UNIVERSITY POLITEHNICA OF BUCHAREST

Faculty of Automatic Control and Computer Science

Ph.D. Thesis

David-Traian IANCU

DETECTION, SEGMENTATION AND PREDICTION IN AUTONOMOUS DRIVING

THESIS COMMITTEE

Prof. Dr. Eng. Florin Pop University Politehnica of Bucharest	Chairman
Prof. Dr. Eng. Adina Magda Florea University Politehnica of Bucharest	Thesis Advisor
Prof. Dr. Eng. Vasile Manta Technical University Gheorghe Asachi Iasi	Member
Prof. Dr. Eng. Viorel Negru West University of Timișoara	Member
Prof. Dr. Eng. Mariana Mocanu University Politehnica of Bucharest	Member

BUCHAREST 2022

Acknowledgements

I would like to thank in the first place to my thesis coordinator, who made this research possible, by proposing the research subject and by offering me the necessary guidance throughout all this years. I would also want to thank to the University Politehnica of Bucharest, which offered the infrastructure required in order to make some of the experiments presented in this thesis and also for offering a place to work for the research. I would like to thank to the team from the AIMAS Laboratory, which contributed in some of the research papers published for this research and helped me with the cameras and the datasets. The current thesis was possible due to the grants obtained by the university in the associated research projects.

Abstract

Autonomous driving has become one of the most important challenges regarding today's research in computer vision and artificial intelligence. In its development the private companies benefited from the help of the research community in order to develop better algorithms. An autonomous car will have obvious advantages in the daily life but for the moment there is still much to do in order to develop a fully autonomous functional and safe car. A typical autonomous car have a lot of components regarding scene understanding, decision making and vehicle control. This thesis studies in detail the scene understanding regarding the autonomous cars and focuses on some of the most relevant tasks regarding the scene understanding: object detection, object tracking, semantic and instance segmentation and depth estimation. At the limit between the scene understanding and decision making is the task of trajectory prediction of the surrounding cars, which is based on scene understanding but it is useful on the decision making process. The purpose of the thesis is to analyze the scene understanding tasks and use them to design a new trajectory prediction algorithm. The novelty is that the trajectory prediction task is made using video generation - an approach never met in the literature.

The thesis analyzes the tasks of object detection and tracking, semantic and instance segmentation, depth estimation and trajectory prediction, making a comprehensive analysis of the most important works and datasets at the moment. For each task, some of the best existing architectures were tested. Also, for each of these tasks multiple experiments were made on new datasets recorded in the campus of University Politehnica of Bucharest, taking into account multiple parameters like the size of the cars or the time of the day, in order to see which of the tested architectures works best in a real life scenario. The final purpose is to find the best architectures that can be combined for the trajectory prediction task using video generation. Finally, for the trajectory prediction task, the thesis proposes a new method based on object detection, road segmentation, depth estimation and video generation. Also, it proposes three new video generation architecture variations with better results for the trajectory prediction task. The biggest advantage is that even if the video generation task is more complex, it eliminates the need of a manually annotated trajectory, which can be a very laborious task. Instead, a video generation algorithm could be trained with any possible driving video, which could lead to better trajectory predictions in the future, with enough training data.

Table of contents

Chapter 1

Introduction

Autonomous driving has been one of the most challenging tasks in the latest years in both industry and academia and it has been in the minds of both researchers and car manufacturers in the last decade. The advantages of an autonomous car are obvious, beginning from safety and financial reasons and finishing with the people comfort. If the safety is discussed, the automation of the cars will lead to a better world if all the cars will incorporate almost perfect systems that will not make mistakes and will respect the traffic rules. Most of the accidents nowadays are made because the drivers engage in dangerous overtakes, they don't respect the traffic lights or the traffic rules, so a perfect system will overpass these aspects. Only a small percent of the accidents are now happening due to car problems or to the weather, so if a perfect algorithm will be designed for autonomous driving, the number of accidents will be substantially lower. Regarding the safety, according to a study, over 94% of the accidents are made by human mistake, the rest of the accidents being related to the failure of the car, the road, weather, or even unknown reasons. Not only that autonomous driving will lead to a safer environment, but also a better and cheaper one - people could share cars, even taxis, in order to go to their destination, being more cheap and also more ecological. Less cars on the street will lead to faster times to reach the destination. However, currently there is no perfect autonomous car. An autonomous car involves a lot of components and a lot of cutting edge algorithms, combining computer vision, artificial intelligence, machine learning and data science. This thesis aims to tackle some of the most important parts regarding an autonomous car and to propose a new trajectory prediction system for an autonomous car, based on video generation, object detection, semantic segmentation and depth prediction.

1.1 Autonomous driving

An autonomous car consists of a lot of different components. There is an acquisition layer, with some sensors (cameras, GPS sensor, LIDAR, RADAR, IMU, etc). After that, the car system has a perception layer, where the car recognizes the environment, its position, the surrounding vehicles, including the tracking of those, the road, the distance to the surrounding vehicles, etc. After the perception layer, there is a decision layer, which incorporates data from the sensors or even from the other cars (in a scenario where there are many autonomous cars on the road).In the decision layer, there are many components – local and global route planning, behavioral planning (the maneuver that must be made, for example crossing a lane, also observing what the other participants do), and motion planning/ path following, to follow the desired action. After all this layers, the final layer commands the car – given a steering angle and a acceleration or brake percentage, the actuators must make those commands.

When autonomous driving is discussed and analyzed, there are 5 levels of automation that were standardized only a few years ago. The first level contains the cars which can maintain the same speed by themselves or even make an emergency braking. However, the human driver still controls the car. The second level implies that the software will take the control over the car, without no intervention from the driver, but the driver must be very careful in order to take back the control of the car. The third level would imply that the driver should not necessarily watch the road, but it would have to drive the car when needed (in forbidden areas, for example). The fourth level will imply that the human will not need to drive at all, but this facility could be allowed only in certain spaces or circumstances. The fifth and last level will imply that there is no need for a steering wheel, because the software controls everything. However, as of 2022, only Honda (functional only on highway) and Mercedes have launched a level three autonomous car, with another few companies waiting to launch cars with a similar degree of automation. Unfortunately, the last two levels will not be seen soon in the market, which gives the researchers the responsibility of developing better algorithms, in order to achieve a fully autonomous car. The research area regarding the autonomous driving is far from being outdated, with new architectures being designed each month.

1.2 Scope of the doctoral thesis

The purpose of this research is to analyze the core concepts regarding autonomous driving - object detection and tracking, semantic and instance segmentation, depth estimation and trajectory prediction. Also, for the trajectory prediction the domain of video generation is analyzed, which is related to the autonomous driving research. The thesis is focused on four specific tasks - object detection, semantic segmentation (especially for the road semantic segmentation), depth estimation and trajectory prediction. For each task there are described the most relevant works in the field and also the studies that have been made regarding the field itself, and also for related areas, like video generation. This research is also relevant for comparing the state of the art studies regarding these topics, and other review articles and studies are described, in order to show what this thesis brings new regarding the current reviews. For each task the existing datasets are analyzed and also new datasets are made, recorded and annotated manually by the team from Politehnica University and there are compared with the existing ones. The datasets were recorded in the university campus and took in account factors like the time of the day and the size of the cars. The dataset construction is also another important contribution regarding this thesis, by bringing new datasets to the research community. For each task some experiments are made using the best networks available at the moment. The experiments were made with the images from the recorded datasets and different statistics were made regarding the quality of the results, the inference time and the possibility of using the algorithms in a real time application. For the final task, the trajectory prediction, many of the previous results were included in order to make a new trajectory prediction algorithm, which takes into account the semantic segmentation, the depth prediction the object detection and also has at its core the idea of video generation - an approach that has not been seen previously in the research literature. Also, three new architecture variations are proposed by modifying a popular video generation network, some of them obtaining better results for the trajectory prediction task than the base model.

1.3 Content of the doctoral thesis

The thesis is structured in 8 chapters. Chapter 1 consists of the introduction and the presentation of the autonomous driving field and of the thesis. The rest of the thesis is structured as follows. In Chapter 2 some useful background information is detailed in order for a regular computer science engineer to understand the thesis - the concept of the neural networks is described, which is used in all the experiments made. The neural networks are new architectures that try to simulate the human brain and are now widely used for computer vision tasks. This is why a short presentation for the neural networks can be seen as mandatory, in order for this thesis to be better understood. In Chapter 3, the related work for each of the four tasks that is discussed in the thesis is analyzed - the object detection, the semantic segmentation, the trajectory prediction and depth estimation. Some brief information regarding object tracking and video generation is also included,which is the core concept of the trajectory prediction model. In this research a new model which predicts the trajectory by using a video generation model is presented, something which has not been tried previously in the research literature. Both the related studies are presented and analyzed and the reviews regarding the best architectures, in order to show what the current thesis bring new related to the other reviews. The next four chapters present the experiments regarding the studies task. In Chapter 4 are described the results considering the object detection task, in Chapter 5 are described the results regarding the road semantic segmentation task, Chapter 6 consists in the description of the experiments made for the depth estimation task and, finally, Chapter 7 consists of the experiments made for the trajectory prediction task and also described the proposed workflow and presents three new modified architectures based on a popular video generation architecture. Each of these four chapters have a similar structure, containing information regarding the most relevant datasets and also regarding the proposed dataset, the experiments made, the metrics involved and also the presentation of the results. Chapter 7 also have another section regarding the new proposed architectures, with their results considering the same task and setup. Finally, the conclusion and the future work is discussed in Chapter 8.

Chapter 2

Background

In this thesis, most of the architectures presented are artificial neural networks (ANN). The neural networks have been developed in the 1960s, trying to make a simplified model of the human brain – there are some nodes called neurons which are connected between them - a very simple architecture copied by the functioning of the human brain. Even if there are many layers of neurons, only two are visible - the input layer and the output layer, and the rest of the layers, which called hidden layers from obvious reasons, and are used in order to compute the weights of the final layer. The connections between the neurons are labeled as edges. In the ANN model each neuron and each edge have, generally, a value, called weight. The weight will determine how important is that specific component regarding the final output. As the model learns by taking more data to model, the weights of specific neurons can change in value - they can increase or decrease. Even if the models are not that knew, the usage of ANNs in machine learning, computer vision and natural language processing has increased only in the last 20 years. Nowadays, many of the computer vision tasks, natural language processing tasks and also other machine learning tasks (for example path finding in robots, action recognition, etc) are made using almost exclusively neural networks. There are many types of neural networks.

2.1 Neural Networks

2.1.1 Feedforward neural networks

The most simple model, which will be briefly explained here, is the feedforward neural network. If it contains hidden layers, it is also called nultilayer perceptron (MLP).

This basic model can be further improved by taking data into batches and also by making different modifications regarding the changes for the weights and bias. There are different optimization algorithms like RMSProp, Momentum, Adam.

The feedforward neural network shows the basics of any neural network – the number of layers, the activation functions, the output function, the cost function, the optimization

method. These parameters can be varied and make an infinite number of artificial neural networks. Generally, there is no known method for obtaining the best network. The most important factor that lead to the growth of the ANNs and better results is the variation of the experiments. The current networks are generally inspired by previous ones – new layers are added and new changes to the structure are made as long as the results are better.

2.1.2 Recurrent neural networks

Besides the feedforward neural network, there are now other neural networks that are used. The feedforward neural network have only connections from a layer to the next one. The next step was to introduce connections between the same layer or even between a layer and the previous one. The most simple example is a fully recurrent neural network (RNN), where each neuron in the network is connected to every other neuron. The basic RNN only take into account the previous hidden states by concatenating the hidden states into a bigger vector. However, the most used recurrent neural networks are using Long Short-Term Memory (LSTM) units or Gated Recurrent Units (GRUs). The most common networks are using LSTMs, which have been able to tackle some problems that the feedforward neural networks have, for example the problem of the vanishing gradient or the exploding gradient, which is why the LSTMs are used for tasks that require time – for example the prediction of the trajectory of a car. The GRUs are a simplified model of the LSTM and are used more in natural language processing, but they also have their usage in computer vision.

2.1.3 Convolutional neural networks

Another type of neural networks are the convolutional neural networks (CNNs). This type of ANNs obtained the best results for tasks regarding images, due to their properties of manipulating the space. It can be stated that RNNs are the best when the time has to be taken into account and the CNNs are the best if the space has to be incorporated. Some moderns neural networks have both recurrent and convolutional components. A convolutional neural network is basically a feedforward neural network which have at least some layers that perform convolutions. The convolution is a special operation that basically reduce the dimension of the input space (for example, an image).

2.1.4 Variational autoencoders

Besides the RNNs and CNNs, there are also other types of artificial neural networks. Not an network itself, but a useful architecture, is the encoder decoder model. It is applied especially regarding recurrent neural networks and it consists of two parts – an encoder, which takes the input and transforms it into a state of a fixed dimension (a

multidimensional tensor), and a decoder, which will take the tensor and try to apply again the transformation. The better the decoder is means that the encoder succeeds in representing the relevant components of the input. The encoder decoder scheme is used in computer vision, natural language processing and other machine learning tasks.

One type of neural network which is very used in the latest years is the Variational Autoencoder (VAE). The Autoencoder itself is used for dimensionality reduction, by using the encoder decoder scheme. However, the variational autoencoder has another important usage - it tries to obtain good properties for the result of the encoder, which is called the latent space, in order to take samples from the latent space and use them for the video generation task, for example.

2.1.5 Generative adversarial networks

The last neural network presented in this section is the Generative Adversarial Network (GAN). This network is designed specifically for generation – either new images or text. In this thesis will be exploited the benefits of the GAN for video generation and video prediction. The idea behind the GAN is very simple – it consists of two different networks – a generator and a discriminator.

There are also other types of neural networks, for example the transformer networks, which are used in natural language processing, or self-organizing map (SOM), another network for dimensionality reduction. However, they usage in scene understanding and autonomous driving is still limited, which is why other architectures are no more detailed.

Chapter 3

Related work

In this chapter are described the state-of-the-art architectures regarding the most important topics for this research and also for autonomous driving - object detection, object tracking, semantic, instance and panoptic segmentation, trajectory prediction and video generation. For each of these topics the evolution of the architectures throughout history is presented, some categories regarding the approaches are made, the best architectures regarding their results in real life applications are mentioned and also some of the best review articles regarding the subject are discussed. Instead of just mentioning some of the best architectures used, this section can be seen as a review and a comparison of the best technologies regarding some of the most important topics in autonomous driving as regarding the year 2022 and should be considered as one of the important contributions of the thesis.

3.1 Related problems

3.1.1 State of the art detectors

The deep learning approaches can be divided in three classes, considering their historical appearance. There are detectors that operates in a two stage process and detectors that have only one stage. All of them use anchors to detect objects. The newest networks don't use anchors at all, so they can be grouped in another category, anchor free detectors.

3.1.2 Object tracking

The task of object tracking involves identifying the same object in different frames – after the object detection, the object tracking should link the same object between frames. An advantage of object tracking is that it can be inferred the position of an object (a person or a car) if it is known for sure that it appears into a certain frame, but the detection will fail, because of various reasons (occlusion, blur in the image cause by the motion,

variations in illumination, resolution, scale, etc).

3.1.3 Segmentation

In the following section, it will analyzed the state of the art regarding semantic and instance segmentation, and also some related studies that tackle the semantic segmentation problem, what it was presented and what is improved in this work, regarding the semantic segmentation and instance segmentation problem. The first discussion is about some of the most used architectures for object classification, that are used for the feature extraction in the semantic segmentation networks, then they are analyzed the instance segmentation task, where the objects are detected, each instance is identified and each pixel of the object is classified, but the background is not classified, and also the semantic segmentation, where each pixel of the image is classified, without taking in account individual instances. It will also be analyzed a new approach, panoptic segmentation, which combines the two methods. At the end, the related review articles will be analyzed and what this thesis brings new to the segmentation study.

3.1.4 Instance segmentation

The instance segmentation task is, generally, based on object detection.

3.1.5 Semantic segmentation

This subsection analyzes the semantic segmentation architectures. They are very important for autonomous driving because they can detect and classify background pixels, including the road, which is one of the most important tasks in autonomous driving, and also the subject of the current study. There are many architectures, most of them based on convolutional neural networks, with different architectures and optimizations.

3.1.6 Panoptic segmentation

In this subsection are described some architectures that do both the instance segmentation and the semantic segmentation task. This approach is called panoptic segmentation, and is a very recent way of doing the segmentation, and also it is useful for autonomous driving.

3.1.7 Depth estimation

In this section the most important depth estimation networks, as well as the most relevant review papers regarding depth networks are analyzed and compared to the current study. The first division regarding depth estimation studies can be made regarding the number of the cameras that are used. The stereo depth networks use two cameras and are better in accuracy but also need a more complex system so the monocular depth networks have their own advantages. In the experiments made there were used only monocular depth estimation networks, which have the advantage of being easier to train and test and also they need a cheaper infrastructure.

3.1.8 Trajectory prediction networks

In this section are described the most relevant networks regarding the related tasks for the experiments made for trajectory prediction. The most relevant networks use LSTM, RNN, GAN, LSTM-CNN or CNN architectures.

3.1.9 Video generation networks

The task of making frames that form a video can be divided into two different categories – there are networks that try to generate random frames that could be considered a real video (without any link with a real one) and also networks that try to predict new frames given an original video. The tasks of video generation and video prediction are, however, related and in many cases the two terms can be used in an interchangeable way. The most relevant networks use LSTM, RNN, GAN, LSTM-CNN or VAE architectures.

Chapter 4

Evaluating object detection for autonomous driving

In this chapter are described the most important results regarding object detection and its related task, object tracking. In the first section are described some classical datasets for the object detection task and also the dataset used in the current experiments. In the next section are described the experiments and the metrics used for the object detection task. In the last section the results are described and analyzed.

4.1 POLI dataset - collecting a new dataset for object detection

Some of the images from the POLI dataset can be found in Figure 4.1.

Fig. 4.1 POLI dataset images

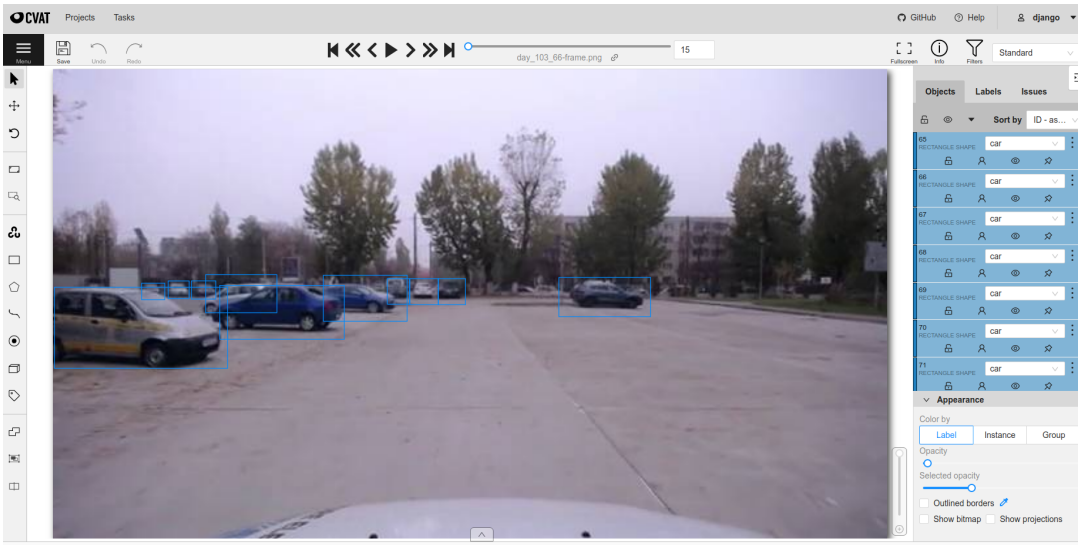

Fig. 4.2 CVAT annotation tool

The dataset was recorded on the campus streets, passing many cars and students. The dataset was manually labeled using an online tool, CVAT , which simplifies the annotation task, by interpolating the bounding boxes from the frames between two annotated frames, then the resulting boxes can be manually adjusted to perfectly match the ground truth. The annotation is one of the most time consuming tasks regarding computer vision, with a lot of manual work that has to be done in order for the bounding boxes to be as close as possible to the ideal one. A screenshot with the capabilities of the CVAT tool can be seen in Figure 4.2.

The dataset consists of 13001 images, containing 60227 objects, from which 90% are either car or person (41064 objects are cars, 14576 are persons). There are also traffic signs and bicycles in the dataset. The dataset is a hard one, because there are frames with many cars annotated in a parking spot where is also seen the crowded road that is in front of the university, with multiple cars crossing every second. The state-of-the-art detectors had a hard time regarding this dataset, as it can be seen in Chapter 5, the recall being smaller than a classical, less crowded dataset.

4.2 Object detection experiments and results

In this section are analyzed the experiments and the metrics made for the object detection task. To summarize, there were used 4 networks – YOLO v3, RetinaNet, Faster R-CNN and SSD and tested it against the BDD100K dataset, in order to see their performances regarding recall and accuracy and how the results vary regarding the time of the day (daytime, dusk or dawn and night) and also a different statistic was made regarding the dimension of the objects. The same architectures were tested against the proposed dataset in the Politehnica University Campus, manually annotated. Also, the same statistics

Table 4.1 Precision on BDD100K Dataset

	DC	DO	DR	DS	DSC	DSO	DSR	DSS	NC	NO	NR	NS
YOLO AP@.50IOU	0.66	0.65	0.64	0.65	0.65	0.65	0.62	0.65	0.63	0.64	0.61	0.64
SSD AP@.50IOU	0.92	0.93	0.92	0.92	0.94	0.93	0.93	0.93	0.90	0.93	0.90	0.91
Faster R-CNN AP@.50IOU	0.84	0.86	0.86	0.88	0.86	0.86	0.87	0.89	0.82	0.86	0.82	0.83
RetinaNet AP@.50IOU	0.28	0.27	0.32	0.32	0.27	0.26	0.29	0.30	0.31	0.33	0.32	0.34
YOLO MAP	0.56	0.55	0.55	0.55	0.55	0.55	0.53	0.55	0.54	0.55	0.53	0.55
SSD MAP	0.79	0.80	0.79	0.79	0.80	0.79	0.79	0.80	0.74	0.79	0.75	0.75
Faster R-CNN MAP	0.67	0.69	0.68	0.70	0.68	0.68	0.68	0.71	0.64	0.68	0.63	0.65
RetinaNet MAP	0.34	0.34	0.37	0.36	0.34	0.33	0.35	0.36	0.37	0.39	0.37	0.38

Table 4.2 Recall on BDD100K Dataset

	DC	DO	DR	DS	DSC	DSO	DSR	DSS	NC	NO	NR	NS
YOLO AR@.50IOU	0.36	0.37	0.36	0.36	0.35	0.37	0.34	0.35	0.20	0.33	0.18	0.22
SSD AR@.50IOU	0.17	0.17	0.19	0.19	0.16	0.17	0.18	0.19	0.12	0.16	0.11	0.14
Faster R-CNN AR@.50IOU	0.14	0.14	0.13	0.14	0.12	0.13	0.12	0.13	0.05	0.11	0.04	0.06
RetinaNet AR@.50IOU	0.09	0.08	0.10	0.10	0.08	0.08	0.09	0.09	0.06	0.09	0.06	0.07
YOLO MAR	0.30	0.31	0.30	0.30	0.29	0.31	0.29	0.30	0.17	0.28	0.15	0.19
SSD MAR	0.14	0.15	0.16	0.16	0.14	0.14	0.15	0.16	0.10	0.14	0.09	0.11
Faster R-CNN MAR	0.11	0.11	0.10	0.11	0.09	0.10	0.09	0.10	0.04	0.09	0.03	0.05
RetinaNet MAR	0.11	0.11	0.11	0.12	0.10	0.10	0.10	0.11	0.07	0.11	0.07	0.08

were made, in order to see which of the networks have better results and also how the results will adapt from a large dataset to a smaller, unknown one, that was not used for fine tuning. This was made in order to see the real performances of the architectures, without having to deal with overfitting.

4.2.1 Object detection results

The results for the precision can be seen in Table 4.1 and Table 4.3 (only for car and person) and the results for the recall can be seen in Table 4.2 and Table 4.4 (only for car and person). The results for the POLI dataset are shown in Table 4.5. Some plots regarding the precision and recall regarding the car size can be seen in Figure 4.3 and in Figure 4.4.

Table 4.3 Precision on BDD100K Dataset - only car and person

	DC	DO	DR	DS	DSC	DSO	DSR	DSS	NC	NO	NR	NS
YOLO AP@.50IOU	0.72	0.72	0.74	0.72	0.72	0.72	0.73	0.73	0.72	0.74	0.73	0.76
SSD AP@.50IOU	0.92	0.93	0.92	0.92	0.94	0.93	0.93	0.93	0.90	0.93	0.90	0.91
Faster R-CNN AP@.50IOU	0.87	0.87	0.87	0.88	0.88	0.87	0.89	0.89	0.85	0.88	0.85	0.85
RetinaNet AP@.50IOU	0.28	0.27	0.32	0.32	0.27	0.26	0.29	0.30	0.31	0.33	0.32	0.34
YOLO MAP	0.60	0.60	0.62	0.60	0.60	0.61	0.61	0.61	0.60	0.62	0.60	0.62
SSD MAP	0.79	0.80	0.79	0.79	0.80	0.79	0.79	0.80	0.74	0.79	0.75	0.75
Faster R-CNN MAP	0.69	0.70	0.69	0.70	0.70	0.69	0.69	0.71	0.66	0.70	0.66	0.67
RetinaNet MAP	0.34	0.34	0.37	0.36	0.34	0.33	0.35	0.36	0.37	0.39	0.37	0.38

Table 4.4 Recall on BDD100K Dataset - only car and person

	DC	DO	DR	DS	DSC	DSO	DSR	DSS	NC	NO	NR	NS
YOLO AR@.50IOU	0.48	0.50	0.50	0.50	0.47	0.50	0.47	0.50	0.30	0.46	0.27	0.34
SSD AR@.50IOU	0.17	0.17	0.19	0.19	0.16	0.17	0.18	0.19	0.12	0.16	0.11	0.14
Faster R-CNN AR@.50IOU	0.20	0.21	0.20	0.21	0.17	0.19	0.17	0.21	0.09	0.16	0.08	0.10
RetinaNet AR@.50IOU	0.09	0.08	0.10	0.10	0.08	0.08	0.09	0.09	0.06	0.09	0.06	0.07
YOLO MAR	0.40	0.43	0.42	0.42	0.39	0.42	0.39	0.42	0.25	0.38	0.22	0.28
SSD MAR	0.14	0.15	0.16	0.16	0.14	0.14	0.15	0.16	0.10	0.14	0.09	0.11
Faster R-CNN MAR	0.16	0.17	0.16	0.17	0.14	0.15	0.14	0.17	0.07	0.13	0.06	0.08
RetinaNet MAR	0.11	0.11	0.11	0.12	0.10	0.10	0.10	0.11	0.07	0.11	0.07	0.08

Table 4.5 POLI results

	AP@0.50IOU	MAP	AR@0.50IOU	MAR
YOLO	0.69	0.55	0.59	0.47
SSD	0.79	0.65	0.12	0.10
Faster R-CNN	0.68	0.54	0.16	0.13
RetinaNet	0.17	0.26	0.06	0.09
YOLO (car and person)	0.71	0.57	0.62	0.50
SSD (car and person)	0.90	0.74	0.13	0.10
Faster R-CNN (car and person)	0.80	0.63	0.17	0.13
RetinaNet (car and person)	0.20	0.30	0.06	0.10

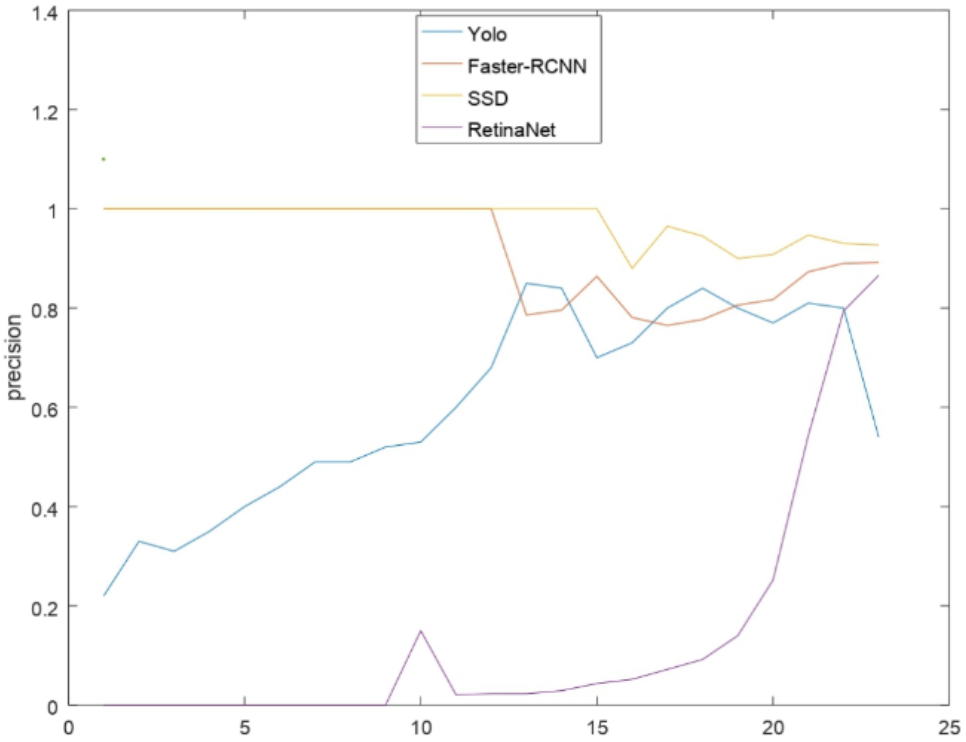

Fig. 4.3 Precision regarding object size

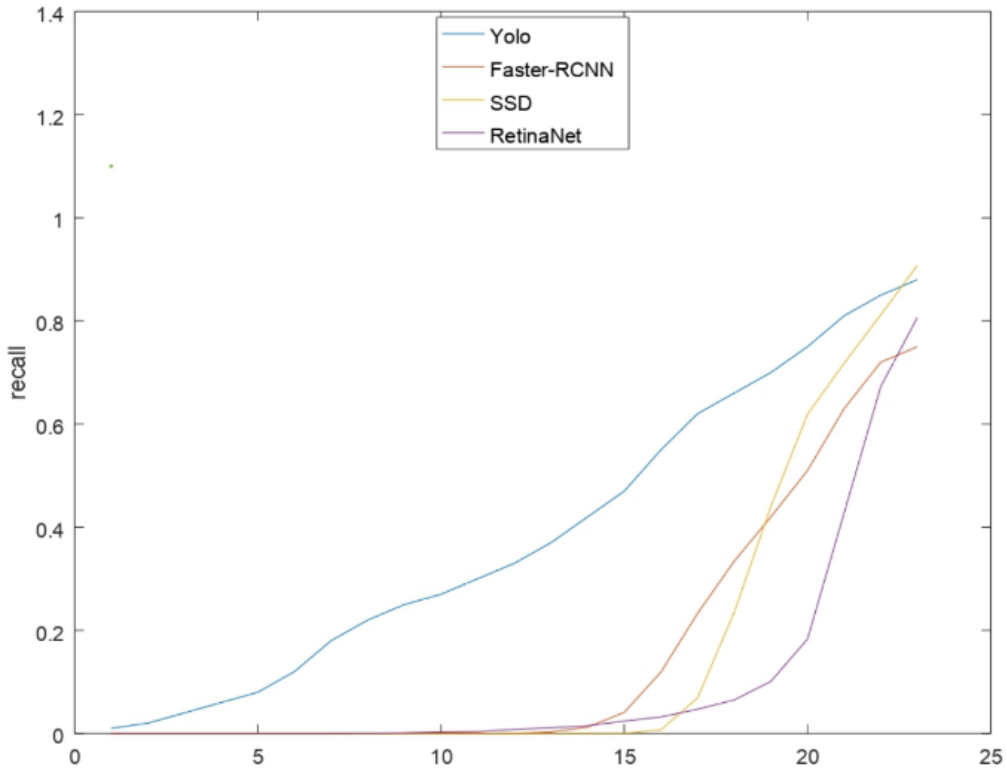

Fig. 4.4 Recall regarding object size

Chapter 5

Evaluating semantic segmentation for autonomous driving

In this chapter are presented the experiments made regarding the semantic segmentation task, with the focus on the road semantic segmentation. The first section of this chapter describes some of the most relevant segmentation datasets used in the literature, but also the POLI segmentation dataset, the dataset used for the current experiments. The next section describes the most important experiments made and also the metrics used for evaluating the quality of the experiments. The last section shows the results and makes an interpretation regarding the presented data.

5.1 POLI segmentation dataset - collecting a new dataset for road semantic segmentation

For the segmentation task, 138 movies were recorded in Politehnica university campus in different scenarios – during the day, during the night and also at sunrise or sunset, which was labeled as dusk, similar to the BDD100k annotations. Some of the images were recorded in the parking area, in front of the building of Automatic and Control Science, with a view to the moving cars on the very crowned boulevard near the faculty, with many cars that could be possibly detected. In Figure 5.1 can be seen the parking area in different light settings - the first two during the day, the third during the dusk and the last during the night.

The images are taken from the streets inside the Univesity Politehnica of Bucharest campus. The recordings were in clear weather (no rain or snow). Each movie contains some frames, totaling about 20.000 manually annotated frames, containing the segmentation for the road. The frames were manually annotated the road using CVAT, an online tool for ground truth generation, which allows the users to do the segmentation by making very complex polygons. The annotation work was very complex, considering that the annotation of the road is not as simple as the annotation of an object, which can

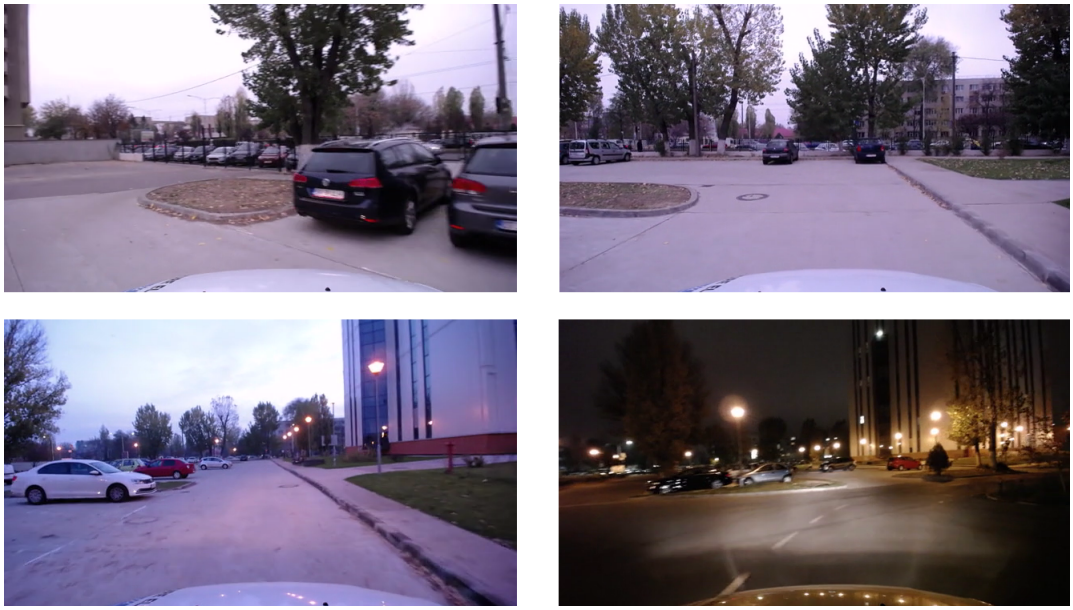

Fig. 5.1 POLI dataset - the parking area

be drawn only using two points (up left and bottom right). The segmentation of the road involves a lot of points, considering that the road will be estimated as a polygon with multiple sides. This is especially difficult when dealing with round shapes, which will require a lot of points in order to simulate the circular form. An example of the final segmentation of the road in all the daylight conditions can be seen in Figure 5.2. Also, an example of the annotation of the road using CVAT can be seen in Figure 5.3.

For the experiments not all the images were used because the segmentation time was quite big for some networks. Instead, only one frame from 20 was selected in order to have a good representative set and also to diminish the inference time for the data. Regarding the type of the pictures, there are 735 images taken in the day, 133 in the dusk and 165 in the night.

5.2 Road semantic segmentation experiments and results

In this section are described the experiments and the metrics made regarding the semantic segmentation. Detecting the road is important from obvious reasons – to avoid crossing the road limits and to keep the good direction. The road lanes could be an important tool, also, but not all the roads have it, so a good autonomous driving system should be able to use only the information regarding the position of the road. The experiments consists of detecting the road in the proposed dataset using some of the best existing networks for segmentation and comparing the results by taking into acount the time of the day.

Fig. 5.2 POLI segmentation dataset examples

5.2.1 Semantic segmentation results

In this subsection are presented the results regarding road semantic segmentation. For each of the network that was used there are experiments and results for the proposed dataset regarding the TP, FP, accuracy and IoU in the day, dusk, night and also an average for the whole dataset. For the autonomous driving scenario, the most important metric is the IoU. The output of the FCN architecture on some frames and also the manually annotated road can be seen in Figure 5.5. The results for the all the tested networks can be seen in Table 5.1. Also, the segmentation time can be seen in 5.4.

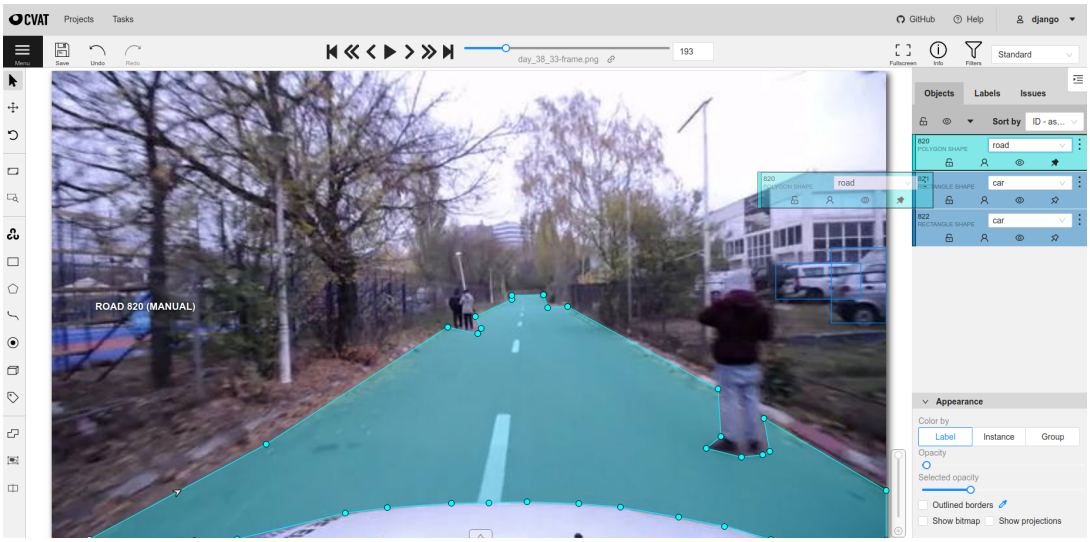

Fig. 5.3 Segmentation of the road using CVAT

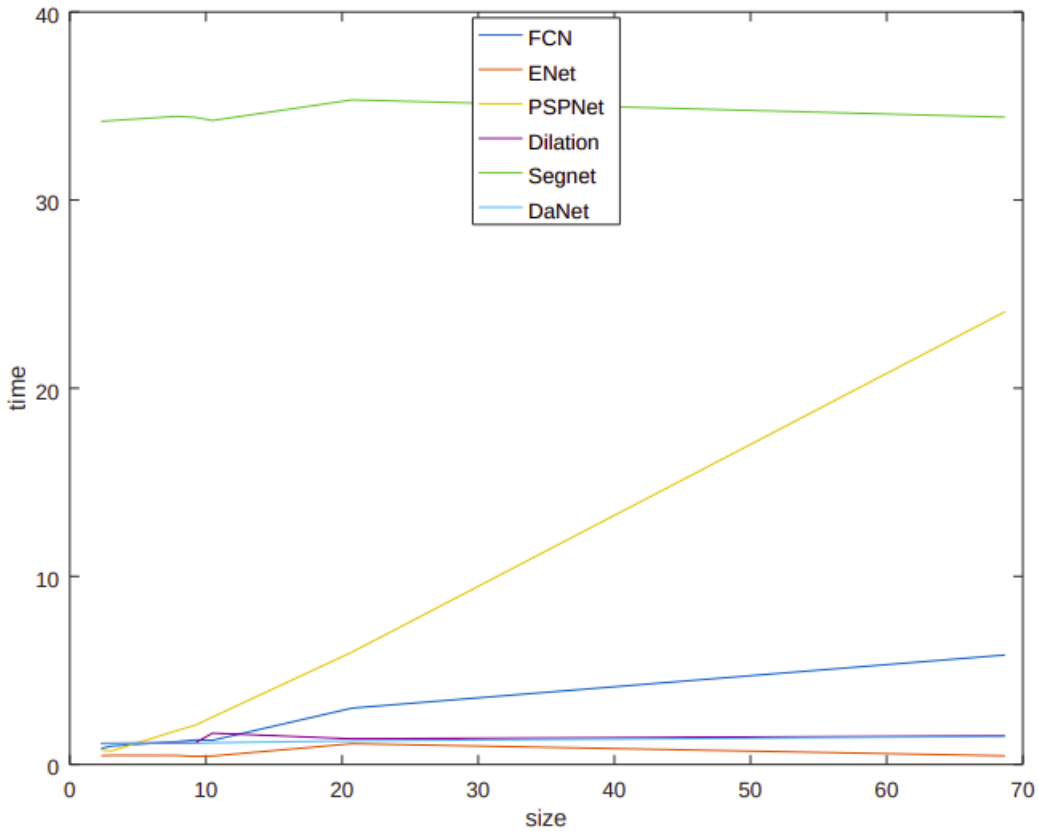

Fig. 5.4 Segmentation time

Table 5.1 Road Segmentation Results

	Day	Dusk	Night	Avg
	FCN			
TP	0.782	0.742	0.572	0.743
FP	0.057	0.051	0.070	0.058
Acc	0.897	0.893	0.835	0.887
IoU	0.673	0.647	0.476	0.638
	ENET			
TP	0.781	0.725	0.711	0.763
FP	0.218	0.448	0.320	0.264
Acc	0.780	0.597	0.686	0.741
IoU	0.527	0.352	0.393	0.483
	PSPNet			
TP	0.940	0.947	0.836	0.924
FP	0.048	0.058	0.089	0.056
Acc	0.948	0.943	0.889	0.938
IoU	0.831	0.820	0.676	0.805
	Dilation			
TP	0.508	0.431	0.958	0.571
FP	0.248	0.176	0.941	0.350
Acc	0.687	0.718	0.299	0.629
IoU	0.338	0.313	0.267	0.324
	SegNet			
TP	0.955	0.903	0.454	0.868
FP	0.157	0.233	0.028	0.146
Acc	0.872	0.803	0.829	0.856
IoU	0.673	0.558	0.414	0.617
	Danet 512			
TP	0.651	0.759	0.673	0.668
FP	0.442	0.604	0.312	0.442
Acc	0.584	0.497	0.681	0.589
IoU	0.307	0.294	0.367	0.315
	Danet 768			
TP	0.770	0.825	0.713	0.768
FP	0.574	0.744	0.325	0.556
Acc	0.518	0.413	0.684	0.532
IoU	0.303	0.274	0.382	0.312

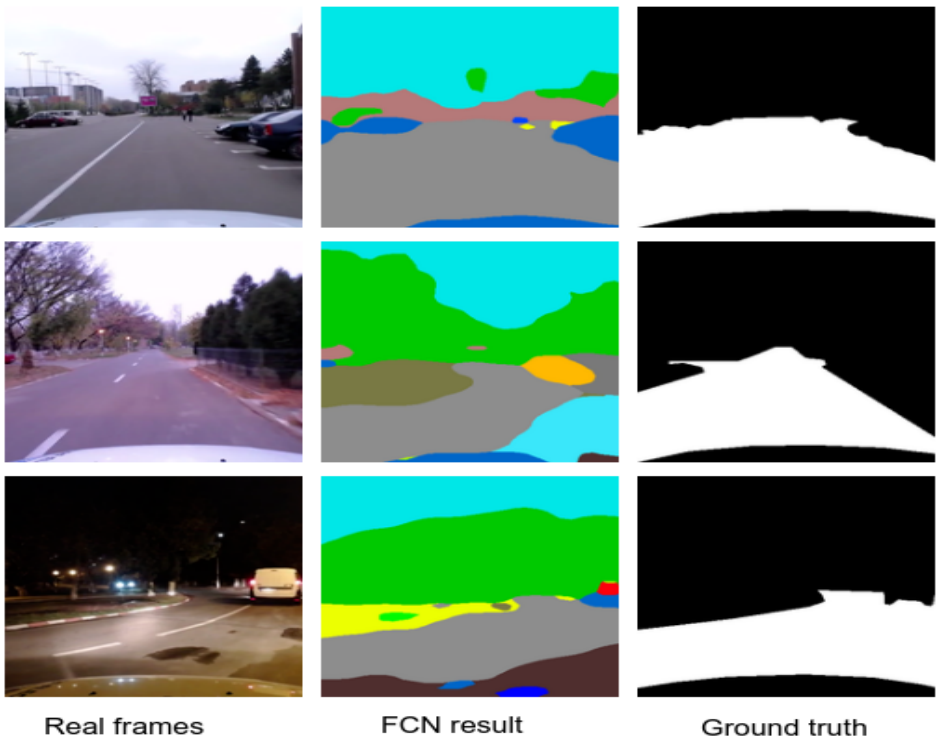

Fig. 5.5 FCN results and the corresponding ground truth

Chapter 6

Evaluating depth estimation for autonomous driving

This chapter proceeds to the analysis of the depth estimation task, which is crucial in analyzing the distance from the vehicle to the surrounding cars. Following the structure of the previous chapters, the first section analyzes the most important datasets for depth estimation and also the datasets used for the current experiments, manually recorded in the University Politehnica of Bucharest campus. The next section describes the experiments made for the depth estimation regarding the surrounding cars and also the metrics used for evaluating the quality of the results. The final section shows the results and also makes an interpretation of the results.

6.1 POLI depth dataset - collecting a new dataset for depth estimation

The depth dataset named POLI depth dataset is made for depth estimation and was recorded using specialized hardware for depth – an Intel RealSense RGB-D camera. Following the idea from the previous dataset made, the POLI depth dataset is divided in three different recording – images from the day, images from the dusk/ dawn and images from the night. This division helped to a better understanding regarding the depth estimation problems given different kinds of light settings.

The recording of the dataset had some challenges, because not always the depth map had the same size as the images. Sometimes some depth images were lost, which required multiple recordings and also an adjustment between the depth map frames and the real frames. Another problem was that the camera had lower accuracy during the night.

The RealSense camera is a D435 model, with frames recorded in HD quality – 1280x720 pixels. However, the depth map were only recorded at 848x480 resolution,

which required a further preprocessing step to match the sizes. Also, some of the depth estimation models required lower resolution for the images and the images had to be downsized. The recorded images were obtained with the camera mounted on the center of the windscreen of the car. Even if only one location have been taken into account for the recordings, having multiple light settings makes the POLI dataset a good dataset for testing.

One of the challenges regarding a depth dataset is the sensor used - even if the RealSense have really good results when testing during the day, the results during the night are far from being accurate, which added some noise into the results. However, the camera is almost twice as cheap as compared to the better version, D415, and a full LiDAR is much more expensive. Considering this financial reason, the D435 sensor was a good compromise and allowed for the experiments to be made. Some of the depth images that are in the POLI depth dataset can be seen in Figure 6.1.

The final dataset consists of 516 images recorded during the day, 1039 during the dusk/ dawn and another 637 images recorded during the night. The final dataset was obtained by sampling the recorded dataset in order to get different frames from different times, instead of having consecutive frames with similar structure.

Besides the depth dataset, for the experiments mad there was used the previous dataset, made for semantic segmentation (POLI segmentation dataset), without a corresponding ground truth, in order to see the relative performance of a model regarding the others. The absolute performance can be observed using the annotated dataset with depth, then the relative performance is measured on the other dataset.

6.2 Depth estimation experiments and results

In this section are described the experiments made with the reference models and with the proposed datasets regarding the depth estimation task. As it was mentioned earlier, two different datasets were used – one dataset is annotated with ground truth and recorded with an RGB-D camera, and the other one is the POLI segmentation dataset and does not have a ground truth for depth estimation. However, the ground truth is considered by taking the result of the best networks from the previous dataset. This helps to study the relative performance of the networks.

Fig. 6.1 POLI depth dataset

6.2.1 Depth estimation results

In this subsection all the results made in the current experiments are presented. The six models that were tested on two datasets are analyzed, all of them containing the RMSE for day, dusk, night and also the average RMSE. As it was mentioned, the first dataset contains also the ground truth recorded with the Intel RealSense RGB-D camera and the second one does not contain a ground truth, but for both datasets the ground truth varied as being one of the results of the best networks. The RMSE considering only the cars from the images was also measured, which is more relevant for the autonomous driving task. For the ground truth the results from the depth camera were considered for the first dataset but also the results from DenseDepth, Megadepth and Monodepth.Besides this the inference time was also measured regarding the image size and the RMSE regarding the object size. In Table 6.1 are shown the results for the first dataset, considering various

ground truth results. The results for the second set can be seen in Table 6.2.

The most relevant experiments can be seen in Table 6.3 and Table 6.4, where there are shown the results for the RMSE regarding only the car, which is more accurate for autonomous driving, where the purpose is to estimate the distance from the ego car to the surrounding vehicles. The cars were manually annotated and based on the annotations the RMSE was computed. In Table 6.3 are shown the results for the first set and in Table 6.4 are shown the results for the second set.

The final experiments are regarding the RMSE regarding the car size and the speed regarding the image size. The car sizes were divided in 14 categories and it was measured the RMSE for each size. In Figure 6.2 can be found the results for the RMSE regarding the car size, in Figure 6.3 can be seen the speed regarding the image size and, finally, in Figure 6.4 can be seen some prediction results for the depth estimation using the Monodepth network.

Table 6.1 Depth results - first dataset

Model	Day	Dusk	Night	Avg
Ground truth - depth camera				
Megadepth	128.12	140.99	139.38	137.59
DORN	72.51	98.46	55.39	82.00
LKVOLearner	98.36	109.63	97.29	103.56
SfMLearner	113.91	126.74	109.32	118.92
Monodepth	122.81	135.66	120.28	128.37
DenseDepth	82.96	83.85	87.65	84.77
Ground truth - DenseDepth				
Megadepth	105.37	109.40	90.15	103.19
DORN	90.96	93.09	85.26	90.38
LKVOLearner	80.78	79.69	64.99	75.98
SfMLearner	96.37	100.18	84.74	95.03
Monodepth	111.97	111.97	100.42	108.74
Ground truth - Megadepth				
DORN	125.66	130.95	139.30	132.23
LKVOLearner	47.89	54.03	57.44	53.69
SfMLearner	45.13	52.83	60.04	53.38
Monodepth	55.49	64.44	70.30	64.26
DenseDepth	105.37	109.40	90.15	103.19
Ground truth - Monodepth				
Megadepth	55.49	64.44	70.30	64.26
DORN	104.46	100.40	114.93	105.76
LKVOLearner	44.49	46.07	51.81	47.46
SfMLearner	42.98	42.74	47.87	44.35
DenseDepth	111.97	111.97	100.42	108.74

Table 6.2 Depth results - second dataset

Model	Day	Dusk	Night	Avg
Ground truth - DenseDepth				
Megadepth	52.72	59.24	44.25	52.27
DORN	60.60	67.36	62.29	61.75
LKVOLearner	51.14	58.69	43.84	51.01
SfMLearner	59.87	67.36	46.28	58.84
Monodepth	64.97	74.54	56.72	64.95
Ground truth - Megadepth				
DORN	20.12	21.49	25.10	52.27
LKVOLearner	12.13	13.45	13.38	61.75
SfMLearner	15.30	17.41	16.05	51.01
Monodepth	19.95	22.34	20.21	58.84
DenseDepth	52.72	59.24	44.25	64.95
Ground truth - Monodepth				
Megadepth	19.95	22.34	20.21	20.30
DORN	13.15	16.75	11.72	13.42
LKVOLearner	15.10	16.81	16.35	15.53
SfMLearner	7.53	9.08	15.33	9.45
DenseDepth	64.97	74.5	56.72	64.95

Table 6.3 Depth results - first dataset (car only)

Model	Day	Dusk	Night	Avg
Ground truth - depth camera				
Megadepth	47.51	71.74	68.75	65.66
DORN	58.31	84.97	34.84	70.68
LKVOLearner	47.68	69.96	48.25	60.73
SfMLearner	51.33	73.68	41.18	62.75
Monodepth	56.77	84.92	42.34	71.17
DenseDepth	62.80	59.18	71.31	62.79
Ground truth - DenseDepth				
Megadepth	52.76	60.23	29.84	53.35
DORN	74.62	75.19	65.95	73.24
LKVOLearner	54.25	56.34	40.38	52.89
SfMLearner	60.48	61.40	50.82	59.13
Monodepth	71.12	75.73	57.89	71.20

	Day	Dusk	Night	Avg
Ground truth - Megadepth				
DORN	45.89	30.01	61.68	42.54
LKVOLearner	24.39	17.72	31.53	22.95
SfMLearner	33.71	19.60	43.24	29.74
Monodepth	38.05	27.08	48.93	35.49
DenseDepth	52.76	60.23	29.84	53.35
Ground truth - Monodepth				
Megadepth	38.05	27.08	48.93	35.49
DORN	19.61	15.08	25.23	18.77
LKVOLearner	22.75	22.41	24.00	22.83
SfMLearner	20.27	18.65	21.29	19.64
DenseDepth	71.12	75.73	57.89	71.20

Table 6.4 Depth results - second dataset (car only)

Model	Day	Dusk	Night	Avg
Ground truth - DenseDepth				
Megadepth	99.52	101.37	104.43	100.56
DORN	91.49	92.99	86.44	90.90
LKVOLearner	81.77	82.27	72.82	80.47
SfMLearner	101.46	101.56	88.81	99.56
Monodepth	114.31	115.04	105.41	113.03
Ground truth - Megadepth				
DORN	114.81	120.19	126.21	117.40
LKVOLearner	40.10	46.43	51.42	42.94
SfMLearner	38.57	44.07	41.18	39.74
Monodepth	45.79	52.42	59.19	49.04
DenseDepth	99.52	101.37	104.43	100.56
Ground truth - Monodepth				
Megadepth	45.79	52.42	59.19	49.04
DORN	104.57	104.59	100.99	104.01
LKVOLearner	39.32	40.10	45.96	40.55
SfMLearner	27.72	28.66	44.76	31.18
DenseDepth	114.31	115.04	105.41	113.03

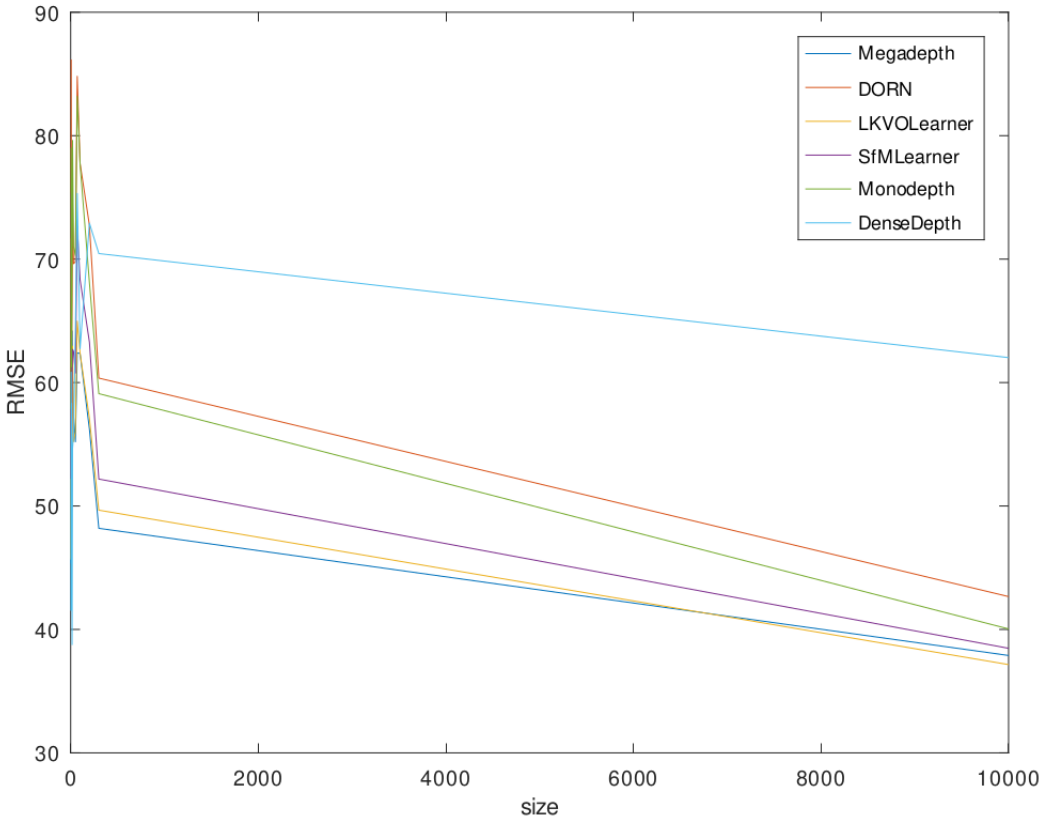

Fig. 6.2 RMSE regarding car size

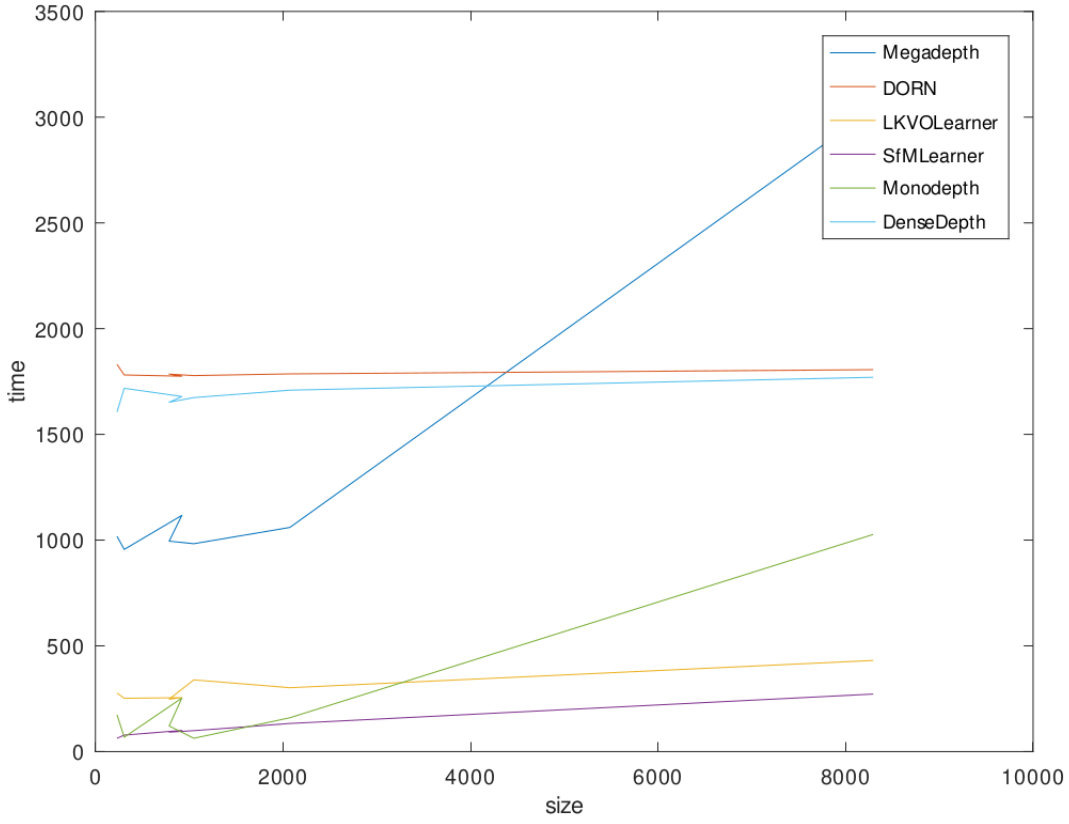

Fig. 6.3 Speed regarding image size

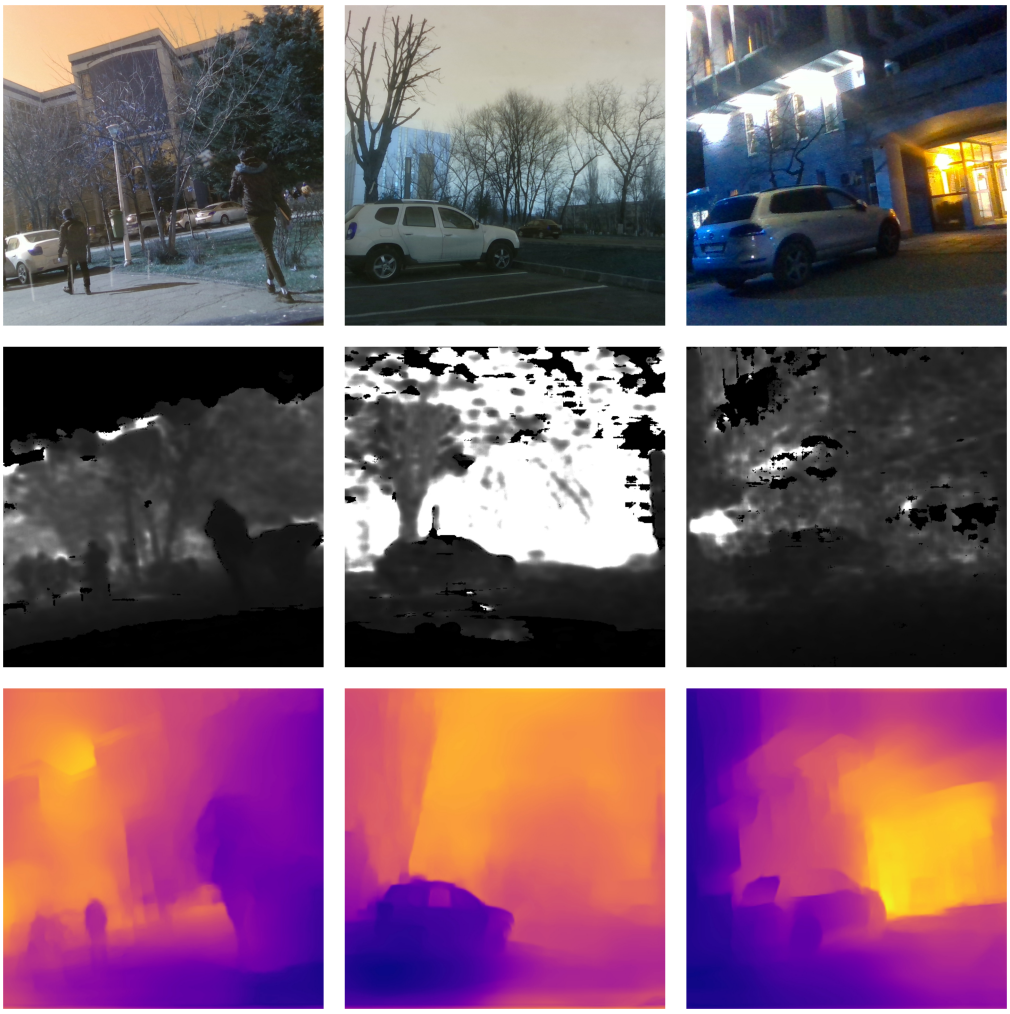

Fig. 6.4 Depth prediction results

Chapter 7

Trajectory prediction using video generation in autonomous driving - arhitecture and implementation

This chapter describes the most important task of the thesis - the trajectory prediction. For the trajectory prediction task, a new architecture is proposed, based on video prediction, object detection and semantic segmentation. The architecture used for this task can be seen as a link between all the other components of this thesis and can also be seen as its main purpose - to design a new trajectory prediction model which can be trained without the need of annotated data, using a video prediction architecture as its core network. The chapter is organized as follows. The first section describes the most important datasets used for the trajectory prediction task and also describes the dataset used for the current experiments. The next section describes the experiments made for this task and also the metrics used in the evaluation of the results. The next section presents the proposed architecture for this task and its variations regarding the current experiments. The final section of the chapter presents and analyzes the results.

7.1 The trajectory prediction dataset

As it was mentioned earlier, a dataset recorded in Politehnica university campus is used for this task, namely the POLI segmentation dataset. The dataset consists of short movies recorder during the day, dusk and night. The original dataset was used for the car segmentation task, as it was already mentioned in one of the previous sections. The most relevant frames have been chosen, with at least one vehicle in it, in order to test the trajectory prediction system on these images. For all the existing images, short movies containing 35 frames were formed. From this number, 30 are considered to be known and used to feed the neural networks and 5 are to be detected. There are 106 videos that were recorded during the daytime, 36 recorded during the dusk or dawn and 47 during

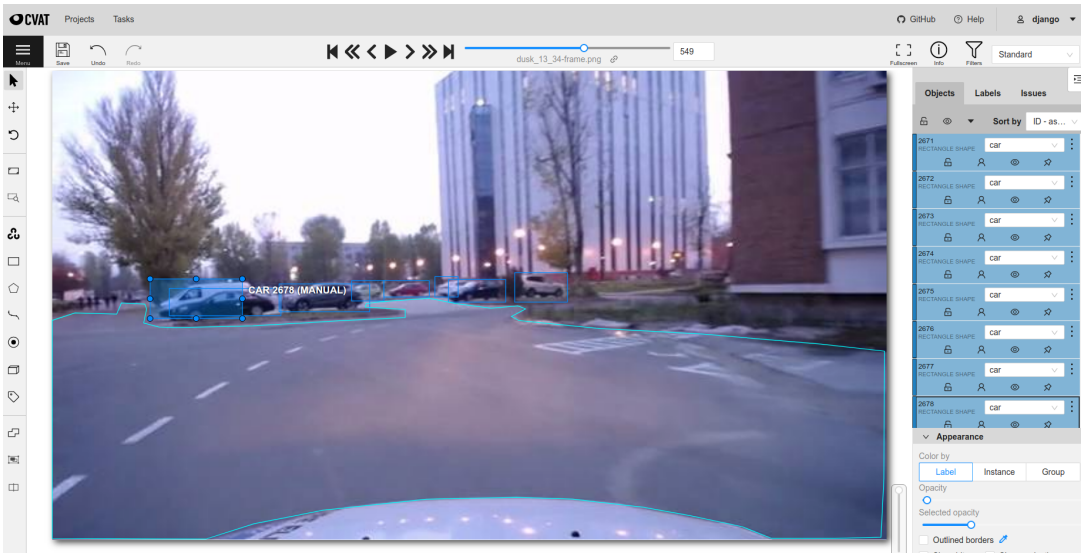

Fig. 7.1 Annotation of both the road and cars for trajectory prediction

the night, so a total of 189 short videos, containing 6675 frames. All the frames are annotated for road segmentation and car detection - the segmentation was kept from the POLI segmentation dataset and the annotation of the cars was made especially for this task, using the CVAT tool. An example of both road segmentation and car annotation using CVAT can be seen in Figure 7.1.

The frames contain about 4000 annotated cars in the real images only in the frames that have to be predicted (in about 945 frames), besides the other cars that were in the first known frames and did not have to be annotated.

7.2 Proposed architecture

7.2.1 A generic model

The architecture proposed for the trajectory prediction involves many networks, one for each of the tasks described earlier – video generation, depth estimation, semantic segmentation and object detection. The input of the architecture consists of small videos, containing about 35 frames – the first 30 are considered to be known and are used in order to feed the video generation networks, and the last 5 should be predicted. The first network involved in this process is the video generation network. The video generation network varied in the experiments made, as it can be seen in the following chapter, which describes the results. After the video generation step, some predicted frames were obtained for the given scene. The prediction was made considering only the last 5 frames predicted. The frames are predicted from the same images – the first 30 frames are used in order to predict the last 5. Some networks add the next predicted frame to the

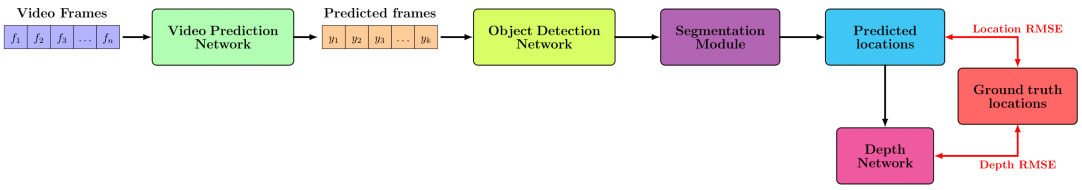

Fig. 7.2 Generic proposed architecture for trajectory prediction

real frames in order to predict another one. Once the predicted images are obtained the frames are going through an object detection network – in the current case, YOLO v4. The cars were also manually annotated for a better estimation of the generation and also in order to remove the mistakes made by the detection network. Some of the generated frames have a low quality and had to be resized to a smaller resolution. Even if a human could still identify the position of most of the cars in a low resolution picture, the object detection network will have trouble regarding the detection of the cars, which is why both the ground truth detection and the YOLO results were tested. After this step, but independent of it, the images are going through a semantic segmentation network – FCN, in the current case. This step is required in order to use the road segmentation as an additional information for the future cars. There is another module that uses the road segmentation and learns the best way to readjust the car position regarding the road. The relative position of the car regarding the road is computed with the old position and also with the new position, then the final position is obtained as an weighted average of the two positions. Even if the improvement is not big, some improvements were obtained considering the results without taking into account the road segmentation.

The final step before computing the actual metrics is to put all the predicted frames and the real frames as an input for a depth estimation network. With these results – the positions of the cars, obtained by taking into account the road segmentation, and also the depth of the frames, two relevant metrics can be computed – the RMSE for the locations and also the RMSE for the depth regarding those locations.

This way, both the distance between the real location and the predicted location and also the distance between the depths of the real car location and the predicted car location can be measured. This could be more precise than taking into account only the distance in pixels. More details about the metrics will be given in the following section. The generic architecture can be seen in Figure 7.2.

7.2.2 Model-specific architectures

The previous architecture is a generic model, without taking into account the networks used. The main purpose of the thesis is to study the most important architectures regarding each of the existing sub-tasks from the main model and use the best models in order

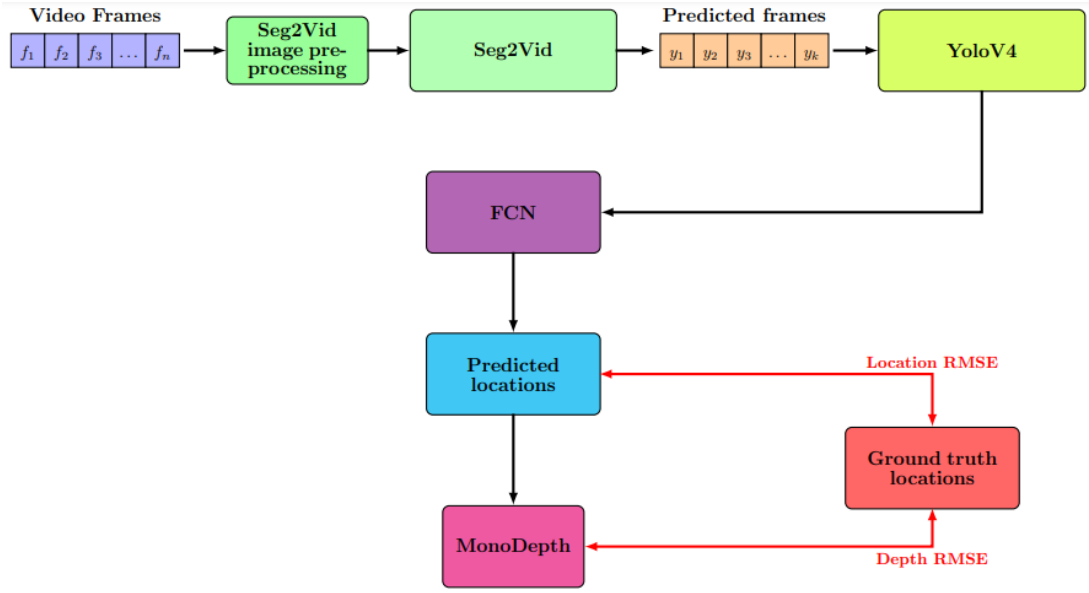

Fig. 7.3 Proposed architecture for SAVP

to have a competitive architecture.

For the current experiments, YOLO v4 has been selected as the best architecture regarding object detection, FCN as the most versatile architecture for semantic segmentation and Monodepth as the best architecture regarding the need of the experiments. Regarding the video generation architecture used, there are three models proposed and tested.

The architecture for the model based on SAVP can be seen in 7.3. The difference between the generic model is that the specific network used are also represented in this figure - SAVP, YOLO, MonoDepth and FCN. The architecture for Segnet can be seen in 7.4 and the architecture for PredNet can be seen in 7.5. Each of these architectures was tested using the dataset described in this chapter and the results are analyzed in the following section. The presented architectures represent one of the most important contributions of the thesis.

7.3 Trajectory prediction experiments and results

In this section are described the experiments used for the trajectory prediction task, how the metrics were computed..

7.3.1 Trajectory prediction results

The results obtained on all the experiments made regarding trajectory prediction and video generation are presented in this section, involving the RMSE for the predicted location, the predicted depth and also the RMSE regarding the car size and the infer-

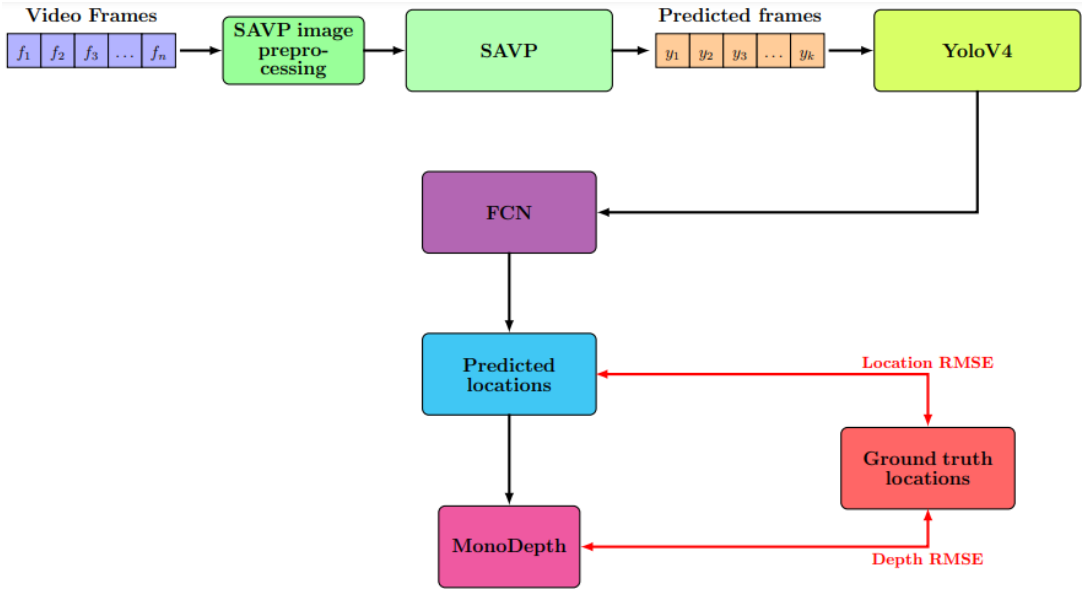

Fig. 7.4 Proposed architecture for Seg2Vid

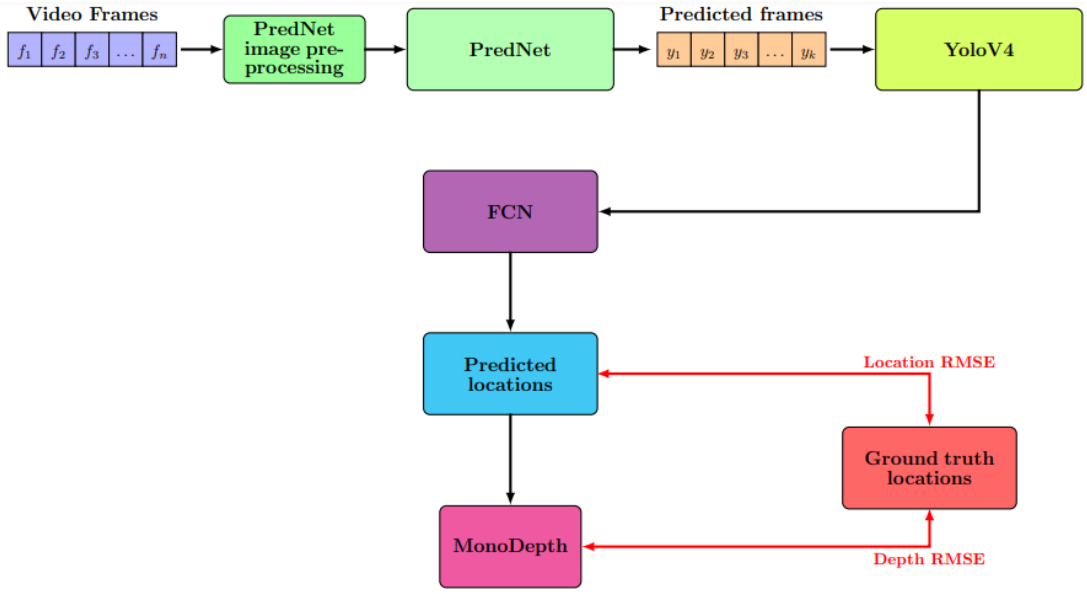

Fig. 7.5 Proposed architecture for PredNet

ence time regarding the image size. In Figure 7.6 some of the predictions are shown. There is a prediction from each of the three networks used (PredNet, SAVP and Segnet) along with the ground truth for each scenario involved (day, dusk and night). In Table 7.1 and Table 7.2 are shown the results for the location and in Table 7.3 and Table 7.4 are shown the results for the depth. Each results are analyzed in the following paragraphs.

Fig. 7.6 Video generation results

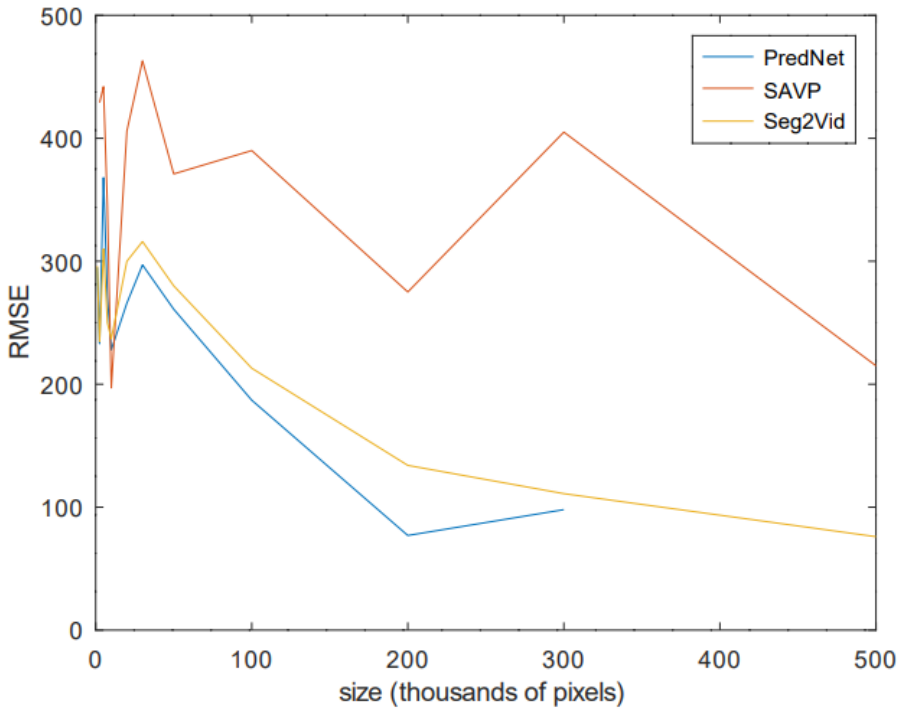

Fig. 7.7 RMSE regarding car size

7.4 An improved model for trajectory prediction

In this section is described three new models proposed as an original contribution in this thesis, considering the trajectory prediction task. The PredNet network is better analyzed

Table 7.1 RMSE for location (GT detection)

	Day	Dusk	Night	Avg
PredNet				
No segmentation	318.20	71.73	94.28	247.91
GT segmentation	317.20	71.10	93.70	247.11
FCN segmentation	317.12	71.24	93.49	247.00
SAVP				
No segmentation	294.46	128.88	116.83	233.71
GT segmentation	294.12	128.84	116.34	233.41
FCN segmentation	294.12	128.85	116.15	233.41
Seg2Vid				
No segmentation	321.20	148.91	115.27	265.85
GT segmentation	316.06	145.65	113.58	261.67
FCN segmentation	320.19	147.16	114.63	264.91
TraPHic	83.78	54.60	114.27	86.29

Table 7.2 RMSE for location (YOLO detection)

	Day	Dusk	Night	Avg
PredNet				
No segmentation	280.01	193.69	46.45	269.98
GT segmentation	275.45	157.91	46.40	263.69
FCN segmentation	275.45	158.58	46.40	263.69
SAVP				
No segmentation	615.78	568.45	543.74	561.84
GT segmentation	494.61	480.95	322.51	390.35
FCN segmentation	497.73	478.46	323.96	393.00
Seg2Vid				
No segmentation	320.88	278.37	103.58	310.92
GT segmentation	306.16	191.44	98.29	287.88
FCN segmentation	306.36	196.98	102.15	289.27

and also three modifications are proposed in this thesis considering the basic model, with better results regarding the trajectory prediction task.

7.4.1 Proposed architectures

On a higher level, PredNet can be seen as multiple recurrent convolutional layers, whose output goes through a rectified linear unit (ReLU) activation and a max-pooling layer with stride 2. Now, regarding the convolutional recurrent layers, they consists of four different layers of convolutions. The first one is a representation layer, which is a recurrent layer that makes a prediction based on the current representation input. The input and the prediction represent another two layers of convolutions. The last layer is an error layer which is computed based on the input and the prediction and it becomes the next input layer. The representation layer at a given step is based on the representation layer at the

Table 7.3 RMSE for depth (GT detection)

	Day	Dusk	Night	Avg
PredNet				
No segmentation	142.08	8.63	23.60	104.78
No segmentation (pred)	142.98	14.02	25.74	105.68
GT segmentation	145.13	8.36	22.89	111.06
GT segmentation (pred)	145.63	13.18	24.13	111.60
FCN segmentation	145.58	8.54	23.46	111.37
FCN segmentation (pred)	146.54	13.92	25.48	112.16
SAVP				
No segmentation	141.49	13.34	25.95	103.33
No segmentation (pred)	126.14	18.58	31.47	93.01
GT segmentation	141.97	13.24	25.23	106.49
GT segmentation (pred)	126.38	17.41	31.29	95.68
FCN segmentation	141.98	13.24	25.70	106.49
FCN segmentation (pred)	126.88	18.50	31.30	95.96
Seg2Vid				
No segmentation	134.10	18.68	27.52	103.76
No segmentation (pred)	129.25	23.52	32.47	100.62
GT segmentation	138.29	18.57	27.41	110.01
GT segmentation (pred)	132.85	22.54	30.79	106.21
FCN segmentation	138.28	18.66	27.41	110.01
FCN segmentation (pred)	132.84	23.36	32.23	106.20

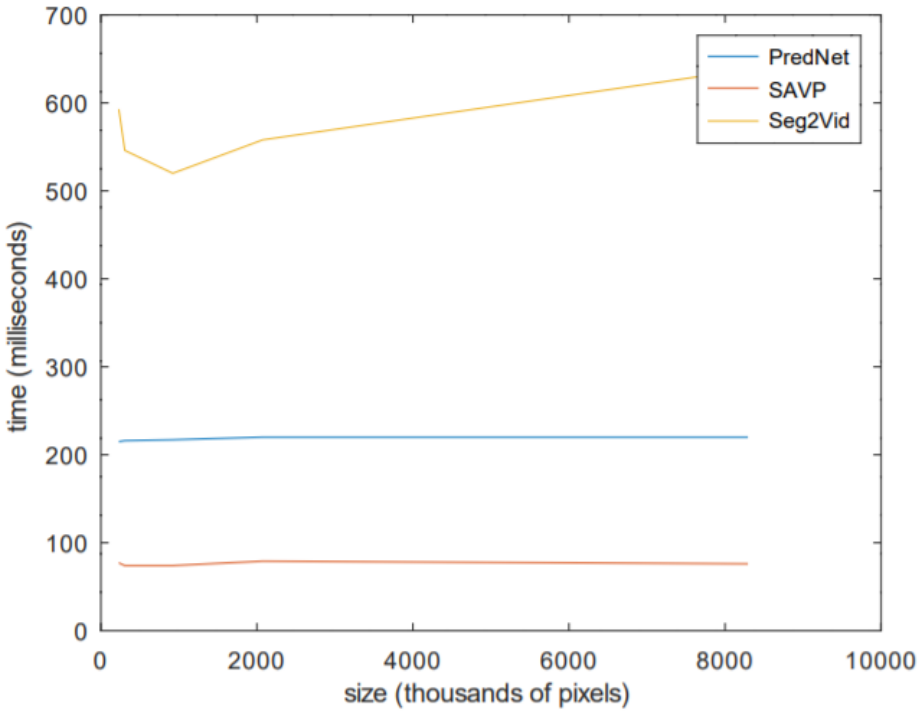

Fig. 7.8 RMSE regarding image size

Table 7.4 RMSE for depth (YOLO detection)

	Day	Dusk	Night	Avg
PredNet				
No segmentation	69.69	20.21	6.87	65.08
No segmentation (pred)	168.77	24.93	0.64	156.86
GT segmentation	62.19	19.58	6.51	58.89
GT segmentation (pred)	165.66	16.27	0.55	154.70
FCN segmentation	65.27	19.58	5.94	61.76
FCN segmentation (pred)	159.33	19.52	0.55	149.27
SAVP				
No segmentation	38.66	29.47	32.69	33.45
No segmentation (pred)	81.66	69.23	44.97	57.21
GT segmentation	34.76	31.80	36.41	36.09
GT segmentation (pred)	78.15	49.28	41.71	52.57
FCN segmentation	34.95	31.80	36.41	36.09
FCN segmentation (pred)	70.73	48.20	42.04	52.57
Seg2Vid				
No segmentation	62.51	25.72	20.26	56.57
No segmentation (pred)	137.39	40.86	19.79	122.98
GT segmentation	64.13	25.91	19.60	58.77
GT segmentation (pred)	136.17	31.78	14.89	123.16
FCN segmentation	64.10	25.91	19.60	58.75
FCN segmentation (pred)	134.93	32.61	19.78	122.65

previous step, the error layer at the previous step and also on the representation layer at the next step (which can be obtained initially by using upsampling). The main network is made in order to predict only a single future frame given an input video, however the network can also be fine-tuned in order to predict up to five frames into the future. For the current experiments, the architectures were also fine tuned to predict five future frames given only the initial video.

This research proposes three different versions of the internal representation of the convolutional layers. The standard version uses a four layer model with 3x3 convolutions for the prediction of driving images, as can be seen on their Git repository. The proposed models are the following:

The P_5_5 simply replaces the 3x3 convolutions with 5x5 convolutions, without adding any additional layers.

The P_3_5 is a 6-layer model with two extra 3x3 convolutional layers, considering the previous model, P_5_5. It also replaces the ReLU activation with PReLU, which instead of zeroing negative values it learns a parameter which is multiplied with the value for the response, acording to the following equation:

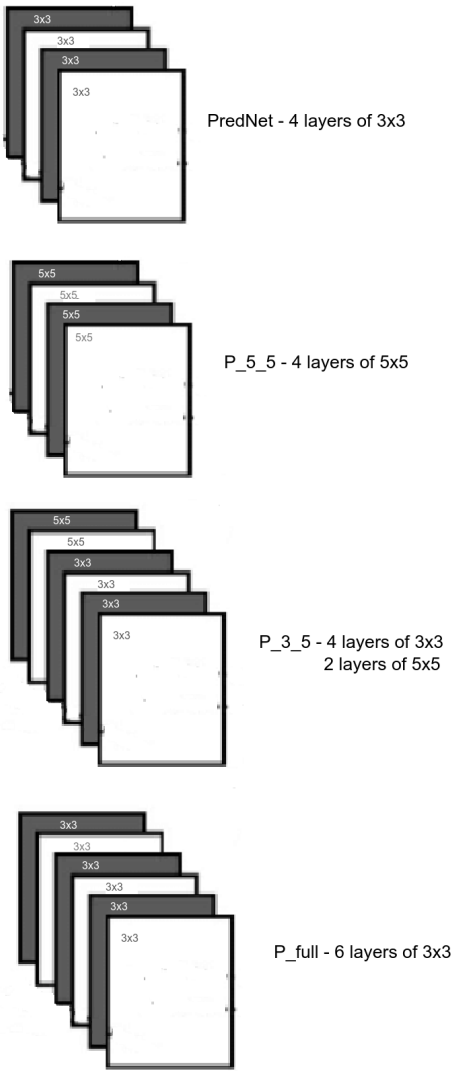

Fig. 7.9 Proposed convolutional structures

$$f(x) = \begin{cases} x & \text{if } x \geq 0 \\ a * x & \text{otherwise} \end{cases} \tag{7.1}$$

Finally, the P_full is also a 6-layer model consisting of only 3x3 convolutional layers and also using the PRELU activation function.

The modifications can be better seen in Figure 7.9.

The workflow regarding the trajectory prediction is similar to the one shown in 7.5, with the only difference that the PredNet architecture is replaced with the new variations, P_3_5, P_5_5 and P_full.

The results can be found in Table 7.5, Table 7.6 and also in Figure 7.10 and in Figure 7.11. In Figure 7.10 there are some images with the second predicted frame in different scenarios from each architecture, including the original Prednet architecture and the ground truth. Table 7.5 contains the NRMSE regarding the location for each of

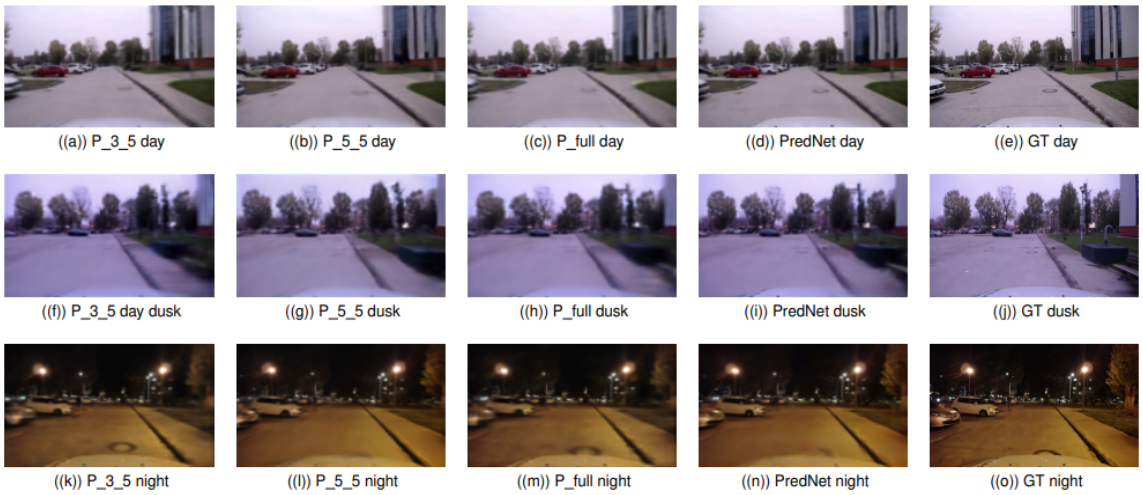

Fig. 7.10 Prediction results - modified architectures

the eighteen setups and regarding the time of the day. Table 7.6 contains the NRMSE regarding the depth for the same setups as in the first table. Lastly, in Figure 7.11 it can be seen a plot for the NRMSE taking into account the car size, considering the last setup - the detections from YOLO and the segmentation from the FCN network. In the tables the segmentation type is described by "no segm", "GT segm" or "FCN segm", considering that no segmentation was used for improving the results, the segmentation was used using the ground truth and the segmentation was used using the FCN network for the predicted images. The detection used is described by "GT det" and "YOLO det", meaning that the position of the cars were considered by the manually annotated data and by the results of the YOLO, respectively. Finally, in Table 7.6, if the depth was computed considering the predicted framed, the abbreviation "pred D" appears.

Table 7.6 Depth NRMSE

	Day	Dusk	Night	Avg
PredNet				
No segm, GT det	0.555	0.034	0.092	0.409
No segm, pred D, GT det	0.559	0.055	0.101	0.413
GT segm, GT det	0.567	0.033	0.089	0.434
GT segm, pred D, GT det	0.569	0.051	0.094	0.436
FCN segm, GT det	0.569	0.033	0.092	0.435
FCN segm, pred D, GT det	0.572	0.054	0.100	0.438
No segm, YOLO det	0.272	0.079	0.027	0.254

	Day	Dusk	Night	Avg
No segm, pred D, YOLO det	0.659	0.097	0.003	0.613
GT segm, YOLO det	0.243	0.076	0.025	0.230
GT segm, pred D, YOLO det	0.647	0.064	0.002	0.604
FCN segm, YOLO det	0.255	0.076	0.023	0.241
FCN segm, pred D, YOLO det	0.622	0.076	0.002	0.583
P_3_5				
No segm, GT det	0.555	0.041	0.083	0.418
No segm, pred D, GT det	0.552	0.046	0.076	0.415
GT segm, GT det	0.559	0.040	0.083	0.436
GT segm, pred D, GT det	0.556	0.045	0.076	0.433
FCN segm, GT det	0.557	0.041	0.083	0.434
FCN segm, pred D, GT det	0.555	0.046	0.076	0.432
No segm, YOLO det	0.293	0.110	0.055	0.263
No segm, pred D, YOLO det	0.699	0.168	0.087	0.623
GT segm, YOLO det	0.264	0.130	0.065	0.246
GT segm, pred D, YOLO det	0.682	0.083	0.066	0.614
FCN segm, YOLO det	0.241	0.116	0.050	0.241
FCN segm, pred D, YOLO det	2.401	0.073	0.070	0.577
P_5_5				
No segm, GT det	0.561	0.046	0.063	0.416
No segm, pred D, GT det	0.561	0.048	0.072	0.416
GT segm, GT det	0.565	0.044	0.063	0.432
GT segm, pred D, GT det	0.562	0.046	0.072	0.430
FCN segm, GT det	0.568	0.046	0.063	0.433
FCN segm, pred D, GT det	0.566	0.048	0.072	0.432
No segm, YOLO det	0.271	0.096	0.045	0.246
No segm, pred D, YOLO det	0.633	0.092	0.052	0.568
GT segm, YOLO det	0.254	0.095	0.041	0.235
GT segm, pred D, YOLO det	0.630	0.059	0.052	0.567
FCN segm, YOLO det	0.257	0.095	0.041	0.237
FCN segm, pred D, YOLO det	0.593	0.063	0.052	0.536
P_full				
No segm, GT det	0.544	0.036	0.071	0.410
No segm, pred D, GT det	0.541	0.048	0.089	0.409
GT segm, GT det	0.557	0.035	0.069	0.434
GT segm, pred D, GT det	0.553	0.045	0.083	0.432
FCN segm, GT det	0.555	0.036	0.071	0.433

	Day	Dusk	Night	Avg
FCN segm, pred D, GT det	0.552	0.048	0.087	0.432
No segm, YOLO det	0.293	0.110	0.055	0.263
No segm, pred D, YOLO det	0.660	0.129	0.048	0.584
GT segm, YOLO det	0.272	0.116	0.051	0.250
GT segm, pred D, YOLO det	0.680	0.083	0.054	0.610
FCN segm, YOLO det	0.237	0.112	0.046	0.237
FCN segm, pred D, YOLO det	0.640	0.069	0.066	0.574

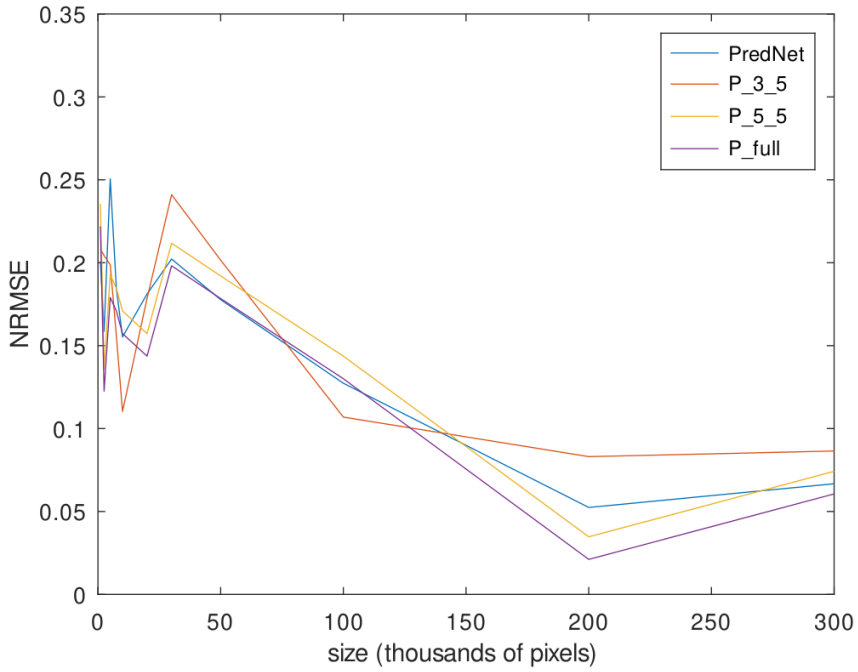

Fig. 7.11 NRMSE regarding car size

Table 7.5 Location NRMSE

	Day	Dusk	Night	Avg
PredNet				
No segm, GT det	0.217	0.049	0.064	0.169
GT segm, GT det	0.216	0.048	0.064	0.168
FCN segm, GT det	0.216	0.049	0.064	0.168
tNo segm, YOLO det	0.191	0.132	0.032	0.184
GT segm,YOLO det	0.188	0.108	0.032	0.180
FCN segm, YOLO det	0.188	0.108	0.032	0.180
P_3_5				
No segm, GT det	0.208	0.056	0.061	0.164
GT segm, GT det	0.207	0.056	0.061	0.170
FCN segm, GT det	0.207	0.056	0.061	0.163
No segm, YOLO det	0.218	0.238	0.063	0.216
GT segm,YOLO det	0.215	0.157	0.065	0.210
FCN segm, YOLO det	0.211	0.156	0.062	0.206
P_5_5				
No segm, GT det	0.204	0.054	0.046	0.158
GT segm, GT det	0.204	0.053	0.046	0.158
FCN segm, GT det	0.204	0.054	0.046	0.158
No segm, YOLO det	0.225	0.136	0.050	0.210
GT segm,YOLO det	0.208	0.103	0.047	0.193
FCN segm, YOLO det	0.208	0.101	0.050	0.193
P_full				
No segm, GT det	0.209	0.057	0.050	0.164
GT segm, GT det	0.209	0.057	0.049	0.164
FCN segm, GT det	0.209	0.057	0.050	0.164
No segm, YOLO det	0.187	0.207	0.033	0.186
‘ GT segm,YOLO det	0.177	0.121	0.028	0.179
FCN segm, YOLO det	0.177	0.122	0.028	0.172
Traffic	0.057	0.037	0.078	0.059

Chapter 8

Conclusions and future work

The present thesis addresses the most important tasks regarding autonomous driving, from scene understanding to trajectory prediction for the surrounding cars. The thesis followed closely four different problems - object detection, road semantic segmentation, depth estimation and trajectory prediction. Besides these, object tracking, instance and panoptic segmentation and video generation were also discussed. For each of these tasks there were described the most important research papers and also review articles and the corresponding datasets for each task. Also, some of the best networks were analyzed on some datasets annotated at the Politehnica University of Bucharest campus, taking into account the time of the day, the size of the cars, the inference time and other statistics, in order to see how it can be achieved the best performance in a real life application for autonomous driving. The final goal of the thesis is to propose and implement an architecture for trajectory prediction, based on the evaluation results for some of the best architectures in the literature for object detection, semantic segmentation, depth estimation and video prediction. This approach could be a game changer for the trajectory prediction task, because it doesn't involve any annotated data and can be trained using any possible driving video from the internet. Also, each task is backed up by at least one research paper, which contain some of the results presented in the thesis. The results are detailed regarding each of these four individual tasks.

8.1 Most important results

For the object detection task Yolo, Faster R-CNN, SSD and Retina Net were tested. In the experiments made, Yolo has the best mean average recall but SSD has the best mean average precision. Also, as expected, the results are better when there were tested only two categories – car and person, the recall is better in the day than in the night, but the precision does not have important variations regarding the day time. For the object detection task a new dataset, which was manually annotated, was recorded in the University Politehnica of Bucharest campus. Regarding the dataset made in Politehnica

University, the recall is better than for the BDD100k dataset, because there are less objects involved, but the precision remained in the same interval. This result shows that generally the detectors are robust and have the same behavior on both POLI dataset and the BDD100k dataset. Regarding the object size, the bigger objects were generally detected, there is an increase in the recall, but the detected class tends to be wrong in many cases than for the smaller objects, which were poorly detected, but in case of a detection the class was correctly assigned to the object. Regarding the time needed for a detection, Yolo had the best results and could perform in a real time scenario (especially regarding their tiny model, which can perform in real time for 30 fps), with the mention that a powerful running GPU is needed in order to achieve these results.

For the semantic segmentation task SegNet , ENet, PSPNet, Dilation and DaNet were tested. The best networks were PSPNet and FCN, which have really good percentages for the metrics shown, and can be used in real applications, and the worst network tested was DaNet, which can't be trusted in a road semantic segmentation without a further tuning of the network especially for this task. However, the model is limited to the road segmentation and didn't include other categories, such as the segmentation of the cars. A specially designed dataset for this task, Poli segmentation dataset, was recorded in the university campus. As expected, the results were better in the day and worse in the night. Unfortunately, regarding the processing time, the networks can't perform in real life scenarios, where a segmentation could be needed for 30 frames per second – at most, they could process only a few images per second. There is still room for improvement regarding the accuracy, but the biggest problem now is the inference time. With only one exception, the time for evaluating an image did not had variations regarding the images size, which is a positive aspect.

For the depth estimation task, two datasets were used – the first dataset was the one recorded with the Intel RealSense camera and the second one is the previously dataset made in the Politehnica university campus and used for semantic segmentation, which was used in order to test the networks against each other, in order to see what are the advantages of such information regarding an unsupervised learning paradigm. All the results were analyzed regarding the day, dusk and night. The RMSE was computed for each of this category and also an average error for all three. The speed of the networks for different image sizes and the influence of the car size regarding the RMSE was analyzed and also both the depth error for all the pixels in the image and the depth error only for the car were analyzed. Unsurprisingly, the error considering only the cars was significantly smaller. Most of the networks performed in similar times, even if the size of the images were different, which is a good result taking in account a real life scenario with full HD or even 4K images. It can also be seen that the precision of the depth estimation was better for bigger cars, which also was an expected result. From

the experiments made, the best networks were SfMLearner and LKVOLearner, but also Monodepth 2 performed well, considering also a manual refinement of the quality of the results. Given that the RGB-D camera is not as precise as a LIDAR scanner, a qualitative estimation of the resulting images showed that Monodepth 2 should also be considered. However, because the experiments were very different for almost every network there is at least one scenario where the network performed better than the rest of the others, which could be explained by the imprecise ground truth and also imprecise results of the networks, in general. The good news is that the networks could be used in real life applications, due to their speed, and an autonomous system could benefit from their estimated depth for the surrounding cars. However, without using expensive sensors, the estimated results could differ a lot for the real depth, especially in the dusk or in the night. The results vary regarding the network used and the ground truth, but generally the best results were obtained during the day.

For the final trajectory prediction task, PredNet, Seg2Vid and SAVP were tested and their results are compared with TraPHic. The POLI Segmentation dataset was also used here. The most relevant network tested was PredNet when considering the RMSE for the location with the YOLO predictions for the cars, however SAVP had also good results. Three new variations of the PredNet models were developed, with the final model, PredNet_full, obtaining better results than the basic version in almost all the experiments. Although the results are still not as good as a specialized trajectory prediction network, the biggest advantage is that a video generation network can be trained on any video, making the training process easier. With a very good generation network, the trajectories could be easily inferred. Also, the results showed that the segmentation of the road could slightly improve the simple detection of the cars in the predicted frames.

8.2 Original contributions

This thesis has several new contributions regarding the object detection, semantic segmentation, depth estimation and trajectory prediction tasks, that could help the research community in order to develop better autonomous driving algorithms.

The first and the most important contribution is a new trajectory prediction model, which is based on the video generation and uses object detection, semantic segmentation and depth estimation. Each of the particular tasks that are related to the final architecture was deeply analyzed in order to use the best existing networks in the final model. Even if a dedicated trajectory prediction obtains, at the moment, better results, a model based on video generation has the advantage that it doesn't require any manually annotated training data and can be trained using any driving video that exists on the internet. Three different architectures based on three different networks were proposed and tested and

the results were presented in the thesis.

The second contribution consists in proposing three different models for video prediction, by modifying the original PredNet architecture regarding the convolutional layers and the activation function. Two of the proposed models have generally better results than the original network, and the last model proposed, P_full, outperforms the original network in almost any experiment.

The third contribution consists in three new different datasets recorded in the University Politehnica of Bucharest campus and manually annotated. The first dataset was used for object detection, the second dataset was made for road semantic segmentation and also for trajectory prediction, containing annotations of the road and the cars, and the last dataset was collected using an Intel RGB-D camera and was used for the depth estimation task.

The fourth contribution consists of an extensive testing of different architectures for autonomous driving considering the light and time of the day for different tasks (object detection, semantic segmentation, depth estimation, trajectory prediction), an approach which is not very often considered in the literature . Because generally the networks are trained on some specifics datasets, the results can vary a lot if the experiments are made during the day, the dawn or dusk or during the night. From the current experiments it can be seen that the results tend to be significantly worse during the night. This is due to the lack of training in dawn or night conditions and should be considered in the developing of other datasets and in the training of future architectures.

The fifth contribution that can be highlighted is an up-to-date review of the state-of-the-art architectures regarding the enumerated tasks and a comparative analysis. The reviews the reviews are discussing architectural aspects and include different tests for some of the best architectures with different statistics like the time of the day, the car size, the inference time.

8.3 Future work

For the object detection task, the detection should be improved in the future in order to trust a network for a self driving car application and the current study can be further continued by proposing a new object detection architecture. Even if the precision has decent values, the recall should be improved in order to have more objects detected – an object which is not detected is a possible cause of an accident, so it is important to have a better recall in the future networks. Also, the time can be improved, because aside of the

detection there are other components that have to run between two frames (segmentation, vehicle and depth prediction, etc), so better inference time are also needed. In the future, it would be a good idea to have these models fine tuned for the Politehnica dataset, in order to see how the parameters will adjust when the networks are trained on the same dataset. Also, the dataset could be increased regarding the number of the objects and their diversity.

For the semantic segmentation, the current study should extend the dataset with more than one class, in order to see the segmentation results at least for vehicles and people. Also, more networks should be fine tuned in future applications of this study especially for the road segmentation task (to output only two categories, road or not road), to see how the results will improve.

For depth estimation, the current study should make a better dataset with LiDAR sensors made in order to better estimate the estimation error. The errors could also be lessened if the networks would be trained using the desired dataset, but for this purpose the dataset should be recorded using stereo cameras, as most of the networks are trained with stereo datasets and tested with monocular ones.

For the trajectory prediction task, the future applications of this study should specifically develop and train a video generation model especially considering the task of trajectory prediction. Also, at each step (detection, segmentation, depth estimation) the corresponding models could be fine tuned in order to work better for a specific dataset and for the task of trajectory prediction in autonomous driving.

Published Papers

D. T. Iancu, A. M. Florea, “An improved vehicle trajectory prediction model based on video generation”, Studies in Informatics and Control. Accepted for publication for vol. 32, no 1, 2023

D. T. Iancu, M. Nan, A. S. Ghita and A. M. Florea, “Trajectory Prediction using Video Generation in Autonomous Driving,” Studies in Informatics and Control, vol. 31, no. 1, pp. 37–48, 2022. WOS:000779783700004, Impact Factor 2.18

A. S. Ghita, A. M. Florea, M. Nan and D. T. Iancu, “People Trajectory Prediction applied on Social Robotics Scenarios,” UPB Scientific Bulletin, Series C: Electrical Engineering and Computer Science, 2022. Under review.

D. T. Iancu, M. Nan, A. S. Ghita and A. M. Florea, “Vehicle Depth Estimation for Autonomous Driving”, UPB Scientific Bulletin, Series C: Electrical Engineering and Computer Science, pp. 3–20, 2021. WOS:000692196300001

V. Radu, M. Nan, M. Trascau, D. T. Iancu, A. S. Ghita and A. M. Florea,“Car crash detection in videos”, in 2021 23rd International Conference on Control Systems and Computer Science (CSCS), pp. 127–132, IEEE, 2021. Under WOS indexing process.

A. S. Ghita, M. Nan, D. T. Iancu and A. M. Florea, “Top-level Scene Information Extraction from Eye-level View Images”, in 2021 23rd International Conference on Control Systems and Computer Science (CSCS), pp. 133–137, IEEE, 2021. Under WOS indexing process.

D.T. Iancu, A. Sorici, A.M.Florea, “Neural road semantic segmentation in driving scenarios, International Conference on Electronics”, Computers and Artificial Intelligence (ECAI) (pp. 1-6), 2020 IEEE. WOS:000627393500073

D.T. Iancu, A. Sorici, A.M.Florea, “Object detection in autonomous driving - from large to small datasets”, International Conference on Electronics, Computers and Artificial Intelligence (ECAI) (pp. 1-6), 2019 IEEE. WOS:000569985400026

Participation in research grants

A. M. Florea et al. PETRA - People Detection and Tracking for Social Robots and Autonomous Cars, PN-III-P2-2.1-PED-2019-4995, 2020-2022.

A. M. Florea et al. ROBIN - Robots and Society: Cognitive Systems for Personal Robots and Autonomous Vehicles, complex project Nr. 72 PCCDI, 2018-2020.