

DETEKSI DAN KLASIFIKASI MEREK MOBIL UNTUK PENENTUAN IKLAN BILLBOARD MENGGUNAKAN CONVOLUTION NEURAL NETWORK

Windra Swastika*¹, Ardian Kurniawan², Hendry Setiawan³

^{1,2,3}Fakultas Sains dan Teknologi, Universitas Ma Chung
Email: ¹windra.swastika@machung.ac.id, ²ardian.smbc@gmail.com, ³hendry.setiawan@machung.ac.id
*Penulis Korespondensi

(Naskah masuk: 4 Juli 2019, diterima untuk diterbitkan: 17 Agustus 2019)

Abstrak

Dunia periklanan di Indonesia saat ini memiliki perkembangan yang sangat pesat. Hal ini dibuktikan dengan semakin bertambah banyaknya media periklanan yang diciptakan, salah satunya adalah iklan billboard pada jalan raya. Iklan billboard ini memiliki kelemahan, yaitu materi atau konten dari iklan yang ditampilkan tidak dapat berubah-ubah, dengan demikian maka target dari periklanan tidak bisa tertuju pada konsumen yang tepat. Untuk mengatasi masalah tersebut maka dibutuhkan pemanfaatan teknologi untuk mendukung keefektifan kinerja dari iklan billboard. Pada penelitian ini dibuat sebuah sistem yang dapat mendeteksi mobil dan mengenali merek dari mobil yang terdeteksi, sehingga materi iklan dapat berubah sesuai dengan merek mobil yang dikenali oleh sistem. Untuk deteksi pada mobil digunakan metode You Only Look Once (YOLO) dan untuk klasifikasi pada merek mobil digunakan metode MiniVGGNet. Proses latih dilakukan dengan menggunakan 1100 buah gambar dan terdapat 11 macam merek mobil yang dapat diklasifikasikan. Dari pengujian yang dilakukan, didapatkan akurasi akhir 93% pada deteksi mobil. Untuk klasifikasi dari merek mobil dilakukan pengujian dengan fungsi optimasi Adam dengan ukuran masukan gambar 64x64 piksel. Untuk akurasi terbaik yang didapatkan adalah 60%.

Kata kunci: *Convolutional Neural Network, You Only Look Once (YOLO), VGGNet, deteksi mobil*

DETECTION AND CLASSIFICATION CAR TYPE TO DETERMINE BILLBOARD ADVERTISING USING CONVOLUTIONAL NEURAL NETWORK

Abstract

The world of advertising in Indonesia today has a very rapid development. This is proven by the increasing number of advertising media created, one example is billboard advertising on the highway. Billboard advertising has a weakness, namely the material or the content of the ads displayed cannot change, therefore the target of advertising cannot be directed at the right consumer. To overcome this problem, the use of technology is needed to support the effectiveness of billboard advertising. In this study a system was created which is can detect the car and recognize the brand of the car detected, so the advertising material can change according to the brand of the car that is recognized by the system. For the detection of cars, using You Only Look Once (YOLO) method and for the classification of car brands, using MiniVGGNet method. The training process is carried out using 1100 pictures and there are 11 kinds of car brands that can be classified. From the tests performed, 93% final accuracy was found in car detection. The classification of the car brand was tested with Adam optimization functions with an image input size of 64x64 pixels. For the best accuracy obtained is 60% using the Adam optimization function with the input image size of 64x64 pixels.

Keywords: *Convolutional Neural Network, You Only Look Once (YOLO), VGGNet, car detection*

1. PENDAHULUAN

Dunia periklanan di Indonesia saat ini memiliki perkembangan yang sangat pesat. Hal ini dibuktikan dengan semakin bertambah banyaknya media periklanan yang diciptakan, salah satunya adalah iklan billboard pada jalan raya. Iklan merupakan setiap kegiatan yang bertujuan untuk

memperkenalkan suatu barang atau jasa dengan maksud untuk menarik perhatian khalayak keramaian atau dinikmati oleh umum. Iklan billboard ini memiliki kelemahan, yaitu materi atau konten dari iklan yang ditampilkan tidak dapat berubah-ubah. Dengan demikian maka target dari

periklanan tidak bisa tertuju pada konsumen yang tepat.

Untuk mengatasi masalah tersebut maka dibutuhkan pemanfaatan teknologi untuk mendukung keefektifan kinerja dari iklan billboard. Salah satu caranya adalah dengan membuat sebuah sistem yang dapat mendeteksi mobil dan mengenali merek dari mobil yang terdeteksi, sehingga materi iklan dapat berubah sesuai dengan merek mobil yang dikenali oleh sistem. Cabang ilmu komputer yang dapat mendukung jalannya sistem tersebut salah satunya adalah Computer Vision.

Munculnya Computer Vision ini sangat berpengaruh dalam kehidupan manusia dalam berbagai macam bidang, seperti salah satu contohnya adalah Face Recognition yang menggantikan fungsi keamanan dari smartphone maupun komputer. Bidang lainnya yaitu Smart City yang juga merupakan tema yang sering menjadi pembahasan dalam berkembangnya Computer Vision. Hal tersebut dibuktikan dengan banyak sekali dilakukan penelitian-penelitian yang mengangkat tema Smart City seperti pada klasifikasi kendaraan dengan metode multilayer perceptron (Irfan dkk, 2017), streaming lalu lintas kendaraan dengan server Raspberry Pi (Rohadi dkk, 2018), Deteksi golongan kendaraan pada jalan tol (Swastika dkk, 2019), pendeteksi jumlah, jenis dan kecepatan kendaraan berbasis Raspberry Pi (Setyawan, 2019).

Untuk penelitian klasifikasi mobil sudah pernah dilakukan oleh (Amaluddin, 2015) yang menggunakan Gaussian Mixture Model (GMM) dan Fuzzy Cluster Mean (FCM). Dalam penelitian tersebut diperlukan proses preprocessing citra yang menjadi dataset. Pada proses preprocessing ini berfungsi untuk memisahkan antara gambar latar dan objek kendaraan, serta melakukan pemotongan atau cropping pada objek yang dideteksi sebagai kendaraan. Dengan adanya proses ekstraksi fitur pada hasil akhir dari penelitian sangat bergantung pada proses ekstraksi fitur itu sendiri. Hal tersebut menjadi kendala dikarenakan proses dari preprocessing tidak selalu berhasil disebabkan oleh berbagai kondisi saat citra gambar tersebut diambil.

Dari permasalahan di atas maka akan dilakukan sebuah penelitian untuk membuat sebuah sistem klasifikasi merek mobil. Klasifikasi merek mobil ini nantinya diharapkan dapat meningkatkan efektifitas penargetan iklan billboard. Penelitian ini memanfaatkan CNN untuk mengidentifikasi merek dari mobil, yang nantinya akan mengganti materi dari iklan billboard dengan sesuatu yang berhubungan dengan merek mobil yang teridentifikasi.

2. METODE PENELITIAN

Secara garis besar, alur penelitian ini dibagi ke dalam tahap-tahap berikut.

2.1. Analisis Kondisi

Pada langkah ini perlu dilakukan analisis apa saja yang diperlukan untuk mendukung jalannya penelitian yang dilakukan. Dalam analisa kondisi ini didapatkan beberapa kondisi yang harus diperhatikan, yang pertama pada saat proses pengumpulan data gambar mobil, diperlukan gambar mobil dengan posisi tampak depan. Kemudian data gambar mobil yang terkumpul akan dipisahkan menjadi 2 bagian, yang bertujuan sebagai data latih untuk deteksi objek mobil, dan sebagai data latih untuk klasifikasi merek dari mobil yang terdeteksi. Kemudian karena resolusi dari data gambar yang terkumpul berbeda-beda, maka diperlukan proses *scaling* untuk menyeragamkan resolusi dari setiap data gambar yang terkumpul.

2.2. Pengumpulan Data

Pada penelitian ini data yang digunakan sebagai sumber informasi untuk CNN yang dibuat adalah citra digital dari beberapa merek mobil yang ingin dikenali. Pengumpulan data akan dilakukan dengan cara melakukan pengambilan gambar mobil yang tampak depan dengan memanfaatkan Google Image Search. Google Image Search merupakan salah satu *search engine* yang telah menerapkan penggunaan *deep learning* dalam proses pencariannya. Dengan menggunakan *deep learning*, Google Image Search dapat melakukan pencarian yang akurat, terutama pada pencarian gambar. Pencarian gambar pada Google Image Search akan dilakukan dengan menggunakan kata kunci untuk 11 jenis mobil yang akan digunakan pada penelitian ini, yaitu: Toyota Avanza, Honda Jazz, Honda Brio, Toyota Innova, Nissan Juke, Suzuki Swift, Daihatsu Xenia, Mitsubishi Pajero, Toyota Alphard, Daihatsu Grandmax, dan Toyota Yaris. Dari setiap jenis mobil tersebut akan diambil 100 buah gambar tampak depan. Setelah data gambar dari setiap merek mobil terkumpul, maka akan dilakukan pengecekan untuk menghindari adanya gambar yang sama. Data gambar yang telah terkumpul nantinya akan sebagai data latih untuk deteksi mobil serta pengenalan dari masing-masing merek mobil. Gambar 1 adalah beberapa contoh dari gambar yang didapatkan dari Google Image Search.



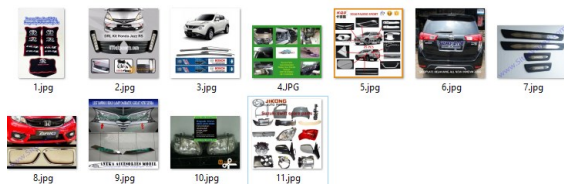
Gambar 1. Contoh gambar data latih

Data citra digital yang terkumpul sebelum dapat digunakan sebagai data latih maupun *input layer* pada CNN, perlu dilakukan sebuah proses yaitu *Preprocessing*. *Preprocessing* merupakan sebuah langkah yang dilakukan dengan tujuan agar citra gambar yang digunakan sesuai dengan kebutuhan *input layer* dari arsitektur yang digunakan. Karena citra gambar yang didapatkan memiliki resolusi atau ukuran yang berbeda-beda, maka dalam proses *preprocessing* ini akan dilakukan proses *scaling*. *Scaling* merupakan sebuah proses yang dilakukan untuk mengubah ukuran dari sebuah citra digital. Untuk contoh proses *scaling* yang dilakukan pada penelitian ini dapat dilihat pada gambar 2.



Gambar 2. *Scaling* pada praproses citra

Kemudian juga dilakukan pengumpulan data untuk contoh iklan yang akan ditampilkan sistem setelah proses klasifikasi selesai. Pengumpulan gambar iklan disesuaikan dengan merek mobil yang dapat diklasifikasi oleh sistem. Untuk contoh gambar dari iklan dapat dilihat pada gambar 3.



Gambar 3. Contoh gambar iklan mobil

2.3. Desain Sistem

Pada tahapan ini akan dibuat desain dari CNN yang akan digunakan dalam deteksi serta pengenalan dari merek kendaraan. Untuk deteksi dari mobil akan digunakan *YOLO object detection* (Redmon dkk, 2015). Sedangkan untuk pengenalan merek mobil akan digunakan metode *MiniVGGNet* (Heryadi dkk, 2019). Metode *MiniVGGNet* dipilih dalam penelitian dengan alasan kebutuhan resource dalam melakukan training relatif tidak terlalu besar dan tingkat akurasi metode *MiniVGGNet* cukup baik seperti nampak pada penelitian Kwon dkk (2017) dan Gezahegn dkk (2019). Untuk alur dari jalannya sistem nantinya akan berjalan diawali dengan memasukkan data gambar yang ingin diklasifikasi, kemudian dilanjutkan dengan proses deteksi objek mobil, setelah objek terdeteksi maka dilanjutkan dengan memotong gambar mobil pada

bagian objek yang terdeteksi, setelah itu dilanjutkan dengan melakukan proses klasifikasi, setelah proses klasifikasi selesai maka akan ditampilkan hasil klasifikasi beserta nilai *confidence*, dan iklan akan ditampilkan sesuai dengan mobil yang berhasil diklasifikasikan.

2.4. Proses Training

Proses *Training* merupakan proses utama dalam CNN yang dapat menentukan hasil dari deteksi maupun klasifikasi baik atau tidak. Pada penelitian ini akan dilakukan 2 kali *training* pada data yang sudah terkumpul. *Training* yang pertama menggunakan 1.100 gambar tanpa dibedakan antara merek mobil. *Training* yang pertama ini nantinya akan berfungsi untuk deteksi objek mobil. Setelah selesai maka dilanjutkan dengan proses *training* yang kedua, yaitu dengan menggunakan 1.100 gambar yang terkumpul dengan membedakan antar merek mobil. Proses *training* yang kedua ini nantinya berfungsi sebagai pengenalan merek mobil yang terdeteksi.

Untuk proses *training* pada metode YOLO, langkah awal yang harus dilakukan adalah memberikan anotasi pada setiap dataset yang sudah terkumpul. Pemberian anotasi yang dimaksud disini adalah memberikan kotak atau yang sering disebut sebagai *bounding box* pada bagian citra yang menandakan bagian dari suatu objek. Untuk contoh pemberian anotasi pada setiap dataset dapat dilihat pada Gambar 4.



Gambar 4. Pemberian anotasi pada dataset YOLO

Untuk proses *training* dari *MiniVGGNet* langkah awal yang harus dilakukan adalah melakukan *crop* dengan cara deteksi objek mobil pada seluruh data latih yang terkumpul. Hal ini dilakukan dengan tujuan agar data latih memiliki ukuran panjang dan lebar yang seimbang sehingga tidak terjadi perubahan bentuk dari objek yang ada pada suatu gambar. Setelah langkah diatas selesai maka dapat dilanjutkan pada proses *training* untuk pembentukan model dari CNN.

Proses *training* dari CNN secara keseluruhan dibagi menjadi dua tahap, yaitu tahap *feedforward* dan tahap *backpropagation* (Simonyan dkk, 2015). Proses *feedforward* adalah proses citra sebagai *input layer* memulai proses dari *convolution layer*, *pooling layer*, hingga sampai memasuki *fully connected layer* yang merupakan lapisan *output*. Hasil dari

proses ini adalah bobot akhir yang merupakan klasifikasi dari citra masukan. Apabila perlu dilakukan perbaikan bobot, maka proses selanjutnya adalah proses *backpropagation*. Proses *backpropagation* ini menggunakan hasil dari proses *feedforward* sebelumnya untuk melakukan proses *feedforward* yang akan dilakukan. Proses *backpropagation* ini berfungsi untuk memperbaiki bobot yang dihasilkan oleh proses *feedforward*

2.5. Uji Coba dan Evaluasi Sistem

Untuk proses perhitungan akurasi akan dilakukan dengan pengujian secara statistik. Pengujian statistik ini menggunakan uji sensitivitas dan uji spesifisitas. Pengujian sensitivitas atau *True Positive Rate* merupakan jenis perhitungan akurasi yang membandingkan jumlah klasifikasi atau identifikasi yang tepat pada suatu *class* dengan seluruh klasifikasi yang terdapat pada *class* tersebut. Pengujian spesifisitas atau *True Negative Rate* merupakan jenis perhitungan akurasi yang membandingkan jumlah klasifikasi yang tidak berhubungan dengan sebuah *class* tetapi bernilai tepat dengan seluruh klasifikasi yang tidak berhubungan pada *class* tersebut. Nilai-nilai dari uji sensitivitas dan spesifisitas nantinya akan memerlukan adanya *Matrix Confusion* untuk perhitungan kedepannya. *Matrix Confusion* biasanya digunakan dalam perhitungan klasifikasi dari beberapa *class*. Untuk contoh bentuk dari *Matrix Confusion* dapat dilihat pada Tabel 1.

Tabel 1. Matrix Confusion untuk klasifikasi 2 variabel

Merek	Klasifikasi waktu(menit)	
	1	2
1	TP	FN
2	FP	TN

Kolom Merek merupakan *class* dari data yang dipakai, sedangkan kolom Hasil Klasifikasi merupakan hasil yang diperoleh dari proses klasifikasi oleh sistem. Apabila Merek nomor 1 teridentifikasi nomor 1, maka akan diberi label *True Positive (TP)*, karena hasil klasifikasi sesuai dengan yang diharapkan pada *Matrix Confusion* di atas maka diberi label *Positive* dan hasil klasifikasi sesuai dengan merek maka diberi label *true*. Apabila Merek nomor 1 teridentifikasi nomor 2, maka akan diberi label *False Negative (FN)*. Apabila Merek nomor 2 teridentifikasi nomor 1, maka akan diberi label *False Positive (FP)*, karena hasil yang diharapkan pada *Matrix Confusion* diatas adalah nomor 1, maka diberi label *positive*, tetapi terhubung hasil klasifikasi tidak sesuai dengan merek yang ada maka diberi label *false*. Apabila merek nomor 2 teridentifikasi nomor 2 pada *Matrix Confusion* yang mengharapakan hasil klasifikasi nomor 1, maka akan diberi label *True Negative (TN)*. Penghitungan akurasi berdasarkan *matrix confusion* dapat dilakukan dengan menggunakan rumus (1).

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Di mana TP dan TN merupakan klasifikasi yang tepat, sedangkan jumlah keseluruhan antara TP, TN, FP, dan FN menghasilkan total jumlah klasifikasi yang dilakukan. Apabila tidak terdapat kesalahan sehingga FP dan FN bernilai 0, yang mengakibatkan hasil akhir menjadi 1. Untuk penghitungan sensitivitas berdasarkan *matrix confusion* dapat dilakukan dengan menggunakan rumus (2).

$$Sensitivitas = \frac{TP}{TP+FN} \quad (2)$$

Di mana TP merupakan klasifikasi yang tepat terhadap keadaan yang bernilai ya, misalnya dalam suatu *Matrix Confusion* diharapkan hasil klasifikasi mobil dengan merek Toyota Innova memiliki sebuah input Toyota Innova, dan ternyata hasil klasifikasi menunjukkan merek Toyota Innova, maka data ini akan diberi label TP. Sedangkan FN merupakan klasifikasi yang salah terhadap keadaan yang bernilai ya, misalnya dalam suatu *Matrix Confusion* diharapkan hasil klasifikasi mobil dengan merek Toyota Innova memiliki sebuah input Toyota Innova, dan ternyata hasil klasifikasi menunjukkan merek Toyota Avanza, maka data ini akan diberi label FN. Untuk penghitungan spesifisitas berdasarkan *matrix confusion* dapat dilakukan dengan menggunakan rumus (3).

$$Spesifisitas = \frac{TN}{TN+FP} \quad (3)$$

Di mana TN merupakan klasifikasi yang tepat terhadap keadaan yang bernilai tidak, misalnya dalam suatu *Matrix Confusion* yang diharapkan hasil klasifikasi mobil dengan merek Toyota Innova memiliki sebuah input Toyota Avanza, dan ternyata hasil klasifikasi menunjukkan merek Toyota Avanza, maka data ini akan diberi label TN. FP merupakan klasifikasi yang salah terhadap keadaan yang bernilai tidak, contohnya citra yang bukan kendaraan tetapi dikenali sebagai kendaraan. Semakin tinggi tingkat spesifisitas, berarti semakin tepat dalam membedakan hal-hal yang bukan merupakan objek tersebut.

3. HASIL DAN PEMBAHASAN

3.1. Pembentukan Model Deteksi dan Klasifikasi

Pada pembentukan model YOLO v2 diawali dengan melakukan pemberian anotasi pada 1100 data latih yang terkumpul. Pemberian anotasi disini berfungsi untuk memberikan tanda pada letak dari sebuah objek pada gambar tertentu. Anotasi dari setiap gambar berisi beberapa parameter, yaitu *width*, *height*, *depth*, *filename* dan letak dari anotasi (*bounding box*) yang disimpan dengan format .xml.

Setelah pemberian anotasi pada data latih yang terkumpul selesai, maka dilanjutkan dengan proses *training* untuk mendeteksi objek yang sudah ditentukan (mobil). Sebelum memulai *training*, ada beberapa hal yang perlu diperhatikan pada konfigurasi YOLO v2, yang pertama adalah jumlah *class* yang ingin dideteksi, dan menentukan bobot atau *weight* yang digunakan, pada penelitian ini menggunakan versi kecil dari bobot yolo yang disebut *tiny yolo weighth*.

Setelah konfigurasi pada YOLO v2 selesai, maka proses *training* dapat dilakukan. Proses *training* untuk pembentukan model YOLO v2 pada penelitian ini berlangsung selama 7 jam dengan 21.000 iterasi dan dapat dilihat pada Gambar 5, bahwa *loss* atau *error rate* pada iterasi 0 sampai 2000 menurun drastis sampai pada nilai 1, dan kemudian menurun secara perlahan pada iterasi ke 2000 sampai ke 21.000 yang mencapai hasil akhir *loss* 0.42. Untuk perhitungan nilai *loss* pada YOLO digunakan fungsi *Sum of Square Error* dengan rumus (4).

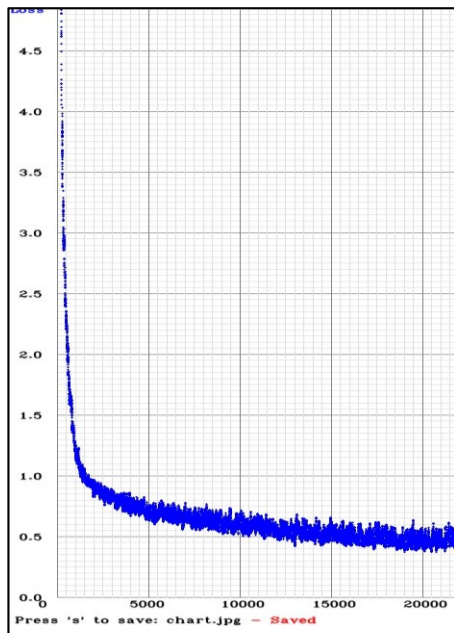
$$SSE = \sum_{i=1}^n (X_i - Y_i) \quad (4)$$

Di mana:

X_i : Nilai pengamatan ke- i

Y_i : Nilai prediksi ke- i

n : jumlah data yang diamati



Sebelum dilakukan proses *training* untuk pembentukan model dari MiniVGGNet, hal pertama yang perlu dilakukan adalah mengimplementasikan deteksi objek YOLO v2 pada seluruh data latih yang telah ada. Hal ini dilakukan dengan tujuan agar data latih yang semula memiliki panjang dan tinggi tidak seimbang atau berbentuk persegi panjang (Gambar 6a) berubah menjadi seimbang atau berbentuk

persegi, sehingga pada saat dibutuhkan proses *scaling*, data latih tidak mengalami perubahan bentuk yang terlalu berlebihan (6b).



Gambar 6. Implementasi YOLO v2 pada data latih

Setelah implementasi YOLO v2 pada seluruh data latih selesai, langkah selanjutnya adalah melakukan konfigurasi pada setiap fungsi optimasi yang digunakan. Ada beberapa alternatif fungsi optimasi yang dapat digunakan untuk penelitian ini, antara lain Adam (Kingma, 2017), Adadelta (Zeiler, 2012), dan SGD (Ruder, 2016). Setelah melakukan eksperimen dengan menggunakan berbagai fungsi optimasi yang ada, dipilih fungsi optimasi Adam yang memberikan nilai akurasi yang paling optimal dibandingkan kedua fungsi optimasi lain.

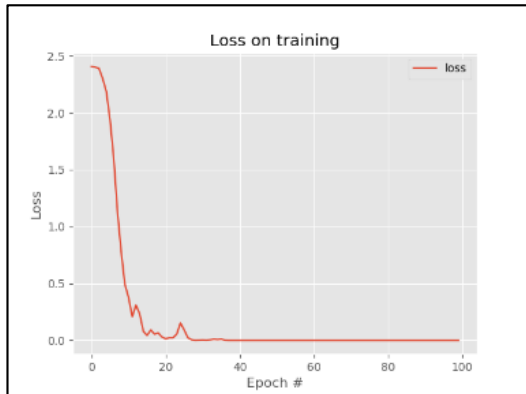
Kemudian untuk *epoch* yang digunakan adalah 100 *epoch* pada masing-masing fungsi optimasi, untuk *batch size* yang digunakan adalah 64, dan *learning rate* yang dipakai adalah 0.001 untuk fungsi optimasi Adam.

Pada proses training, perubahan *loss* dari *epoch* pertama menunjukkan angka 2.4 dan menurun secara cepat sampai *epoch* ke 18, kemudian pada *epoch* ke 24 sampai ke 100 mengalami kestabilan yang baik dengan nilai akhir *loss* 0. Untuk grafik *loss* pada *training* model MiniVGGNet menggunakan fungsi optimasi Adam dengan masukan gambar berukuran 64x64 piksel dapat dilihat pada Gambar 7. Untuk waktu yang dibutuhkan dalam proses *training* adalah 3 jam untuk masukan gambar dengan ukuran 64x64 piksel.

3.2. Pengujian YOLO

Pengujian deteksi mobil pada YOLO v2 dilakukan dengan menggunakan sensitifitas *threshold* 0.01. Dimana apabila nilai dari *threshold* semakin besar, maka akan semakin sulit untuk mendeteksi adanya suatu objek pada suatu citra. Pada saat proses *training* dari model YOLO v2, model dapat mendapatkan hasil akurasi *training* yang bagus yaitu dengan hasil akurasi akhir 99.58% atau *loss* 0.42 pada iterasi yang ke 21.000. Dari 110 data uji yang digunakan untuk pengujian YOLO v2, hasil akurasi dari deteksi objek adalah 103 terdeteksi dari 110 buah data uji (tabel 3). Waktu yang

dibutuhkan untuk melakukan deteksi adalah 12 detik.



Gambar 7. Grafik loss training model MiniVGGNet dengan optimasi Adam dan input citra 64x64 piksel

Tabel 2. Hasil pengujian YOLO v2 untuk deteksi mobil

Objek Deteksi	Hasil Deteksi	
	Terdeteksi	Tidak Terdeteksi
Mobil	103	7

3.3. Pengujian MiniVGGNet

Untuk pengujian dari model MiniVGGNet digunakan metode *true positive rate* dengan menggunakan *matrix confusion*. Pengujian model pada MiniVGGNet diawali dengan model dengan masukan gambar 64x64 piksel. Waktu yang dibutuhkan untuk mengklasifikasikan gambar adalah 5 detik. *Matrix confusion* pada model MiniVGGNet menggunakan fungsi optimasi Adam dengan masukan gambar berukuran 64x64 piksel dapat dilihat pada Tabel 3.

Tabel 3. Hasil pendeteksian jenis mobil menggunakan MiniVGGNet Adam dengan input 64x64 piksel

Jenis	Hasil Deteksi											Total
	A	B	C	D	E	F	G	H	I	J	K	
A	7	0	0	0	1	1	0	0	0	0	0	9
B	0	7	0	1	1	0	1	0	0	0	0	10
C	0	1	4	0	0	2	0	1	0	1	1	10
D	0	0	0	2	0	0	0	1	3	1	0	7
E	0	0	0	0	6	2	0	0	0	1	0	9
F	0	0	2	0	2	4	0	0	0	0	1	9
G	0	2	0	0	0	0	6	2	0	0	0	10
H	1	2	1	2	2	0	0	2	0	0	0	10
I	0	0	0	0	0	0	1	1	8	0	0	10
J	0	0	0	0	0	3	0	0	1	3	2	9
K	0	0	0	0	0	0	0	1	0	1	8	10

Di mana kolom dan baris A-K mewakili jenis mobil, yaitu: Toyota Avanza (A), Honda Jazz (B), Honda Brio (C), Toyota Innova (D), Nissan Juke

(E), Suzuki Swift (F), Daihatsu Xenia (G), Mitsubishi Pajero (H), Toyota Alphard (I), Daihatsu Grandmax (J), dan Toyota Yaris (K).

Untuk hasil perhitungan dari Sensitifitas, Spesifisitas, dan Akurasi dilihat pada Tabel 4.

Tabel 4. Hasil uji sensitivitas (ss), spesifisitas (Sp) dan akurasi (ak)

Obj	Hasil Deteksi										
	A	B	C	D	E	F	G	H	I	J	K
ss (%)	78	7	4	2	6	4	6	8	8	3	8
sp (%)	100	9	9	9	9	9	9	9	9	9	9
ak (%)	60										

Berdasarkan tabel 4, akurasi keseluruhan yang didapat adalah 60%. Pada model ini terdapat 7 merek mobil yang mendapatkan nilai sensitifitas dan spesifisitas yang baik, dan 4 merek mobil mendapatkan nilai sensitifitas yang cukup rendah tetapi memiliki nilai spesifisitas yang tinggi. Untuk merek mobil yang mendapatkan hasil cukup baik antara lain Toyota Yaris (K), Toyota Alphard (I), Mitsubishi Pajero (H), Toyota Avanza (A), Honda Jazz (B), Nissan Juke (E), dan Daihatsu Xenia (G).

Untuk merek mobil yang mendapatkan hasil kurang baik antara lain Suzuki Swift (F), Honda Brio (C), Daihatsu Grandmax (J), dan Toyota Innova (D).

Hasil akurasi yang kurang baik dapat disebabkan oleh beberapa hal. Dalam penelitian ini yang menyebabkan kurang baiknya akurasi yang didapatkan dikarenakan jumlah dari data latih yang terbatas yaitu hanya sejumlah 1.100 gambar atau 100 gambar untuk setiap merek dari mobil. Selain itu sudut pandang dari data latih kurang sesuai dengan sudut pandang yang terdapat pada data uji. Pada data latih sudut pandang pengambilan mobil terletak tepat pada posisi depan mobil, sedangkan pada data uji sudut pandang ada yang kurang tepat pada posisi depan mobil. Hal ini juga dapat menyebabkan berkurangnya akurasi yang didapatkan dari penelitian ini.

Hal lain yang dapat menyebabkan berkurangnya akurasi adalah kemiripan bentuk atau ciri khas dari setiap merek yang diujikan (Gambar 8), Secara penglihatan normal pengenalan merek kendaraan bisa jadi mudah, tetapi dalam *machine learning* membedakan merek kendaraan yang memiliki ciri khas yang mirip memerlukan banyak pelatihan.



Gambar 8. Kemiripan objek yang akan diklasifikasikan

4. KESIMPULAN

Berdasarkan sistem deteksi dan klasifikasi untuk penentuan iklan pada *billboard* yang sudah dibuat dan diuji, didapatkan kesimpulan:

1. Model YOLO v2 yang dilatih dapat digunakan untuk mendeteksi objek mobil baik pada gambar maupun *video* dengan akurasi akhir sebesar 93%.
2. Fungsi optimasi yang terbaik untuk digunakan pada model MiniVGGNet pada penelitian ini adalah fungsi optimasi Adam dengan masukan gambar berukuran 64x64 piksel yang menghasilkan akurasi sebesar 60%.
3. Penentuan iklan *billboard* menggunakan sistem deteksi dan klasifikasi secara *realtime* memungkinkan untuk dilakukan apabila perangkat keras yang digunakan lebih memadai untuk proses deteksi dan klasifikasi.

DAFTAR PUSTAKA

- AMALUDDIN, F., MUSLIM, M.A. AND NABA, A., 2015. Klasifikasi Kendaraan Menggunakan Gaussian Mixture Model (GMM) dan Fuzzy Cluster K Means (FCM). *Jurnal EECCIS*, 9(1), pp.19-24.
- HERYADI, Y., BUDIHARTO, W. AND ABDURACHMAN, E., 2019, April. Logo and Brand Recognition from Imbalanced Dataset Using MiniGoogLeNet and MiniVGGNet Models. In *Asian Conference on Intelligent Information and Database Systems*(pp. 385-393). Springer, Cham.
- GEZAHEGN, Y.G., GEBRESLASSIE, A.K., HAGOS, M.A., IBENTHAL, A. AND ETSUB, E.A., 2019, May. Classical Machine Learning Algorithms and Shallower Convolutional Neural Networks Towards Computationally Efficient and Accurate Classification of Malaria Parasites. In *International Conference on Information and Communication Technology for Development for Africa* (pp. 46-56). Springer, Cham.
- IRFAN, M., SUMBODO, B.A.A. AND CANDRADEWI, I., 2017. Sistem Klasifikasi Kendaraan Berbasis Pengolahan Citra Digital dengan Metode Multilayer Perceptron. *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, 7(2), pp.139-148.
- KINGMA, D.P. & BA, J.L. 2017, Adam: A Method for Stochastic Optimization, ArXiv, 1412:6980.
- KWON, O., LEE, H.G., LEE, M.R., JANG, S., YANG, S.Y., PARK, S.Y., CHOI, I.G. AND YEO, H., 2017. Automatic Wood Species Identification of Korean Softwood Based on Convolutional Neural Networks. *목재공학 (Journal of the Korean Wood Science and Technology)*, 45(6), pp.797-808.
- REDMON, J, DIVVALA, S, GIRSHICK, R, FARHADI, A 2015, You Only Look Once: Unified, Real-Time Object Detection, Diakses tanggal 16 Februari 2019, <<https://pjreddie.com/media/files/papers/yolo.pdf>>
- ROHADI, E., CHRISTINE, A.M., PRASETYO, A., ASMARA, R.A., SIRADJUDDIN, I., RONILAYA, F. AND SETIAWAN, A., 2018. Implementasi Video Streaming Lalu Lintas Kendaraan Dengan Server Raspberry Pi Menggunakan Protokol H. 264. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 5(5), pp.629-634.
- Ruder, S., 2016. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.
- SETYAWAN, G.E., ADIWIJAYA, B. AND FITRIYAH, H., 2019. Sistem Deteksi Jumlah, Jenis dan Kecepatan Kendaraan Menggunakan Analisa Blob Berbasis Raspberry Pi. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 6(2), pp.211-218.
- SIMONYAN, K. & ZISSERMAN, A. 2015, Very Deep Convolutional Neural Network for Large-Scale Image Recognition, *ArXiv*, 1409:1556
- SWASTIKA, W., ARIYANTO, M.F., SETIAWAN, H. AND IRAWAN, P.L.T., 2019, March. Appropriate CNN Architecture and Optimizer for Vehicle Type Classification System on the Toll Road. In *Journal of Physics: Conference Series* (Vol. 1196, No. 1, p. 012044). IOP Publishing.
- ZEILER, M.D., 2012. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Halaman ini sengaja dikosongkan