidimensional array* atau disebut dengan istilah *tensors*.
2. Fitur pembelajaran mesin serta fitur pemrograman yang mendukung komputasi jaringan syaraf dalam. *Tensorflow* dapat menarik pengetahuan yang diberikan kedalamnya untuk dipergunakan dalam proses penyelesaian tugas perhitungan ekspresi matematika.
3. *Graphics Processing Unit* (GPU) yang transparan, serta kemampuan optimalisasi dan otomatisasi manajemen memori yang sama dengan data yang dipakai. Dimana *tensorflow* memiliki kemampuan untuk menuliskan *similarity code* serta dapat menjalankan dengan

maksimal pada bagian *Central Processing Unit* (CPU) ataupun pada bagian GPU. Lebih khususnya lagi, *tensorflow* dapat mengetahui mana saja bagian perhitungan yang akan dan harus ditransfer ke dalam GPU.

4. *Large data sets* serta *high computing scalability* pada seluruh bagiannya, sehingga memberikan akurasi yang baik dalam pemanfaatannya.

Arsitektur umum dari *Tensorflow* dapat dilihat pada Gambar 1 di bawah ini:



Gambar 1. Arsitektur Umum *Tensorflow* ([www.tensorflow.org](http://www.tensorflow.org))

## 3. PERANCANGAN SISTEM

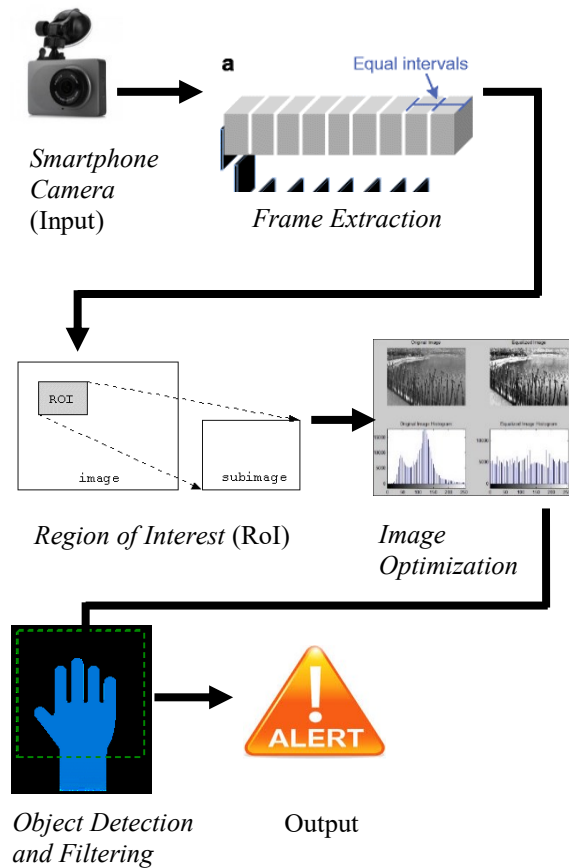
Penelitian yang dilakukan oleh Kilicarslan, dkk menggunakan kamera video untuk mendeteksi jarak objek yang ada di depan mobil. Akan tetapi tidak dilakukan pendeteksian dan pengklasifikasian objek tersebut. Pendeteksian dilakukan menggunakan *motion profiling* untuk menangkap adanya pergerakan objek pada video (Kilicarslan & Zheng, 2017). Kelemahan dari sistem tersebut adalah kemungkinan terjadinya *false positive error* pendeteksian apabila ada hal yang berubah dari kontur jalan karena sistem membaca perubahan tersebut sebagai objek.

Pada artikel penelitian Kim, dkk., dilakukan pendeteksian objek menggunakan kamera video dengan metode *optical flow*. Dimana dengan metode ini, objek dideteksi dengan cara mengekstraksi *background* menggunakan *Mixture of Gaussians Model* (Kim & Kwon, 2016). Dengan kata lain objek dideteksi berdasarkan perubahan yang terjadi pada *background*. Banyak hal yang bisa mengakibatkan terjadinya kesalahan pembacaan seperti keberadaan bayangan dan perubahan pencahayaan.

Pada penelitian ini tidak hanya dilakukan pendeteksian objek, tetapi juga dilakukan pengklasifikasian jenis objek tersebut. Dalam lingkup penelitian ini, objek yang dideteksi dan diklasifikasikan hanya mencakup 3 jenis kendaraan bermotor, yaitu kendaraan penumpang (*car*), *bus*, dan *truck*.

Pendeteksian objek menggunakan *dashboard camera* yang diimplementasikan dengan

menggunakan kamera pada perangkat Android untuk sistem peringatan pencegah kecelakaan pada mobil terdiri dari 6 bagian utama seperti terlihat pada Gambar 2 di bawah ini, yaitu:



Gambar 2. Blok Diagram Sistem

1. **Input**  
Sebagai input dari sistem digunakan kamera belakang pada perangkat *smartphone* berbasis Android. Kamera haruslah diletakkan sedemikian rupa agar bisa mencakup padangan bagian depan dari mobil, yaitu dari sisi sudut sebelah kiri sampai dengan sudut sebelah kanan dari mobil. Upayakan tidak ada objek lain yang menghalangi pandangan kamera seperti hiasan, *sticker*, atau pernak-pernik pada *dashboard* dan kaca depan mobil yang dapat mendistraksi pendeteksian objek.
2. **Frame Extraction**  
Pada bagian ini video hasil tangkapan kamera akan dikonversi menjadi *frame* citra penyusunnya. Dimana setiap *frame* citra ini akan masuk ke tahap berikutnya.
3. **Region of Interest (RoI)**  
Pada bagian ini setiap *frame* akan dilakukan penentuan *Region of Interest* (RoI). Tujuan dari penentuan RoI ini adalah untuk membatasi atau meminimalkan area yang menjadi pendeteksian objek sehingga tidak memperberat sistem dan

mengurangi kemungkinan terjadinya *noise* yang dapat mempengaruhi keakurasian sistem.

#### 4. *Image Optimization*

Di bagian ini setiap *frame* citra akan dioptimalisasi agar tercipta komposisi citra yang baik sebelum masuk ke tahap berikutnya. Komposisi citra yang baik akan membantu meningkatkan keakurasian sistem.

#### 5. *Object Detection & Filtering*

Bagian ini adalah inti dari sistem. Dimana pada proses ini dilakukan pendeteksian dan pengklasifikasian objek pada kamera. Untuk proses pendeteksian dan pengklasifikasian objek dilakukan dengan menggunakan *Object Detection Model* yang ada pada *Tensorflow Open Source Machine Learning Library*. Pada penelitian ini tidak membahas proses *re-train* untuk pendeteksian dan pengklasifikasian objek, melainkan langsung menggunakan *pre-trained quantized COCO SSD MobileNet v1 model* bawaan dari *Tensorflow* yang sudah cukup baik akurasi. Selain itu pada bagian ini juga dilakukan *filtering* berdasarkan jenis kendaraan yang menjadi *interest* pada sistem ini, yaitu mobil, bus, dan truk, sehingga hanya ketiga objek itu saja yang akan diproses lebih lanjut. Sedangkan objek lain akan diabaikan. Objek yang terdeteksi dengan *confidence level* lebih dari 0,8 (80 persen) akan diambil dan akan ditandai dengan *blob* berupa kotak persegi empat yang mengelilingi objek tersebut.

#### 6. *Output*

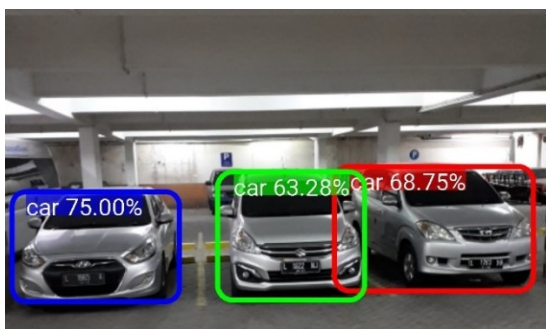
Hasil pendeteksian dan klasifikasi kendaraan bermotor pada tahap sebelumnya akan dihitung jaraknya. Untuk perhitungan jarak dilakukan dengan menggunakan pendekatan empiris dan tidak dilakukan kalibrasi secara pasti. Hal ini dilakukan karena ukuran/dimensi masing-masing kendaraan bermotor berbeda-beda. Dengan demikian masing-masing jenis kendaraan bermotor tersebut harus memiliki perhitungan kalibrasi yang berbeda-beda pula. Untuk menyederhanakan cara kerja sistem, dilakukan pendekatan empiris dengan cara membaca dimensi lebar dari *object blob detection*. Mobil penumpang, truk dan bus memiliki lebar yang relatif sama. Sedangkan kamera diposisikan untuk menangkap pandangan dari kiri ke kanan lebar pandangan kaca depan mobil. Karena itu apabila lebar dari *blob detection* sudah melewati ambang batas 90 persen dari lebar *frame* kamera secara keseluruhan, maka bisa dikatakan bahwa objek tersebut sudah cukup dekat. Sebaliknya apabila lebar dari *blob detection* masih di bawah ambang batas tersebut, maka bisa disimpulkan jarak kendaraan tersebut masih cukup jauh. Apabila jarak kendaraan sudah terdeteksi cukup dekat, maka selanjutnya sistem akan melakukan

*trigger* berupa bunyi alarm serta menampilkan pesan peringatan pada layar.

#### 4. HASIL DAN PEMBAHASAN

Proses uji coba dari sistem yang dibuat ini dilakukan dengan 2 (dua) cara, yaitu pengujian hasil pendeteksian serta pengklasifikasian objek kendaraan bermotor serta pengujian fungsi sistem peringatan pencegah kecelakaan.

Pengujian pendeteksian dan pengklasifikasian objek kendaraan bermotor dilakukan dalam kondisi statis untuk mengenali kendaraan bermotor dengan tipe mobil penumpang dan bus. Apabila ada mobil penumpang dan bus yang tampil pada *frame* video dan sistem mengenalinya, maka objek tersebut akan ditandai dengan *blob detection* atau kemunculan kotak disekelilingnya serta ditampilkan teks jenis kendaraan dan nilai *confidence* pendeteksian sistem.



Gambar 3. Hasil Pendeteksian Objek Kendaraan Dengan Jenis Mobil Penumpang (MPV)



Gambar 4. Hasil Pendeteksian Objek Kendaraan Dengan Jenis Mobil Penumpang (Sedan)



Gambar 5. Hasil Pendeteksian Objek Kendaraan Dengan Jenis Bus (Minibus)

Pada Gambar 3 dan 4 terlihat bahwa sistem mampu mengenali dan mengklasifikasikan lebih dari satu jenis kendaraan mobil penumpang (MPV dan sedan) sebagai *Car* dengan nilai *confidence* diatas 60 persen. Adapun *frame* video mengambil tampilan gambar mobil tersebut dari arah depan.

Sedangkan pada Gambar 5 terlihat bahwa sistem juga mampu mengenali lebih dari satu jenis kendaraan minibus. Terdapat perbedaan hasil pengklasifikasian pada minibus sebelah kiri dan kanan. Minibus sebelah kiri diklasifikasikan oleh sistem sebagai *Car* dengan nilai *confidence* 64,45 persen. Sedangkan yang sebelah kanan diklasifikasikan oleh sistem sebagai *Bus* dengan nilai *confidence* 62,11 persen. Akan tetapi hasil yang berbeda ini masih bisa ditoleransi karena sistem masih dapat mendeteksi kedua kendaraan bermotor tersebut.

Untuk pengujian fungsi sistem peringatan pencegah kecelakaan dilakukan dengan menggunakan objek mobil penumpang. Tampak pada Gambar 6 bahwa dimensi lebar *blob detection* (kotak warna biru) yang mengelilingi mobil belum mencapai 90 persen dari lebar *frame* secara keseluruhan. Dengan demikian dapat disimpulkan bahwa mobil tersebut masih dalam jarak yang cukup jauh dengan mobil yang dipasang kamera sehingga belum ada *trigger* berupa tulisan visual dan alarm. Dalam gambar tersebut juga tampak beberapa objek kendaraan lain yang terdeteksi di posisi samping dan belakang.

Sedangkan pada Gambar 7, terlihat bahwa dimensi lebar *blob detection* (kotak warna biru) sudah melewati ambang batas 90 persen dari keseluruhan lebar *frame*. Dengan demikian sistem akan memberikan *trigger* sistem peringatan pasif berupa tulisan "STOP! Press the Brake!" pada tampilan layar visual sekaligus mengaktifkan bunyi alarm pada perangkat *smartphone*.



Gambar 6. Posisi Mobil Masih dalam Jarak Aman





Gambar 7. Posisi Mobil Dalam Jarak Cukup Dekat

## 5. KESIMPULAN

Dari hasil pengembangan perangkat lunak dan pengujian yang dilakukan terlihat bahwa sistem ini dapat mendeteksi dan mengklasifikasikan 3 jenis kendaraan bermotor, yaitu mobil, bus, dan truk, dengan tingkat akurasi yang baik. Selain itu sistem juga telah mampu memberikan peringatan secara visual dan alarm ketika kendaraan yang ada di depan sudah berada pada posisi yang cukup dekat. Penggunaan kamera pada perangkat *smartphone* berbasis Android untuk fungsi peringatan pencegah kecelakaan secara pasif terbukti efektif dan efisien untuk diterapkan. Aplikasi ini bisa menjadi salah satu alternatif sederhana untuk diterapkan pada mobil non premium yang belum mendapatkan fitur *Collision Avoidance System* bawaan dari pabrik.

## 6. UCAPAN TERIMA KASIH

Peneliti mengucapkan terima kasih kepada Direktorat Riset dan Pengabdian Masyarakat, Direktorat Jenderal Penguatan Riset dan Pengembangan, Kementerian Riset, Teknologi, dan Pendidikan Tinggi yang telah membiayai penelitian ini melalui skema Penelitian Dosen Pemula Tahun Anggaran 2019 berdasarkan kontrak penelitian nomor 001/SP2H/LT/P3M/IKADO/IV/2019.

## DAFTAR PUSTAKA

- ADLA, R., AL-HOLOU, N., MURAD, M. & BAZZI, Y.A., 2013. Automotive Collision Avoidance Methodologies: Sensor-Based and ITS-Based. ACS International Conference on Computer Systems and Applications (AICCSA), pp. 1-8.
- CICCHINO, J.B., 2017. Effectiveness of Forward Collision Warning and Autonomous Emergency Braking Systems in Reducing Front-to-rear Crash Rates. Accident Analysis and Prevention, vol. 99, pp. 142–152.
- DEVIKAR, P. 2016. Transfer Learning for Image Classification of Various Dog Breeds. International Journal of Advanced Research in Computer Engineering and Technology (IJARCET), Vol.5, pp. 2707-2715.
- HNOOHOM, N. & THANAPATTHERAKUL, T., 2016. Image Problem Classification for Dashboard Cameras. IEEE Xplore: 2016 12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS).
- HORN, B.K.P., FANG, Y. & MASAKI, I., 2009. Hierarchical Framework for Direct Gradient-based Time-to-contact Estimation. IEEE Intelligent Vehicle Symposium, pp. 1394-1400.
- IHZAN, M.Z., WIDYAWAN & EDI, L.N., 2012. Deteksi Gerak Dengan Menggunakan Metode Frame Differences Pada IP Camera. Proceedings of International Conference on Information Technology and Electrical Engineering, pp. 52-55.
- KILICARSLAN, M. & ZHENG, J.Y., 2017. Direct Vehicle Collision Detection from Motion in Driving Video. 2017 IEEE Intelligent Vehicles Symposium, pp. 1558-15564.
- KIM, D.S. & Kwon, J., 2016. Moving Object Detection on a Vehicle Mounted Back-Up Camera. Sensors, 16(1), 23.
- SUSMITA, A.M. & MALVIYA, A.V., 2013. Traffic Surveillance by Counting and Classification of Vehicles from Video Using Image Processing. International Journal of Advance Research in Computer Science and Management Studies, 1(6), 2321-7782.
- YADAV, G.K., KANCHARLA, T. & NAIR, S., 2011. Real Time Vehicle Detection for Rear and Forward Collision Warning Systems. ACC 2011: Advances in Computing and Communications, pp. 368-377.

*Halaman ini sengaja dikosongkan*