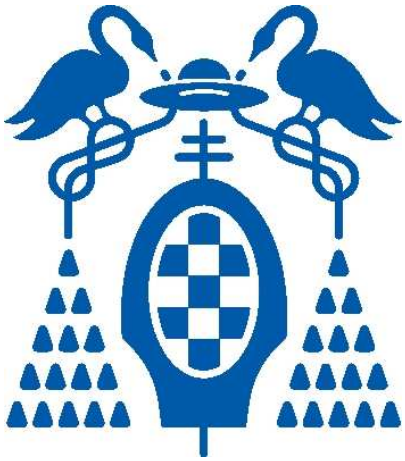
UNIVERSIDAD DE ALCALÁ

ESCUELA POLITÉCNICA SUPERIOR

DEPARTAMENTO DE ELECTRÓNICA



Doctoral Thesis

Supervised Learning and Inference of Semantic Information from Road Scene Images: Objects and Layout

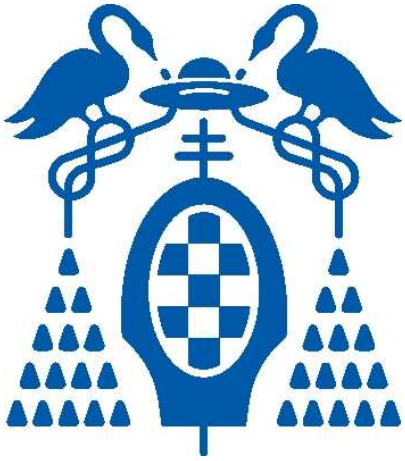
José Javier Yebes Torres

2014

UNIVERSIDAD DE ALCALÁ

ESCUELA POLITÉCNICA SUPERIOR

DEPARTAMENTO DE ELECTRÓNICA



Supervised Learning and Inference of Semantic Information from Road Scene Images: Objects and Layout

Author

José Javier Yebes Torres

Advisor

Dr. Luis M. Bergasa Pascual

2014

Doctoral Thesis

A Laura: fuente de energía y ánimo constante.

Y a mi familia: por entender el camino de la investigación.

"Imagination is more important than knowledge,
because knowledge is limited
while imagination embraces the entire world."
Albert Einstein

"The whole point of getting things done
is knowing what to leave undone."
Lady Reading

Agradecimientos/Acknowledgements

Y ponían en sus recuerdos unas notas de palpitante realidad.

"El camino", 1950, Miguel Delibes.

Esta sección está dedicada con especial cariño y entusiasmo a todas aquellas personas que directa o indirectamente han estado cerca durante el desarrollo de mi Tesis Doctoral, y que inevitablemente han influido en mi crecimiento personal e investigador en estos años.

En primer lugar, sin duda, quiero agradecer a mi director de Tesis, Luis Miguel Bergasa, su dedicación y continua supervisión a lo largo de estos años. Más allá de los formalismos y después de haber conocido a otros doctorandos dentro y fuera de la Universidad de Alcalá, creo sinceramente que fuí muy afortunado al recibir su primera llamada en el verano de 2008. La financiación conseguida, los proyectos en los que he colaborado y la libertad de investigación pero con el rigor adecuado, han hecho posible la culminación de esta Tesis. Y a pesar de las dificultades propias de la investigación, las estancias nos han recompensado a los dos.

En segundo lugar, mil gracias y un saludo muy fuerte a los compañeros de RobeSafe y ahora también de ISIS. Siempre ha habido un gran ambiente colectivo y me llevo muy buenas amistades que me han hecho crecer personalmente y disfrutar de grandes momentos tanto en la universidad como fuera de ella. Pedro me contagió la fiebre de Kubuntu y tras él continué la estirpe de "ciclistas RobeSafe"; Jesús fue el primero en rellenar de correcciones mi primer paper; Pablo, gran investigador y persona cuyo apoyo e inspiración fue fundamental en mis primeros pasos; Iván, un caballero y gran docente en continua monitorización de la somnolencia; Almazán, todo un señor e incansable investigador con el que compartí muchas horas de laboratorio; Fer, ¿qué sería del día a día sin sus anécdotas, ni las celebraciones sin su gestión?; Llama, Carlos, Sergi, Raúl, Llorca, Nacho, Sebas, Álvaro, Edu, Balqui, Óscar y Gavi que estuvieron desde mi llegada a RobeSafe aportando nuevos desarrollos al mundo ITS y de la Robótica. Más recientemente y no por ello menos importantes, Roberto, gran compañero y excelente alcarreño con el que he compartido muchas horas de trabajo y reflexiones en los dos últimos años. También Edu, al que agradezco los mapas 3D con láser y sus interesantes tertulias expertas en cine y wrestling entre otros. Mención especial a las grandes investigadoras del departamento: Noe, Fani y Christina; sin tampoco olvidarme de Álvaro, que ya esta hecho todo un suizo.

A parte de los nombrados, muchos otros han pasado por el lab de manera temporal, con los que hubo charlas ingenieriles. Y muy recientemente me alegra haber podido ayudar y aprender

de Alberto, Sergio y Mario, que espero que el futuro les depare grandes metas.

Gracias también al resto de compañeros del Departamento de Electrónica, en especial a Revenga, por estar siempre disponible y saber "de todo", y a Rafa por su visión de la bioingeniería desde el lado futbolero. Además, gracias a M.A. Sotelo por hacer posible que el IV'12 se celebrase en la UAH y por los congresos compartidos en Baden-Baden y Washington.

Además, en este apartado no podría olvidarme de mis colegas de profesión internacionales.

Special thanks to Prof. Raquel Urtasun, who accepted me as a visitor researcher, when I was not yet aware of her incredible research record and her reputation in the computer vision and machine leaning community. Visiting TTI and Chicago was a very remarkable milestone in my life and a great opportunity to approach 1st class research. Thanks for your supervision and your research passion at any time. Besides, dear Alex Schwing, thank you for your support while trying to *'hit the roads'*, I learned a lot from your research work and I enjoyed the stay during the harsh (at least for a Spanish guy) Chicago winter. I also recall the summer with Martin Helmer, Uri Patish and Aydin Varol and the trips to Sister Bay and New York. Cheers also to Jian Yao, Subhransu Maji and others in TTI. Thanks to Mathieu Salzmann for his availability during a 'remote exam' and the interesting chats and good 'beer moments'.

Furthermore, Darmstadt was also an important piece of my research and life experience. Special thanks to Prof. Stefan Roth, who accepted me in his Visual Inference Group and provided interesting and helpful comments on the object detection with DPM. Many and kindly regards for my office mate Ulrich Krispel and the sportsman and Munich fan Matthias Kirchner. We shared interesting talks, relaxing moments and sports events. You are great friends. Cheers also to some more colleagues in GRIS department.

Para finalizar, me gustaría agradecer con especial cariño a Laura su compresión y apoyo durante estos años, y sobre todo en la última fase, donde la Tesis, por horas, pareció reemplazar lo ireemplazable. Ahora es tiempo de seguir disfrutando y mirar al horizonte. Gracias a mi padres también por soportar tantas veces la distancia y los retos de una vida investigadora. Ellos me han dado la oportunidad de ser quién soy a día de hoy. Lo que yo he conseguido también es vuestro. Gracias a mi hermana, por aportarle una chispa diferente a la vida, y gracias a mis abuelos por la gran inversión que hicieron al emigrar de sus pueblos queridos a las frenéticas e industrializadas ciudades. Y sobre todo ellas, las abuelas, cuyo cariño infinito siempre estará presente. Además, nunca imaginé que acabaría escribiendo estas líneas y gran parte de la tesis en la habitación donde compartí con vosotros mi infancia y donde os arropé durante la vejez. Gracias de todo corazón.

Resumen

En la actualidad, la industria del automóvil utiliza cámaras y técnicas de visión para integrar funcionalidades avanzadas que asisten a las personas durante la conducción. Sin embargo, la investigación en *vehículos autónomos* supone un paso más allá de los sistemas ADAS y es un área de gran interés tanto en el sector académico como industrial. Son muchos los desafíos que surgen a raíz de las plataformas robóticas autónomas en escenarios urbanos, debido principalmente a su complejidad en cuanto a la estructura de la escena y a los participantes dinámicos (peatones, vehículos, vegetación, etc.). Por este motivo, proveer a dichas plataformas de las capacidades para el entendimiento de escenas es un objetivo esencial, ya que las cámaras captan las escenas 3D de forma muy similar a como es percibida por una persona. De hecho, la necesidad de realizar *entendimiento de escenas 3D*, ha provocado un creciente interés en el etiquetado conjunto de los objetos y la estructura de la escena. Concretamente, con el objetivo de inferir la geometría y otra información semántica relevante en entornos urbanos. En este aspecto, esta Tesis aborda dos desafíos: 1) la predicción de la geometría de intersecciones de carreteras y/o calles y, 2) la detección y la estimación de la orientación de coches, peatones y ciclistas. Para llevar a cabo dicho etiquetado automático, se extraen distintas características visuales de imágenes estéreo pertenecientes a la base de datos pública conocida como KITTI. En consecuencia, para inferir los objetos y las intersecciones en escenas de carretera, esta Tesis propone un aprendizaje supervisado de modelos discriminativos, haciendo uso de técnicas robustas de "aprendizaje máquina" para recolectar la información relevante de las características visuales.

Para llevar a cabo la primera de las tareas, se emplean mapas 2D de ocupación, que se construyen a partir de las secuencias estéreo capturadas por un vehículo en movimiento en una ciudad de tamaño medio. En base a estas imágenes de vista de pájaro, se propone una parametrización para carreteras rectas y otra para intersecciones de 4 vías. A su vez, las dependencias entre las variables aleatorias discretas que definen dicha geometría se representan mediante Modelos Gráficos Probabilísticos. A continuación, el problema se formula como una predicción estructurada, utilizando *Conditional Random Fields* (CRF) para el entrenamiento y *convex Belief Propagation* (dcBP) y *Branch and Bound* (BB) para realizar inferencia. La validación de la metodología propuesta se realiza mediante un conjunto de pruebas a partir de imágenes reales e imágenes sintéticas con diferentes niveles de ruido aleatorio. Además se incluye un análisis de las dificultades observadas para el caso de escenas reales, ya que, estas imágenes recuperadas de las secuencias estéreo presentan unos mapas de ocupación dispersos y ruidosos.

En relación a la detección y la estimación de la orientación de objetos en scenas de carretera, el objetivo de esta Tesis es competir en el desafío internacional conocido como *KITTI evaluation benchmark*, que anima a los investigadores a avanzar el estado del arte actual en métodos de reconocimiento visual, y en particular para el entendimiento de escenas 3D urbanas. Esta Tesis propone modificar el detector de objetos basado en partes y ampliamente conocido como DPM, con el propósito de aprender modelos mejorados a partir de datos 2.5D (color y disparidad). Por este motivo, se revisa el planteamiento del DPM, que está basado en descriptores HOG y "mixture models" que se entrenan mediante "latent SVM". En base a ello, esta Tesis realiza una serie de modificaciones sobre el método DPM: I) Se extiende el proceso de entrenamiento del DPM para adaptarlo a las nuevas "3D-aware features" diseñadas. II) Se realiza un análisis detallado del aprendizaje paramétrico supervisado para distintas configuraciones. III) Se introducen dos planteamientos adicionales con el objetivo de mejorar la detección de objetos: "whitening" de las características visuales y análisis de consistencia entre las vistas estéreo. Adicionalmente, a) se analiza la base de datos de imágenes KITTI y detalles importantes en relación al protocolo de evaluación; b) un largo conjunto de experimentos de validación cruzada muestran el rendimiento de las contribuciones propuestas y se comparan contra una línea de base que usa DPM y, c) finalmente, los resultados de nuestra propuesta se publican en el ranking de la web de KITTI, siendo el primer planteamiento que se publica basado en datos estéreo, obteniendo una mayor precisión en la detección de coches (3%-6%) y consiguiendo el primer puesto para la detección de ciclistas.

Palabras clave: detección de intersecciones, CRF, detección de objetos, DPM, datos 2.5D.

Abstract

Nowadays, vision sensors are employed in automotive industry to integrate advanced functionalities that assist humans while driving. However, *autonomous vehicles* is a hot field of research both in academic and industrial sectors and entails a step beyond ADAS. Particularly, several challenges arise from autonomous navigation in urban scenarios due to their naturalistic complexity in terms of structure and dynamic participants (e.g. pedestrians, vehicles, vegetation, etc.). Hence, providing image understanding capabilities to autonomous robotics platforms is an essential target because cameras can capture the 3D scene as perceived by a human. In fact, given this need for *3D scene understanding*, there is an increasing interest on joint objects and scene labeling in the form of geometry and semantic inference of the relevant entities contained in urban environments. In this regard, this Thesis tackles two challenges: 1) the prediction of road intersections geometry and, 2) the detection and orientation estimation of cars, pedestrians and cyclists. As source of data for these semantic labeling tasks, different features extracted from stereo images of the KITTI public urban dataset are employed. Then, in order to predict the objects and intersection layouts in road scenes, this Thesis proposes a supervised learning of discriminative models that rely on strong machine learning techniques for data mining visual features.

For the first task, we use 2D occupancy grid maps that are built from the stereo sequences captured by a moving vehicle in a mid-sized city. Based on these bird's eye view images, we propose a smart parameterization of the layout of straight roads and 4 intersecting roads. The dependencies between the proposed discrete random variables that define the layouts are represented with Probabilistic Graphical Models. Then, the problem is formulated as a structured prediction, in which we employ *Conditional Random Fields* (CRF) for learning and *convex Belief Propagation* (dcBP) and *Branch and Bound* (BB) for inference. For the validation of the proposed methodology, a set of tests are carried out, which are based on real images and synthetic images with varying levels of random noise. Besides, we include an analysis of the difficulties observed when employing the sparse and noisy grid maps recovered from stereo images.

In relation to the object detection and orientation estimation challenge in road scenes, this Thesis goal is to compete in the international challenge known as *KITTI evaluation benchmark*, which encourages researchers to push forward the current state of the art on visual recognition methods, particularized for 3D urban scene understanding. This Thesis proposes to modify the successful part-based object detector known as DPM in order to learn richer models from 2.5D data (color and disparity). Therefore, we revisit the DPM framework, which is based on HOG features and mixture models trained with a latent SVM formulation. Next, this Thesis performs

a set of modifications on top of DPM: I) An extension to the DPM training pipeline that accounts for our contributed 3D-aware features. II) A detailed analysis of the supervised parameter learning for different setups. III) Two additional approaches are presented with the aim of improving object detection: “feature whitening” and “stereo consistency check”. Additionally, a) we analyze the KITTI dataset and several subtleties regarding to the evaluation protocol; b) a large set of cross-validated experiments show the performance of our contributions against a baseline DPM approach and, c) finally, our best performing approach is publicly ranked on the KITTI website, being the first one that reports results with stereo data, yielding an increased object detection precision (3%-6%) for the class ’car’ and ranking first for the class ’cyclist’.

Keywords: intersections layout, CRF, object detection, DPM, 2.5D data.

Contents

List of Figures

List of Tables

List of Acronyms

ADAS	Advanced Driver Assistance Systems.
AOS	Average Orientation Similarity.
AP	Average Precision.
AuC	Area under the Curve.
BB	Branch and Bound.
BeP	Bird's eye Perspective.
BoF	Bag of Features.
CAD	Computed-Aided Design.
CNN	Convolutional Neural Networks.
CRF	Conditional Random Fields.
CV	Computer Vision.
DARPA	Defense Advanced Research Projects Agency.
dcBP	distributed convex Belief Propagation.
DPM	Discriminative Part-based Models.
DRC	DARPA Robotics Challenge.
DTPBM	Discriminatively Trained Part-Based Models.
ELAS	Efficient Large-Scale Stereo Matching.
ESS	Efficient Subwindow Search.
FN	False Negatives.
FP	False Positives.
FPPI	False Positives Per Image.
GPS	Global Positioning System.
HOG	Histogram of Oriented Gradients.

HSC	Histogram of Sparse Codes.
IG	Integral Geometry.
IoU	Intersection over Union.
LAMR	Log-Average Miss Rate.
LiDAR	Light Detection And Ranging.
LSVM	Latent Support Vector Machine.
ML	Machine Learning.
MRF	Markov Random Field.
NMS	Non-Maximum Suppression.
NP-hard	Non-deterministic Polynomial-time hard.
PASCAL	Pattern Analysis, Statistical Modelling and Computational Learning.
PCA	Principal Component Analysis.
PGM	Probabilistic Graphical Models.
RaDAR	Radio Detection And Ranging.
SGM	Semi-Global Matching.
SLAM	Simultaneous Localization And Mapping.
SSVM	Structured Support Vector Machine.
SVD	Singular Value Decomposition.
SVM	Support Vector Machine.
TN	True Negatives.
TP	True Positives.
VOC	Visual Object Classes.
WHO	Whitened Histograms of Orientations.
ZCA	Zero Components Analysis.

List of Symbols

Σ	Sample covariance matrix.
C	Constant regularization parameter for SVM variants.
Δ	Loss function for learning.
Z_d	Depth or distance in meters of the scene elements from a stereo camera.
D	Disparity obtained from stereo images.
$\mathbf{x}$	Observed variable: a feature vector from color, disparity or BeP images.
$\boldsymbol{\mu}$	Sample mean vector.
$\boldsymbol{\theta}$	Vector of parameters that are learned on training.
ϕ	Potential employed for discriminative learning and inference. It can be viewed as a function that depends on different number of random variables.
y	Random or deterministic variable instance.
$\mathcal{Y}$	Random variable set.
$Z(\mathbf{x})$	Set of latent hypotheses in DPM.

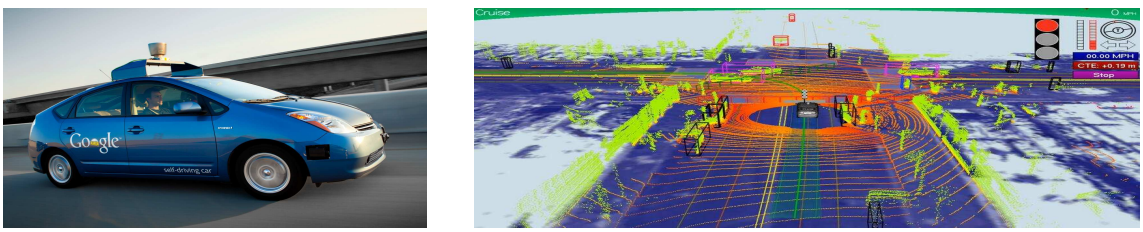
Chapter 1

Introduction

- Robotic traffic agent: "Didn't you see the cyclist?".
- Autonomous vehicle: "There was a human-driven car partially occluding my sight".
- Pedestrian: "I'm disoriented. Could anybody provide an understanding of this scene?"

A science fiction Tale? by J. Javier Yebes.

Probably, this science fiction tale may become a reality in the next 20 or 30 years, in which, assisted by robotic platforms, road fatalities in urban and interurban environments could asymptotically approach to zero and autonomous navigation would dominate our mobility patterns. Actually, the uncertainty and freedom of human decision-making (but also governed by laws and rules) cause risks and reliability at the same time. Therefore, as many times devised in books and films, could we live in a world fully dominated by machines in a total reliable way? We do not know yet, although autonomous or driverless vehicles [Google, 2011, MRG Oxford, 2012, Broggi et al., 2013] have arrived and will be present in the market within the next years. However, many challenges remain unsolved and the only way of imagining a robotic world is through an intermediate step: the co-existence of autonomous and non-autonomous robotic platforms.



(a) Google's car (b) Google's self driving tests

Figure 1.1: Current 3D scene understanding for autonomous vehicles. The images have been obtained from [Google, 2011].

In relation to autonomous navigation, having huge mapping data and relying on GPS is not enough, because of the constant need of accurate map updates, the loss and limited precision of

GPS signal and the dynamic changes of the environment. Indeed, man-made environments use to be very dynamic, i.e. moving vehicles, pedestrians, cyclists, urban structure and road changes, traffic signals modifications, etc. On the other hand, the basic human sensory sources while driving are the eyes and brain training, i.e. the visual perception of the scene and the previous knowledge about traffic rules, etc. Hence, providing image understanding to autonomous vehicles and robots in general, is a key issue that the research community and the industry have been recently working on and will continue doing [Buehler et al., 2009, DARPA RC, 2013, Geiger et al., 2014].

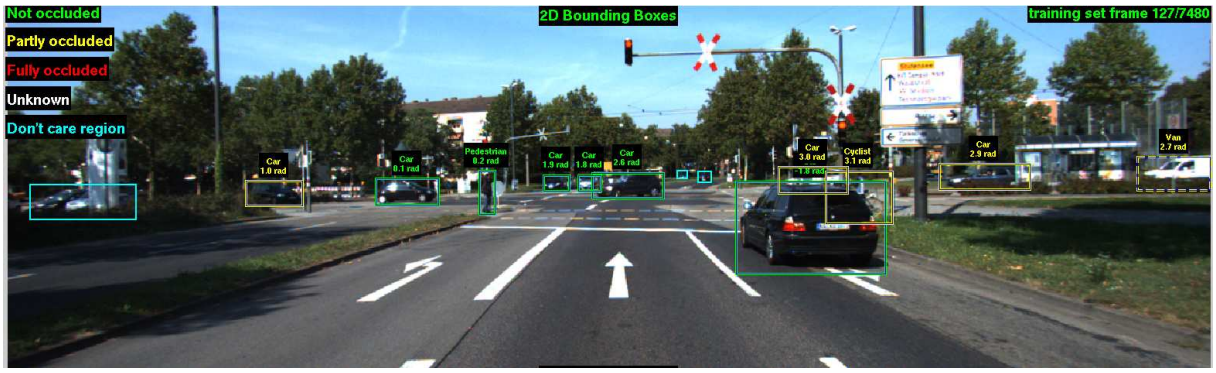


Figure 1.2: Current 3D scene understanding for autonomous vehicles. Ground truth labeled objects from [KITTI, 2012].

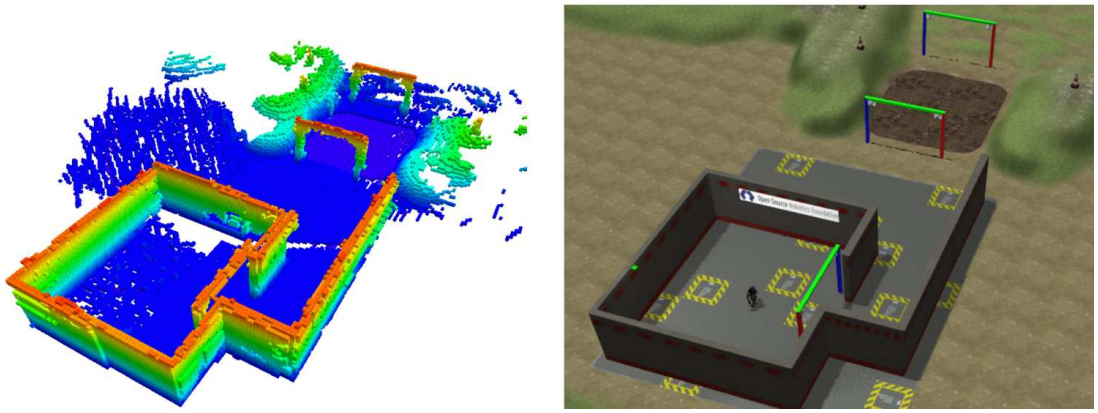


Figure 1.3: Current 3D scene understanding for humanoid robots. Sample scenario from the DARPA Virtual Robotics Challenge [Molinos et al., 2013].

The improvements in vision sensors, their price and size reduction, added to the progress in *Machine Learning (ML)* and *Computer Vision (CV)* approaches, have increased the appealing of vision systems to the industry and research community. In addition, providing 3D scene understanding and extracting useful semantic information from images require the employment of CV and ML advanced techniques, which allow to learn descriptive models from large datasets and to infer about objects and scene layout from new images. Two main branches can be distinguished[1]: **unsupervised** or **supervised** learning [Bishop, 2006]. The first one is able to cluster or to estimate the distribution density of the training samples without the need of manually labeled or annotated data. However, the supervised approach grants the definition

[1]There are also many approaches in the literature that are categorized outside of supervised and unsupervised methods. More specifically, they are usually framed under *reinforcement* and *semi-supervised* learning. However, they will not be treated in this Thesis.

of error measurements from the valuable annotations, favoring the learning of semantics from complex and real scenes, which added to the current availability of varied and large-enough annotated datasets, makes supervised learning an important field of research. This Thesis chooses this approach to solve object and scene layout prediction tasks. The section below gives a general introduction to ML concepts for supervised learning and inference from visual data.

1.1. Supervised Learning and Inference of semantic information

The general goal of image analysis and computer vision is to extract some interpretable or high-level information from matrices of numbers, which are the images. This yet undiscovered semantic information that has its origin in the 3D scenes captured by any imaging sensor, which is in turn approached by light rays traversing an optical lens, is in the form of pixels. Indeed, this is a nice idea for digitizing our human memories and surroundings with the aim of permanently saving them, preventing the risks of photo paper corruption along the time. However, this has opened infinite possibilities for image enhancing and "on purpose" picture modification with editing programs [Adobe, 2014]. This could be viewed as an initiatory supervised machine intelligence where specific algorithms equalize, blur, blend, stitch, filter, resize or, in general, retouch an image under the manual supervision of a human. However, the role of Machine Learning is beyond the previous idea and it is, actually, an horizontal discipline aimed at learning from data.

Teaching machines to differentiate semantic entities (e.g. scene layout and objects in our case) from a bunch of pixels is a state-of-the-art challenge [INRIA, 2012], which regards finding the appropriate representations, variables and structures to build a model relating the image data to the high-level information, i.e. road intersections layout, objects category, location and orientation in our case. The intermediate representations are commonly visual features in the form of engineered computations on the image pixels like color spaces, edges, gradients or more complex local and global descriptors [Lowe, 2004, Oliva and Torralba, 2001]. Typically, the variables can be separated in observed (image or visual features), output (e.g. object class) and auxiliary (sometimes latent or hidden), while their interactions are reflected in the model structure [Nowozin and Lampert, 2011].

Moreover, recognizing patterns in data can be done *supervisedly* when the image dataset is annotated such that the input variables (e.g. images) are related to the output variables (e.g. category) with known labels, i.e. the dataset is provided with a ground truth. On the contrary, *unsupervised* ML techniques cluster unlabeled samples in search of some dominant patterns or distributions in the data. Finally, there is also a *semi-supervised* strategy when there are some missing or noisy labels and a *reinforcement* learning where the algorithm searches for the optimal outputs depending on a set of defined rewards. [Duda et al., 2001].

Therefore, the supervised learning of parametric models, which is the case for the models presented in this Thesis (Chapters 3 and 4), consists in estimating and adjusting the values of a fixed number of parameters from labeled data, i.e. the KITTI image dataset [Geiger et al., 2012]. This process, also known as training the model, relies on pattern recognition and statistical tech-

niques [Bishop, 2006], which allows to learn complex relationships between the visual data and interest variables. In order to encode these interactions between observed and unknown variables, *Probabilistic Graphical Models (PGM)* provide a concise representation that could describe a family of probability distributions and their dependencies in a graph [Koller and Friedman, 2009]. More specifically, the methodologies proposed in this Thesis are based upon discrete models defined by undirected graphs (also known as *Markov Random Fields (MRFs)*) constructed by hand (which required deep knowledge and domain experts advice). The parameters of the structured model can be learned via *Conditional Random Fields (CRF)* or *Structured Support Vector Machines (SSVMs)* [Nowozin and Lampert, 2011, Hazan and Urtasun, 2010] through the definition of the corresponding loss functions and optimization algorithms. Moreover, these discriminative approaches make predictions based on modeling conditional distributions, which implicitly involves the need of taking decisions based on $P(Label|Data)$, as opposed to the generative models that try to learn the distribution $P(Data|Labels)$.

Once the model is defined and the parameters are learned, the inference (or also called testing) step consists in effectively predicting the labels for the output variables given new measurements or observed data. *Message Passing* [Bishop, 2006] and *Branch and Bound* [Lampert et al., 2009] are examples of approximate and exact inference algorithms that estimate the labels of the random variables defined in a graphical model. Hence, this grants to machines the capacity of extracting useful semantic information and 3D scene understanding from numeric matrices such as images.

Usually, the training is carried out as an offline process (but can be also approached as an incremental learning), while the testing can be done online. In fact, the ultimate goal is to learn a model that can be applied in real-time or interactive systems for the service robotics or the autonomous vehicles. This Thesis studies and proposes machine vision systems which can be integrated as market products in the near future. However, all the research work reported in this document has intensified on the supervised learning and inference techniques for object and scene layout prediction, leaving the technological transference to the application-oriented research projects and to the specific collaborations with industry.

1.2. Research questions

The aim of this section is to highlight and summarize the motivations, research hypotheses and approaches of this dissertation. First of all, Fig. 1.4 depicts the gist of this Thesis as a word map that presents the concepts introduced before. In fact, the research work in this Thesis contributes to the state of the art in 3D scene understanding for autonomous robotics platforms. Particularly, in two challenges: 1) the prediction of road intersections geometry and, 2) the detection and orientation estimation of cars, pedestrians and cyclists. For automatically labeling the road scene and the road users, the goal of this Thesis is to learn and infer this semantic information using different features extracted from stereo images (2.5D data) of the KITTI public urban dataset [Geiger et al., 2012]. Then, two methods for supervised learning of discriminative parametric models will be presented: CRF [Hazan and Urtasun, 2010] and *Discriminative*

Part-based Models (DPM) [Felzenszwalb et al., 2010b]. For inference, three methods will be employed: *distributed convex Belief Propagation (dcBP)* [Schwing et al., 2011], *Branch and Bound (BB)* [Lampert et al., 2009] and the DPM detector [Felzenszwalb et al., 2010b]. Moreover, the intersection and object models will be described as PGMs [Koller and Friedman, 2009] and they will be validated on real images employing a thorough set of experiments.

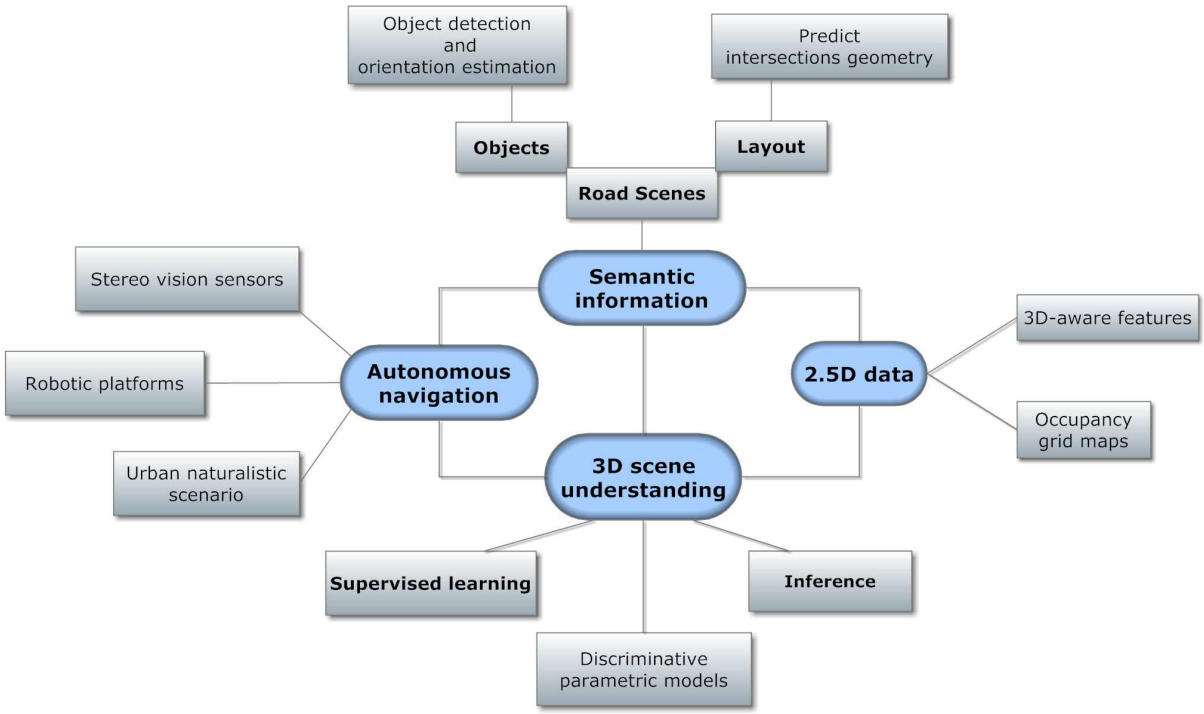


Figure 1.4: Word map depicting the gist of the Thesis.

In brief, the contributions of this Thesis are divided in two main blocks corresponding to the two challenges mentioned above.

1. Different parameterizations of discrete random variables are proposed to describe the geometry of intersections from bird's eye view images, i.e. occupancy grid maps from stereo sequences. Particularly, several discriminative models are studied for straight roads and 4 intersecting roads, which are an alternative to the seminar generative approach in [Geiger et al., 2011a]. The parameters of our graphical models are learned with CRFs and the labels from the random variables are predicted with dcBP and BB.

2. For the object detection and orientation estimation tasks, this Thesis proposes the first approach employing stereo data in the international challenge known as *KITTI evaluation benchmark* [KITTI, 2012]. The part-based object detector known as DPM is revisited and modified in order to learn richer models from 2.5D data (color and disparity). In particular, I) the DPM training pipeline is extended to account for 3D-aware features, II) a detailed analysis of the supervised parameter learning is provided and, III) two additional approaches are explored: "feature whitening" and "stereo consistency check". Additionally, a detailed review of the KITTI dataset statistics and the subtleties of the evaluation protocol are studied.

1.3. Organization of the Dissertation

The remainder of this dissertation is organized as follows: the state of the art is discussed in Chapter 2. Then, the supervised learning methodologies of the parametric models for object and scene layout inference are presented. In particular, Chapter 3 describes how to infer road intersections from bird's eye view perspective images formulating the problem as a structured prediction. This proposal is fully demonstrated on synthetic images with noise and the challenges of real data are also addressed. Chapter 4 studies the supervised learning of a state-of-the-art part-based object detector, contributing with new features based on 2.5D data (color and disparity) and the additional tunning of the training pipeline to favor better learned parameters. Next, in Chapter 5, the KITTI image dataset is reviewed and a large set of experimental results for the object detection and orientation estimation challenge are included, with a special focus on the evaluation protocol employed for comparison. Moreover, it shows the results published on the KITTI benchmark website [KITTI, 2012]. Finally, Chapter 6 concludes the dissertation with the main contributed approaches and results, as long as the identified future guidelines.

Chapter 2

State of the Art

Nowadays, **vision sensors** are employed **in automotive industry** to integrate advanced functionalities that assist humans while driving. During the last years, a big research effort has been made to design and study *Advanced Driver Assistance Systems (ADAS)* [González et al., 2013, Daza et al., 2014] and autonomous vehicles [Geiger et al., 2014] that rely on cameras as sensing technology and source of data [Sivaraman and Trivedi, 2013]. On the contrary, other sensing modalities as *Global Positioning System (GPS)*, *Light Detection And Ranging (LiDAR)* and *Radio Detection And Ranging (RaDAR)* have a well-established market as on board integrated systems for navigation, active safety and primary obstacles detectors [Lissel et al., 1994, Sukkarieh et al., 1999, Autoliv, 2014], yet information fusion is an open field of research [Matzka et al., 2012, Erbs et al., 2013]. In this context, the *Defense Advanced Research Projects Agency (DARPA)* Urban Challenge [Buehler et al., 2009] was a breakthrough for autonomous vehicles, in which the competitors (e.g. [Urmson et al., 2007, Montemerlo et al., 2008, Kammel et al., 2008]) based their systems on manually labeled maps, aerial imagery, GPS signal, RaDAR sensors and accurate and expensive LiDAR devices. However, vision cameras and image processing techniques were almost not employed for real obstacle detection and environment perception, relying on the data from several RaDAR and LiDAR sensors. This technological approach has also been followed by the well-known existing Google's car in order to accurately perceive the 3D environment, while an on board camera is in charge of traffic lights detection.

So far, **autonomous navigation** on highways and non-naturalistic pathways (lacking realistic numbers of vehicles, obstacles, pedestrians, urban structure, etc.) has been effectively demonstrated with the above DARPA-UC approaches. However, **urban scenarios** remain a big challenge due to their naturalistic complexity, GPS signal loss and the need of very accurate maps which are also impractical. Hence, providing image understanding capabilities to autonomous vehicles will be an important part of the solution in urban environments because cameras can capture the 3D scene as perceived by a driver, who performs a local navigation based on the scene features, while global waypoints can still be established by other navigation techniques. In addition, given this need for 3D scene understanding, there is an increasing interest on joint objects and scene labeling [Wojek and Schiele, 2008, Geiger et al., 2011b] in the form of geometry and semantic inference of the relevant entities contained in urban environ-

ments. Consequently, this Thesis tackles both challenges of object and scene layout estimation from images. More specifically, the inference of road intersections geometry and the detection and orientation estimation of cars, pedestrians and cyclists with respect to the vehicle.

With regard to **visual perception in robotics**, the tasks of autonomous navigation and recognition of places and objects are receiving a lot of funding and research interest around the world. In particular, the *DARPA Robotics Challenge (DRC)* [DARPA RC, 2013] is an open competition that concerns robot systems and software teams to build robots capable of assisting humans in responding to natural and man-made disasters. It is focused on the development of humanoid robots with the aim of operating in rough terrain and austere conditions, where many tasks has to be solved, but regarding to visual recognition, the following tasks can be pointed out: environment perception, mapping and navigation planning [Molinos et al., 2013]. In parallel, there is also a great interest in service robotics, e.g. to aid the visually impaired [Alcantarilla et al., 2012] and to track people indoors for robot-human interaction [Munaro and Menegatti, 2014]. All these robotic applications can be benefited from the theoretical and practical concepts presented in this Thesis. Indeed, they also require an image understanding of the surroundings and they pose similar challenges under the assumption of moving robotics platforms.

On the other hand, the improvements in camera features, their price and size reduction, added to the progress in machine learning and computer vision approaches for robotic platforms, have increased the appealing of vision systems to the service and military robotics, the automotive industry and the research community. Imaging devices provide a higher level of abstraction and semantic information more natural to interpret by humans compared to other sensors, e.g. guiding visually impaired users [Rodríguez et al., 2012], assisting in catastrophes [Molinos et al., 2013], intelligent parking [Toyota, 2014], lightbeam [Alcantarilla et al., 2011], autonomous vehicles [Geiger et al., 2014], etc.

The extraction of this high level information involves many **challenges on image scene understanding** in order to obtain more precise data that can leverage the autonomous driving platforms and assistance systems. These challenges may include and are not limited to object detection under occlusion [Pepik et al., 2013, Hejrati and Ramanan, 2012], estimation of objects orientation on 3D scenes [Pepik et al., 2012a], detection at far distances [Park et al., 2010], determining geometric layout of the scene [Wojek et al., 2013, Geiger et al., 2011b], dealing with varying illumination conditions [Milford and Wyeth, 2012], appropriate modeling and parametric learning of complex scenes [Zhu et al., 2012] and large-enough and naturalistic datasets.

As a matter of fact, a lot of research effort lies on the existence of **public datasets and common evaluation metrics** for advancing the performance of visual recognition systems. There are many benchmarks, some of them also widening to a higher number of categories non-restricted to road environments, like Caltech-101 [Fei-Fei et al., 2004], *Pattern Analysis, Statistical Modelling and Computational Learning (PASCAL) Visual Object Classes (VOC)* [PASCAL VOC, 2012], EPFL Multi-view car [Ozuysal et al., 2009], Middelbury stereo [Scharstein and Szeliski, 2001], ETH Multi-Person [Ess et al., 2008] and MIT-CSAIL LabelMe [Torralba et al., 2003], among others.

This Thesis employs the recent KITTI Vision Benchmark Suite [Geiger et al., 2012], which provides a wide set of video sequences and stereo images (in color and gray-scale) of road scenes captured from a vehicle in urban and inter-urban naturalistic environments. The images are split in different challenging benchmarks for the tasks of stereo reconstruction, optical flow, visual odometry/*Simultaneous Localization And Mapping (SLAM)*, object evaluation, object tracking and road segmentation. All datasets are provided with ground truth labeling plus common evaluation protocols to engage labs and researchers on pushing forward the state of the art in visual recognition systems for robotic applications and autonomous/intelligent vehicles. In addition, the KITTI autonomous driving platform provides positioning data from an inertial navigation system (GPS/IMU) and dense 3D point clouds from a Velodyne Laserscanner. They are used for the annotation of the images (object bounding boxes labeling, tracklets, loop closures and fine-grained odometry, etc.) and are also available to be employed in new systems and method proposals. However, the contribution of this Thesis is specifically on the use of stereo images for 3D scene understanding based on the supervised learning of complex models to infer objects and layout on road scenes.

In the next section, an overview of the computer vision trending topics is introduced, which is followed by the literature relating the two challenges addressed along the Thesis, i.e. scene layout inference (Section 2.2) and objects detection and orientation estimation (Section 2.3).

2.1. Trending topics in computer vision

Within the computer vision area, three trending topics can be identified from the state of the art of the *IEEE Conf. on Computer Vision and Pattern Recognition 2013*. This helps to depict a general view of the research work carried out in this Thesis encapsulated in the main vision branches under research and to set the basement for future guidelines.

1. **RGB-D sensor data** is currently the feed for many papers [Jiang and Xiao, 2013] in order to parse the images into its constituent objects, which is tightly related to the affordable Microsoft's Kinect [Microsoft, 2014] and Asus Xtion [Asus, 2014] color and depth sensors for indoor environments and the emergent *Time of Flight* cameras [Jiménez et al., 2012]. Outdoors, LiDAR [Velodyne, 2014] is still the most reliable device for 3D point cloud registering, but stereo vision cameras are a more cost-effective an easy to integrate solution. Therefore, this Thesis contributes to the current state of the art that relies on 2.5D data (color and disparity) obtained from stereo images and, particularly in this case, available in the public dataset [KITTI, 2012].

2. **Mid-level patch discovery** is a recent idea starting to be exploited with different approaches, which basically intends to discriminate image patches (e.g. object parts) that can be learned from large datasets, constraining the patches to be the most informative ones [Maji and Shakhnarovich, 2013]. This Thesis studies a parametric part-based model that is supervisedly learnt from a large naturalistic urban dataset for object detection and orientation estimation. However, future guidelines also point out to research proposals for unsupervised learning [Singh et al., 2012].

3. **Deep-learning and feature learning** are complex and highly dimensional problems being under active research in the computer vision community [Sermanet et al., 2013] and also supporting the unsupervised fashion of directly learning from readily-available unlabeled data. In fact, feature learning substitutes the common engineered visual features such as HOG [Dalal and Triggs, 2005], SIFT [Lowe, 2004], SURF [Bay et al., 2008] in search of automatic learned features that better describe the appearances in 3D scenes. This would be a natural line to evolve current versions of featured-based approaches as the ones presented in this Thesis.

2.2. Scene layout: inferring road intersections

Lane estimation and road detection for assisting drivers have been under active research during the last years [McCall and Trivedi, 2006, Danescu and Nedevschi, 2009], typically for lane departure warning. In fact, this is a mature research line that very recently has been successfully extended to onboard smartphones [Bergasa et al., 2014]. Beyond it, there has also been an increasing interest on automatically estimating the shape and geometry of intersections from moving vehicles with different approaches: detection and tracking based on a previously digitized map [Gengenbach et al., 1995], classification of intersections shape and road edges tracking employing color appearance [Rasmussen, 2003] and a more complete urban scene understanding from image superpixels [Ess et al., 2009].

The capability of detecting intersections in urban environments can lead the recognition of other semantic entities like pedestrian crossings, traffic lights, etc. Besides, intersections correspond to places where navigation decisions must be made, thus, it is of great interest for autonomous driving. [Geiger et al., 2011a] introduced a generative model for 3D urban scene understanding to estimate the geometry and topology of intersections and other traffic semantics. This approach employed a set of bird's eye view perspective images (or occupancy maps) built from stereo sequences captured by a moving vehicle in a mid-sized city of Germany. Static (occupancy grids from 3D voxelized scenes) and dynamic (3D flow vectors) features were used to learn non-parametric data distributions, which dependencies were defined with a Bayesian network. Alternatively, [Singh and Kosecká, 2012] presented a street intersection recognizer based on a boosting classifier where the feature vectors were defined as a normalized histogram of five pixel-wise labels from the urban scenes. In this case, the authors employed omnidirectional images making use of the different views to obtain several sources of appearance and geometry information of a given location.

Inspired by [Geiger et al., 2011a], we also support the idea of in-vehicle integration of the cost-effective stereo-vision sensors. Despite the higher noise and weaker depth information at larger distances compared to LiDAR, they are more appealing to industry. Hence, the inherited difficulties while employing stereo data have to be solved by applying probabilistic models that account for uncertainty and by the application of learning algorithms that capture patterns from stereo data. Inferring the intersections geometry from stereo-vision occupancy maps is a very challenging task due to the sparsity of the 3D maps reconstructed from

stereo video sequences and the uncertainty associated to the disparity measurements. However, occupancy grids have been largely used for obstacle detection and driving assistance in the context of robotics and autonomous navigation [Thrun, 2003], e.g. for estimating the free space ahead [Badino et al., 2008], for estimating the grid in combination with optical flow cues [Braillon et al., 2008] and to provide useful information for obstacle detection and also tracking [Nguyen et al., 2012] in urban environments.

On the other hand, 3D scene understanding has become a key component in autonomous driving and robotics applications with the aim of estimating the spatial layout in man-made environments [Tretyak et al., 2012]. In [Gupta et al., 2010], the urban scene layout is recovered through a set of 3D blocks with volume and mass that are governed by physical and geometric rules. This approach does not achieve a metric reconstruction but a qualitative and globally-consistent scene from single images, which provides a semantic understanding of the environment. In relation to indoor scenes, [Wang et al., 2010] tackles the room layout prediction in the form of faces (walls, floor, ceiling), adding some latent variables that account for the clutter usually present in the rooms (furniture, etc.). The problem is formulated as a discriminative structured prediction in which the 3D parametric box that best fits the room is selected from a discrete space of hypothesis. Actually, indoor scenarios can be more tightly constraint with the "box" idea [Hedau et al., 2010], which assumes the *Manhattan World* where three dominant and orthonormal vanishing points can be reliable determined. Hence, the room and objects faces are modeled as 3D boxes aligned with these dominant directions. Going beyond these works, [Gupta et al., 2011] proposes a proof-of-concept for the modeling of human-scene interactions, jointly estimating the scene geometry and human poses in workspaces.

Summarizing the above paragraph, this human-oriented viewpoint and the *Manhattan-world* assumption allow to infer additional semantics from daily environments considering monocular images and physical world constraints. Similarly, the goal of this Thesis is on providing semantic and geometric information to autonomous robotics platforms, but navigating in dynamic and naturalistic urban scenarios. We make use of the stereo information as the single source of data to predict the road intersections layout based on geometric constraints without the *Manhattan-world* assumption and without any map priors. In addition, we learn a parametric discriminative model (as opposed to the generative approach of [Geiger et al., 2011a]), inspired on the discriminative 3D indoor scene understanding of [Schwing et al., 2012] and the room faces of [Wang et al., 2010]. According to it, the intersection layout will be described as an undirected graphical model of discrete random variables, which poses a structured prediction problem where the parameters are learnt with a CRF [Hazan and Urtasun, 2010]. We also make use of the integral geometry idea [Schwing and Urtasun, 2012] to efficiently compute the features of the scene and to allow the decomposition of the CRF potentials. However, as the *Manhattan World* assumption does not hold, we define variables encoding the angles of the streets but also the intersecting points, because the vanishing points are unknown, as it will be explained in Chapter 3.

2.3. Object detection using Part-Based Models

Nowadays, 3D scene understanding is viewed as a joint problem where the prediction of scene structure and the semantic entities on it can be thought as a symbiosis, in which both inference problems can benefited from each other. Autonomous vehicles and personal robotics are the direct areas of application for indoor and outdoor environments [ICCV Workshop, 2013] and this Thesis intention is to also contribute to this research line. Previously, the main reference works on scene layout prediction have been reviewed, with the focus on the open problem of road intersections prediction. This section inspects the main related works on object detection and it is divided in three subsections: object detection and the part-based model employed as baseline; additional works extended to 3D reasoning and improved implementations and, finally, related literature on the use 2.5D data (color and disparity).

2.3.1. Object detection and DPM

Object detection from images has been under active research since 1970s and closely related to the beginning of computer vision a decade before. Actually, the early work on pictorial structures and spring-like object parts [Fischler and Elschlager, 1973] has motivated the flourishing of several approaches for visual recognition of objects that have been tested on a common and well-known benchmark [Everingham et al., 2010, PASCAL VOC, 2012]. For example, *Bag of Features (BoF)* approaches [Yebes et al., 2011] were fruitful multi-class categorization algorithms applied to natural scenes and intelligent vehicles. However, they had the shortcoming of not predicting the object location, despite the proposal of spatial pyramid matching [Lazebnik et al., 2006]. This was solved by another remarkable approach: Multiple Kernels [Vedaldi et al., 2009] also employing visual words, but combined with additional features and a three-stage classifier for locating and classifying objects. A different approach based upon compositional models [Malisiewicz et al., 2011] announced the novel *Exemplar-SVMs* trained from only one positive example and millions of negative ones. This work supported the idea of direct association between object detections and the training exemplar such that any meta-data attached to the exemplar could be transferred to obtain better scene understanding beyond the 2D bounding boxes.

However, given the strong research interest on the object classes 'Car', 'Pedestrian' and 'Cyclist' for autonomous navigation in urban areas [ICCV Workshop, 2013], the main challenge is at increasing the hit ratios and reducing false positives in complex, dynamic and naturalistic urban scenarios [KITTI, 2012]. Indeed, pedestrian detection has received a lot of attention in order to reduce road fatalities and it is an area with many contributed works during the last years [Dollár et al., 2012]. Despite the advances for this extent, the generalization towards recognizing wider sets of road participants (cars, trucks, cyclists, etc.) has motivated the application of successful object detectors to the road scenario [Geiger et al., 2011b], where LiDAR data can be used for labeling the ground truth and camera sensors can provide the input data for recognition, as they are cheaper and easier to integrate in the autonomous vehicles of the future.

Considering that, this Thesis contributes to the state of the art about Part-Based detectors by the addition of 2.5D sensor data (color and disparity) and tackles the object detection and orientation estimation challenge posed in [KITTI, 2012]. The baseline for comparison is the original *Discriminatively Trained Part-Based Models (DTPBM)* [Felzenszwalb et al., 2010b] or commonly referred to as DPM in the literature. It classifies and locates objects at different scales based on a pyramid of appearance features, i.e. a scale pyramid of modified *Histogram of Oriented Gradients (HOG)* descriptors [Dalal and Triggs, 2005] and it has been successfully tested on PASCAL challenges [PASCAL VOC, 2012] and applied to many other works and datasets, showing outstanding performance in 2D bounding boxes inference for object detection and classification, but also in segmentation, person layout and action classification tasks. Besides, it received a "Life Achievement Prize" for its contribution to community and its free distribution [Felzenszwalb et al., 2010a].

Regarding to the application of part-based models, [Geiger et al., 2011b, Geiger et al., 2012] made an adaptation of DPM for the object detection and evaluation challenge [KITTI, 2012]. Basically, they discretized the number of possible object orientations, i.e. 16 bins for cars, so that, every component of the mixture model corresponded to one orientation. Besides, they enlarged small examples by factor 3 and harvested random negatives from positive images, keeping for training only those negatives with a bounding box overlapping less than 20% with a positive label. Two versions (supervised and unsupervised) were reported on [KITTI, 2012] and they are the baseline precision-recall curves for the experiments in Chapter 5. The goal of this Thesis is to provide further evidence of the supervised learning with DPM while adding 2.5D cues from disparity maps, such that better precision-recall curves can be obtained.

In [López-Sastre et al., 2011], *mDPM* was evaluated for object category pose estimation where some supervised adaptations were proposed: fixing the latent component to the object pose available in the ground truth, removing bilateral symmetry and developing a modified training pipeline that regarded the coordinate descent algorithm and the selection of negatives examples from opposite views. Despite their improvement in orientation estimation tested in four different datasets, they only employed visual features from color images and KITTI could not be compared concerning the joint challenge on detection and orientation estimation. Thus, this Thesis provides results and a discussion applying some of the suggestions from [López-Sastre et al., 2011] while employing both color and disparity features on KITTI urban scene stereo images.

On the other hand, a new approach (OC-DPM) for explicit occlusion reasoning [Pepik et al., 2013] based on the DPM framework, has recently reported increased ratios, both in object detection and orientation estimation of cars [KITTI, 2012], but employing 12 viewpoints instead of 16. This is actually a very promising approach to overcome the missed detections and false positives of DPM over KITTI. However, despite the benefits of occlusion modeling, it is not yet clear whether the improvements came directly from it or due to the decreased number of viewpoints. A similar approach is also supported by [Hejrati and Ramanan, 2012] that proposed the detection and analysis of the geometric 3D configuration of objects in real-world images with heavy occlusion and clutter but tested on PASCAL cars [PASCAL VOC, 2012]. Their model reasons about 2D and 3D shapes and 3D viewpoints, which requires learning local,

global and relational structures and parameters, but again, based only on color images. One remarkable difference is that the space of object viewpoints is not discretized, as opposed to previous methods. Moreover, their approach relies on landmark appearance and global visibility changes.

2.3.2. Extensions of DPM: 3D reasoning and efficient implementations

Although the explicit 3D reasoning about objects and the computationally-efficient implementation of DPM are out of the scope of this Thesis, the following paragraphs introduce some noticeable proposals on these topics, which can inspire future guidelines.

Extending DPM to account for 3D peculiarities of the objects in the scene can leverage the semantic image understanding and it is, actually, a recurrent topic in computer vision and object modeling. However, very recently, several works have appeared in the literature that formulate varied solutions building over DPM. [Pepik et al., 2012b, Pepik et al., 2012a] have proposed an object detection and orientation estimation as a structured prediction problem that can learn from 3D geometric constraints with the support of *Computed-Aided Design (CAD)* models. Thanks to this synthetic data, the model is able to account for 3D deformable parts, whose locations are consistently estimated across viewpoints. However, the experiments do not show relevant improvements in the 2D bounding box inference and achieves state-of-the-art results in parallel tasks as wide-baseline matching, yielding more interesting contributions towards 3D object models, explicit 3D reasoning and fine-grained viewpoint estimation in these contexts.

More complex and futuristic methods have devised a higher level of abstraction, i.e. to include a 3D cuboid model [Fidler et al., 2012], in which DPM is extended in the features and filters size to learn objects 3D location and orientation from monocular images. The authors propose a deformable 3D cuboid composed of faces and parts, which are both subjected to deformations with respect to the anchors in the cuboid (stitching points). Alternatively, in [Wang et al., 2013], a joint object detection and occlusion reasoning approach is formulated as a novel structured Hough voting scheme for indoor environments extracting visual features from RGB-D data.

Differently, the reduction of the high computational requirements of DPM has been studied in [Kokkinos, 2012b, Kokkinos, 2012a, Kokkinos, 2011], which presented an efficient object detection with an algorithmically enhanced version of the objects image search inside DPM, comparing BB [Lampert et al., 2009] and Cascaded Detectors [Viola and Jones, 2001a]. Another efficient approach [Dubout and Fleuret, 2013] has described a general and exact method to speed up the DPM by a reduction of the computational cost of the convolutions between the model parameters and the features at multiple scales. In particular, the authors have proposed to use the properties of the Fourier transform demonstrating its effectiveness. Similarly, there is a very recent work on fast computation of feature pyramids [Dollár et al., 2014] reporting significant time savings during training with a slight decrease in detection performance. Certainly, in relation to hardware acceleration, [Hirabayashi et al., 2013] has implemented DPM on GPUs to parallelize and accelerate its computation for real-time applications, in which smartphones or other camera sensors can remotely connect to a cluster of servers for object detection.

Finally, due to the great activity in this research field, the DPM has been very recently outperformed by the *feature pooling* approach [Benenson et al., 2013], which relaxes the fixed cell griding of the original HOG proposal to learn more representative features from color and gradients. This work also involves a higher hypotheses space with a costly training process in memory, CPU and time. Besides, this is complemented in the paper with a thorough evaluation of feature selection, preprocessing and training methods, yielding impressive results compared to other state-of-the-art methods. Consequently, providing flexibility to features shapes is a promising approach according the results presented by Benenson et. al.

2.3.3. Using color and disparity. Features and approaches

DPM relies on the HOG visual features computed from color images to learn the appearance patterns of objects. These features have been widely employed since the seminar paper [Dalal and Triggs, 2005] in conjunction with a linear SVM classifier. The image patch under analysis is divided in squared cells of fixed size and a histogram of color gradients is calculated and normalized depending on neighboring blocks. Typically, the alternative visual features on image classification and image matching have been based upon the computation of keypoints and their associated descriptors [Lowe, 2004, Bay et al., 2008]. Since all these works, several extensions have been proposed showing improved results in several tasks, but in general terms, they can not be applied to every vision problem with a granted level of success. In fact, there is an open discussion on which approach would be the best one: better features and more data or better models and learning algorithms [INRIA, 2012, Zhu et al., 2012]. The experiments on this Thesis contribute to extract some useful knowledge in this regard.

In relation to HOG extensions, *3DHOG* [Kläser et al., 2008] was devised as a spatio-temporal descriptor based on HOG and the extraction of interest points with the *Harris3D* detector. The authors formulated the problem of binning 3D gradients using convex regular polyhedrons, applying the algorithm to action recognition in videos. Similarly, we will extend the HOG features to account for disparity gradients that can provide a 3D notion of the scene. However, we formulate an atemporal problem in 2D, based on the feature vector length and the model complexity rather than 3D vectors.

In line with the recent wave on feature learning, [Ren and Ramanan, 2013] proposes the new features called *Histogram of Sparse Codes (HSC)* that are not engineered as HOG, but learned from data. Dictionaries are learned from data with *K-SVD* and per-pixel sparse codes are aggregated to form local histograms. The authors pose the question on whether we may already be saturating the capacity of HOG and they show that HSC outperforms HOG on top of common baselines like DPM. However, this approach requires very large feature vectors (the original 32 of HOG vs 300 of HSC), which causes feasibility problems for training. In this Thesis, the longest extended HOG vectors that will be tested have 92 elements.

With the aim of combining color and disparity cues for pedestrian detection, [Walk et al., 2010] proposed the *HOS* and *DispStat* features. The first one based upon the *HOF-like* feature by [Rohrbach et al., 2009] extracted from the depth field, but in the case of *HOS*, it was directly computed on the disparity map. The second one (*DispStat*) based on a sim-

ple average of disparity values in a HOG cell. Their experiments showed a lower miss rate when concatenating several features (HOG + HOF + HOS + DispStat), which suggests the benefits of adding depth/disparity cues for the pedestrian detection problem. Inspired by this work, the application of longer features combining color and disparity have not been tested on more object classes. Hence, this Thesis extends the research to additional classes (cars and cyclists), for object location and orientation estimation and employing a larger dataset [Geiger et al., 2012].

A different fusion of intensity and depth [Makris et al., 2013] integrates both sources of data in a probabilistic framework during detection. The whole approach is based on a dictionary (or BoF) of SIFT local features and a part-based model. Although the work shows comparative performance values against DPM, the gains for occluded vehicles are minor and the precision decreases for unoccluded ones. However, compared to the same approach without disparity features, the main conclusion is that stereo measurements help to discard object parts out of the vehicle's depth vicinity. This matches our intuition of the possible benefits while adding disparity features to DPM.

Moreover, the use of disparity as an additional feature in outdoor environments can provide enriched measurements, but also carry uncertainty (larger at farther distances) and outliers (due to occlusion, disparity estimation errors and the sparsity of the disparity maps). Several methods have been proposed and are under research for a fine-grained and reliable disparity map computation [Mattoccia, 2013], which usually include a left-right consistency check when matching features or blocks between the left and right frames of the stereo rig. This opens the question of whether a "married matching" based on epipolar geometry constraints can yield lower number of false positives that improves the object detector performance. Consequently, this Thesis carries out a set of experiments on the left-right simultaneous detection idea.

In relation to previous proposal, [Bao et al., 2012] introduced the object co-detection to match an object instance in multiple views by measuring appearance consistency between objects parts and geometry of the same object in different images, but no disparity information was employed, though. For the specific task of object detection using stereo images, the results published in that work show increased average precision compared to DPM for the tested datasets. Another related approach, presented the concept of 3D scene-consistency for the stereo matching [Bleyer et al., 2012], applied to the pixel-wise labeling task (or object segmentation) in the PASCAL challenge [PASCAL VOC, 2012]. Basically, it jointly estimated objects labels and disparities by minimizing a defined energy function given some physical constraints, which accounted for 3D reasoning. As can be seen, mixing color and disparity appearance for object recognition is a topic of a great interest and in this case, constraining to 3D scene physical rules. Similarly, we will make use of the epipolar geometry for 3D scene-consistency.

An even greedier approach [Roig et al., 2011] proposed a multi-object detection for a multi-camera system with consistency checks between views, in which a CRF was defined to model the objects occlusions and interactions between views, while the underlying class detectors were based on DPM [Felzenszwalb et al., 2010b]. Although tested on urban environments for the same object classes contained in this Thesis, the fixation of the cameras in urban structure provided a background subtraction prior, helping the inference tasks. On the contrary, inferring objects from moving platforms involves changing background, which increases the complexity.

Finally, the scaling of disparity values to accommodate to the objects height variances depending on their depth in the scene, is also an important issue pointed in [Walk et al., 2010, Helmer and Lowe, 2010]. In the first citation, disparity is scaled depending on the ratio of the current object hypothesis and a reference height for the object class. The latter work is applied to indoor environments to detect small objects for mobile robotics. It employs a disparity prior that accounts for depth and scale agreement of the bounding box, leading to a reduction in the false positives and increased scores for the correct detections. Following the same considerations when computing the disparity features proposed in this Thesis, we also demonstrate the benefits of adding this scaling to increase object detection ratios.

2.4. Overview

On the next pages, three tables provide schematic details of the main literature reviewed along this Chapter. More specifically, Table 2.1 summarizes the state-of-the-art works in autonomous navigation and recent ADAS systems, which are the motivation for this dissertation. The table indicates a variety of sensors employed in different state-of-the-art systems and autonomous vehicles. In the second column ('vision'), *several* refers to distinct cameras around the vehicle, while *mono* and *stereo* are related to monocular or stereo vision systems without differentiation between grayscale or color modalities.

Next, Table 2.2 depicts indoor and outdoor scene understanding approaches from vision sensors, in relation to our urban scene layout prediction of road intersections. The approaches are categorized as generative, discriminative or alternative non-probabilistic methodologies. Besides, the learning and infering algorithms are also indicated on the table.

Finally, Table 2.3 gathers related works for the object detection and orientation estimation challenge [KITTI, 2012] in three main aspects: extensions and other approaches on the successful DPM part-based detector, complex 3D object reasoning and the addition of disparity data for image understanding. Besides, due to the large variability of approaches, validation datasets and features, a set of brief comments have been attached to the table to highlight the main characteristics of every work.

Table 2.1: Brief review of the state of the art in ADAS and autonomous navigation

Reference	Vision	GPS	IMU	LiDAR	RaDAR	SoNAR	Observations
[Lissel et al., 1994]	no	no	no	no	yes	no	Car applications with RaDAR sensors
[Sukkarieh et al., 1999]	no	yes	yes	no	no	no	Sensors fusion applied to autonomous land vehicles
[Sotelo et al., 2006]	stereo	no	no	no	no	no	ADAS. Pedestrian detection from stereo images
[Urmson et al., 2007]	stereo	yes	yes	yes	yes	no	*Tartan racing* team entry in the DARPA-UC
[Montemerlo et al., 2008]	stereo	yes	yes	yes	yes	no	*Stanford racing* team entry in the DARPA-UC
[Kammel et al., 2008]	no	yes	yes	yes	no	no	*AnnieWAY's* team entry in the DARPA-UC
[Buehler et al., 2009]	stereo	yes	yes	yes	yes	yes	Book of the 2007 DARPA Urban challenge
[Google, 2011]	stereo	yes	yes	yes	yes	no	Google autonomous car
[Matzka et al., 2012]	stereo	yes	yes	yes	yes	yes	Sensor fusion for automotive vision systems
[Geiger et al., 2012]	stereo	yes	yes	yes	no	no	KITTI vision benchmark
[Dollár et al., 2012]	mono	no	no	no	no	no	ADAS. Pedestrian detection survey
[MRG Oxford, 2012]	several	yes	yes	yes	yes	no	RobotCar autonomous and electric vehicle
[Erbs et al., 2013]	stereo	no	no	no	yes	no	Fusion of image, depth and radar data for semantic segmentation
[Sivaraman and Trivedi, 2013]	several	no	no	no	no	no	Review on recent monocular and stereo detection systems from vehicles in road scenarios
[González et al., 2013]	mono	yes	no	no	no	no	ADAS. Traffic panels recognition
[Broggi et al., 2013]	several	yes	yes	yes	no	no	Braive autonomous car
[Daza et al., 2014]	mono	yes	no	no	no	no	ADAS. Fusion of indicators for driver drowsiness detection
[Geiger et al., 2014]	stereo	yes	yes	yes	no	no	3D urban scene understanding
[Autoliv, 2014]	no	no	no	no	yes	no	ADAS. Commercial system for obstacle detection
[BMW, 2014]	several	yes	no	no	yes	no	ADAS. BMW Intelligent parking, driving and vision commercial systems
[Toyota, 2014]	mono	no	no	no	yes	yes	ADAS. Toyota Intelligent parking also based in ultrasonic sensors

Table 2.2: Summary of the indoor and outdoor scene layout prediction approaches

Reference	Scene	Data	Approach	Learning	Inference	Observations
[Gengenbach et al., 1995]	outdoor	mono	Kalman tracking	-	-	Recognition of intersections and lane structures
[Rasmussen, 2003]	outdoor	polycam	discriminative	SVM	SVM	Road intersections detection via shape analysis
[McCall and Trivedi, 2006]	outdoor	several sensors	feature extraction and tracking	-	-	Road lanes detection-and-tracking
[Badino et al., 2008]	outdoor	stereo	generative	distributions modeling	dynamic programming	Estimation of navigable space from stochastic occupancy grids
[Danescu and Nedevschi, 2009	outdoor	stereo	particle filter	-	-	Probabilistic lane tracking
[Ess et al., 2009]	outdoor	mono	discriminative	AdaBoost	Boosting classifier	Urban scene classification: road users and layouts based on superpixels analysis
[Gupta et al., 2010]	outdoor	mono	generative	-	interpretation by synthesis	Urban blocks governed by physical and geometric constraints
[Wang et al., 2010]	indoor	mono	discriminative	SSVM	ICM	Room layout prediction of the walls and furniture faces in cluttered scenes. Based on vanishing points
[Hedau et al., 2010]	indoor	mono	generative	logistic regression	exact	Room geometry with 'Manhattan world' assumption
[Geiger et al., 2011a]	outdoor	stereo	generative	distributions modeling	MCMC	Intersections geometry and topology with a complex probabilistic framework
[Singh and Kosecká, 2012]	outdoor	omnidir.	discriminative	boosting	boosting	Semantic urban layout and streets intersection classification
[Tretyak et al., 2012]	outdoor	mono	discriminative	Probabilistic Hough	discrete approximation	Estimation of spatial layout in man-made environments
[Schwing et al., 2012]	indoor	mono	discriminative	SSVM, CRF	dcBP	3D indoor scene understanding with structured prediction
[Schwing and Urtasun, 2012]	indoor	mono	discriminative	SSVM, CRF	BB	3D indoor scene understanding with exact inference
[Bergasa et al., 2014]	outdoor	several sensors	events detection and tracking	-	-	Lane departure warning on an on-board smartphone
[Geiger et al., 2014]	outdoor	stereo	generative	distributions modeling	MCMC	3D outdoor scene understanding. Joint scene and objects labeling
ours	**outdoor**	**stereo**	**discriminative**	**CRF**	**dcBP, BB**	**Urban scene layout. Inferring road intersections**

Table 2.3: Summary of the literature in object detection and related approaches

Reference	Data	Method	Objects	Dataset	Observations
[Fischler and Elschlager, 1973	mono	Pictorial structures and dynamic programming	faces	own	First proposal of visual recognition with object parts connected by 'springs'
[Lazebnik et al., 2006]	color, weak features, SIFT	Spatial Pyramid Matching	several	Caltech, others	Object classification and location estimation
[Kläser et al., 2008]	color, 3DHOG	spatio-temporal descriptors from 3D gradients	actions	video datasets	Action recognition in videos with the 3DHOG descriptor
[Vedaldi et al., 2009]	color, SIFT, others	linear and non-linear kernel SVMs	several	PASCAL	Object classification with visual words and a 3-stage classifier
[Everingham et al., 2010]	color	Visual Object Classes challenge	several	PASCAL	Object recognition benchmark
[Walk et al., 2010]	stereo color, HOG, HOF, HOS, DispStat	Color and disparity features and combining different classifiers	pedestrian	ETH, TUD	New features with color and disparity plus classifiers combination for pedestrian detection
[Malisiewicz et al., 2011]	color, HOG	Exemplar SVMs	several	PASCAL	Object detection training with a single positive sample. Meta-data for scene understanding
[Yebes et al., 2011]	color, SIFT, SURF	Bag of Features	several	Caltech, others	Multiple object classification applied to occupants monitoring
[López-Sastre et al., 2011]	color, HOG	mDPM	car	ICARO, EPFL, PASCAL	Object category pose estimation
[Geiger et al., 2012]	stereo color	KITTI Vision Benchmark	car, pedestrian, cyclist	KITTI	Complex, dynamic and naturalistic urban scenes
[Dollár et al., 2012]	color	Survey of state-of-the-art in pedestrian detection	pedestrian	several	ADAS. Pedestrian detection survey
[Bleyer et al., 2012]	stereo color	3D-scene consistency and stereo matching with GMMs	scene	Middlebury	Joint pixelwise object segmentation and depth estimation
[Pepik et al., 2012a]	color, HOG	3D deformable part models, CRFs	car	PASCAL, EPFL	3D CAD models of objects and geometric constraints
[Hejrati and Ramanan, 2012]	color, HOG	DPM, MRFs and 3D models	car	PASCAL	3D objects shape and geometry from monocular images. Continuous objects viewpoint space
[Kokkinos, 2012a]	color, HOG	DPM, Branch&Bound, Cascade detectors	several	PASCAL	Efficient implementation of the DPM detection stage
[Dubout and Fleuret, 2013]	color, HOG	Accelerated training of DPM	several	PASCAL	Reformulation of the convolution between features and model, based on Fourier properties
[Ren and Ramanan, 2013]	color, HSC	High-dimensional features learned with K-SVD	several	PASCAL, INRIA	HSC features automatically learned from data and applied to people detection
[Pepik et al., 2013]	color, HOG	OC-DPM	car	KITTI	Modeling occlusion patterns, but employing a lower number of objects viewpoints
[ICCV Workshop, 2013]	stereo color	Reconstruction Meets Recognition Challenge	several	RMRC	Indoors and outdoors datasets
[Behley et al., 2013]	laser	Mixture of Bag-of-Words	car, pedestrian, cyclist	KITTI	Object detection only employing laser data
[Geiger et al., 2014]	color, HOG	LSVM-MDPM	car, pedestrian cyclist	KITTI	3D outdoor scene understanding based on modified DPM
ours	**stereo, 3D-aware features**	**Supervised learning of DPM adaptations, 2.5D data, whitening and stereo consistency**	**car, pedestrian, cyclist**	**KITTI**	**Object detection and orientation estimation from naturalistic road scenes**

Chapter 3

A discriminative model for learning road scene layout

Providing image understanding capabilities to autonomous vehicles in urban environments is challenging [Pepik et al., 2013, Wojek et al., 2013]. However, cameras can capture the 3D scene as perceived by a driver, who performs a local navigation based on the scene features. In addition, given this need for 3D scene understanding, there is an increasing interest on joint objects and scene labeling [Geiger et al., 2014] in the form of geometry and semantic inference of the relevant entities contained in urban environments. Chapter 4 will describe the object detection while this chapter describes the Probabilistic Graphical Models (PGM) and Machine Learning (ML) techniques for supervised learning and inference of intersections layout. Assuming that the topology is known, i.e. the number of intersecting roads, the geometry of straight roads and 4 intersectings roads are predicted from stereo data captured by a moving vehicle.

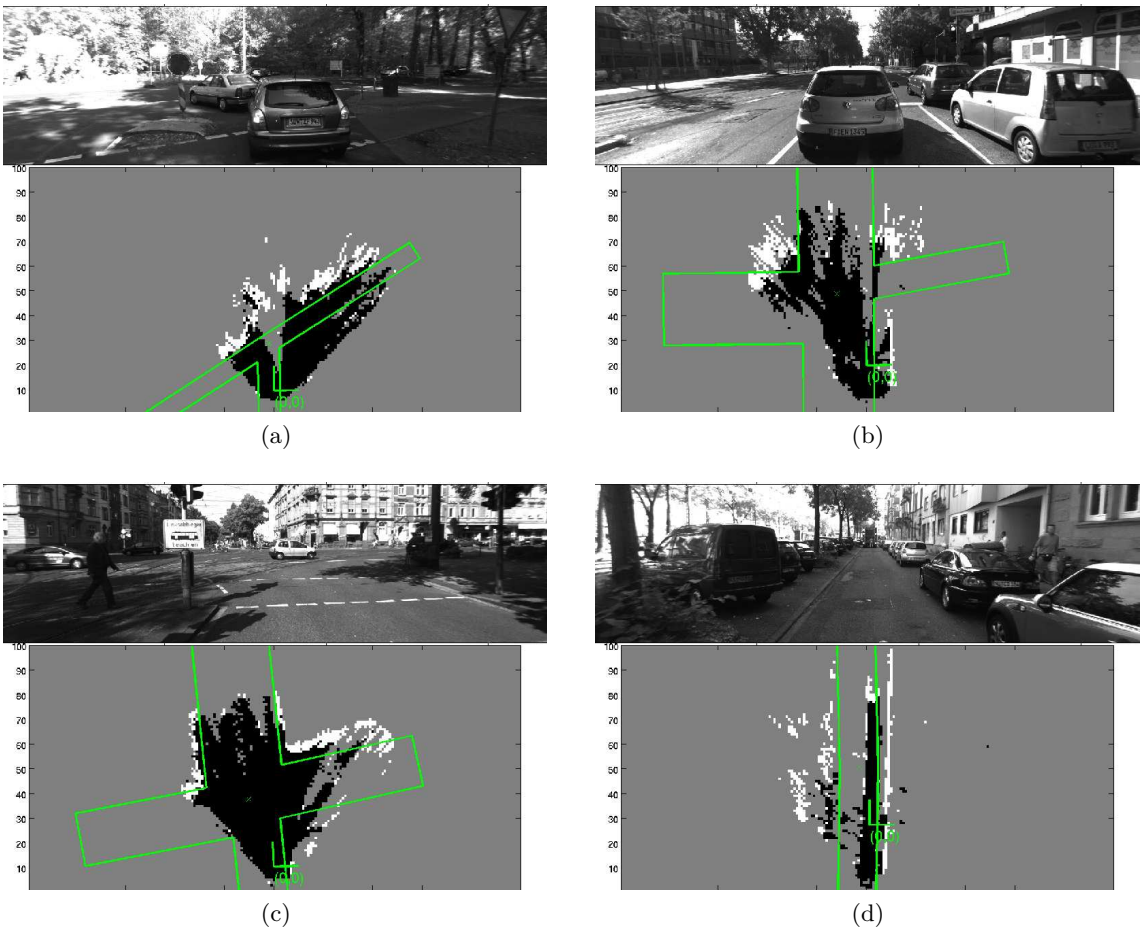
Firstly, the introduction depicts a general view of the approach including the problem description, general formulation and the specific goals to achieve. Then, the models for the different intersection topologies are proposed and defined in detail. The last section provides experimental results for the different models using synthetic and real images.

3.1. Introduction

As reviewed in Chapter 2, lane estimation and road detection for lane departure warning have been under active research during the last years [McCall and Trivedi, 2006, Danescu and Nedevschi, 2009]. Beyond it, there has also been an increasing interest on automatically estimating the shape and geometry of intersections from moving vehicles [Ess et al., 2009, Singh and Kosecká, 2012]. Similarly, service robotics can also be benefited from the estimation of the spatial layout in man-made environments [Tretyak et al., 2012], but in this case relying on the *Manhattan world* assumption, which defines the two or three dominant vanishing points of an outdoor or indoor scene, respectively.

3.1.1. Problem description

Let us consider a moving observer (e.g. robot, mobile platform, vehicle) with a stereo camera on it, which is navigating through structured and non-open spaces, such as streets inside cities (or corridors within buildings). The images from the camera are processed by the *Efficient Large-Scale Stereo Matching (ELAS)* library [Geiger et al., 2010], which reconstructs and voxelizes the 3D scene in order to obtain the 2D occupancy grids of the environment. This 2D map is a *Bird's eye Perspective (BeP)* image of the local surroundings in front of the observer, which is built from a sequence of frames of variable length. The grids of this occupancy map are referenced to the first frame of the sequence and their metric equivalence to the 3D world is reported in [Geiger et al., 2011a]. Fig. 3.1 displays four examples of these occupancy maps together with the last image frame in grayscale. The white grids indicate occupied areas, the black ones correspond to unoccupied or free space and the gray ones are just unobserved samples. This approach assumes that urban structure (buildings, parked cars, vegetation, etc.) are obstacles bounding the free space that could be traversed, thus, they are represented in white. Besides, the lines in green reproduce the ground-truth location of the roads boundaries which define rectilinear intersections. Finally, the unitary axes marked as $(0, 0)$ are the reference coordinate system for the last frame of the sequence, because internally, the model of [Geiger et al., 2011a] is represented with respect to this frame.



(a) (b) (c) (d)

Figure 3.1: Examples of bird's eye perspective images from [Geiger et al., 2011a].

As it can be seen from the BeP images above, the prediction of the green-labeled intersections from the occupancy maps is very challenging due to the sparsity of the stereo reconstruction (figures 3.1.a,b,c,d); the additional noise because of moving objects (figure 3.1.b) or vegetation between opposite street lanes (figure 3.1.d); the intrinsic uncertainty of disparity measurements at farther distances and the lack of scene structure that bounds the streets (figure 3.1.a).

Therefore, learning a parametric model of the roads layout from the 113 provided sequences [Geiger et al., 2011a] must be supported with tight physical constraints of the intersection geometry and powerful statistical algorithms that can work under high level of noise. Moreover, our proposed layout inference will be based upon grid map features counting, without any prior map of the environment, thus, facing a pure stereo-vision challenge.

3.1.2. Problem formulation: structured prediction and CRF

In order to predict the intersections layout of outdoor urban scenes from a single BeP image, the layout will be represented in terms of several discrete random variables, which capture the geometry of the intersecting streets. These variables are not independent, so that we have a structured prediction problem [Nowozin and Lampert, 2011]. This means that, as opposed to a classification problem in which one individual variable (i.e. the category) can be predicted, a structured prediction algorithm will jointly infer about the values of the unknown variables, which are related to each other. Besides, these relationships will be properly represented with Probabilistic Graphical Models [Koller and Friedman, 2009] in the next sections.

In general terms, we will define a graphical model with a parameterized conditional probability distribution that has the form of a "Boltzmann distribution", which is usually known as a *log-linear model* in ML community:

$$p(y|x,\boldsymbol{\theta}) = \frac{1}{Z_p(x,\boldsymbol{\theta})} exp(-E(x,y,\boldsymbol{\theta})) \tag{3.1}$$

where $y \in \mathcal{Y}$ corresponds to the structured labels defining the model, $x \in \mathcal{X}$ are the observed objects and $\boldsymbol{\theta}$ are the parameters or weight vector to be learned. The quantity $Z_p(x,\boldsymbol{\theta})$ is usually called the *partition function* and it has normalization purposes being a summation over the possible configurations of y. Lastly, E is the energy function, which is linear dependent on $\boldsymbol{\theta}$ and can be decomposed as $E(x,y,\boldsymbol{\theta}) = \boldsymbol{\theta}^T \cdot \phi(x,y)$, such that the potential $\phi : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^d$ maps the (x,y) pairs to multidimensional feature vectors.

For **structured prediction**, i.e. for predicting the values of the random variables y that define our desired output (e.g. intersection layout), the goal is to solve the next inference task employing the parameters $\hat{\boldsymbol{\theta}}$, which have to be previously learned.

$$\hat{y} = \underset{y}{argmax} E = \underset{y}{argmax} \left\{ \hat{\boldsymbol{\theta}}^T \cdot \phi(x,y) \right\} \tag{3.2}$$

Every specific problem requires a particular description of the potentials $\phi(x,y)$, which can be any function of any number of variables such that the order of each potential is given by this number. Besides, the cardinality of each discrete variable defines the size of every potential

function. For example, given 3 discrete random variables of 20 labels each one, the total size would be 20^3. On the other hand, to have feasible learning and inference procedures, it is desirable to design low-order potentials (tuples of two or three discrete variables). Sometimes, the problem definition imposes high-order functions that need to be reduced in size by decreasing the number of states of each variable. Consequently, this limitates the search space and causes discretization artifacts [Wang et al., 2010]. Differently, the problem can be decomposed in a sum of any number of functions of low order without reducing the search space [Schwing et al., 2012]. We follow the latter approach, which favors faster and more accurate learning and inference tasks. However, this decomposition is not trivial as we will show in the next sections for the specific case of road intersections modeling.

As stated before, we will make use of PGMs to describe the structure of the problem. Then, every potential can be represented by a factor graph (e.g. Fig. 3.4), which is a *bipartite* graph that encodes the relationships between the random variables $y_k \in \mathcal{Y}$ (rounded nodes in the graph) linked to the corresponding feature or potential (square nodes that are called factors) [Koller and Friedman, 2009].

To sum up, building the potentials is equivalent to the generation of the hypotheses space where inference will find the solution with the maximum energy (Eq. 3.2). In particular, each potential is restricted to the existence domains of the random variables linked to the factor representing that potential. Thus, it is a subset of the whole space. In general, this inference task can not be solved by brute-force analysis[1], which consists in exhaustively computing all combinations to search for the global maximum. In fact, it is an *Non-deterministic Polynomial-time hard (NP-hard)* problem that can be approached with different algorithms [Koller and Friedman, 2009]. In this Thesis, it will be solved with an efficient approximate message-passing algorithm [Schwing et al., 2011] based on *Belief Propagation* [Bishop, 2006]. Moreover, we will also employ a BB strategy which relies on the *Efficient Subwindow Search (ESS)* by [Lampert et al., 2009] and the exact inference solution of [Schwing and Urtasun, 2012].

For **parameter learning**, the objective is to calculate the vector $\hat{\boldsymbol{\theta}}$ that makes $p(y|x, \hat{\boldsymbol{\theta}})$ as close to an hypothetical true conditional distribution $tcd(y|x)$ as possible and, also, generalizes well to unseen data (x). Although this distribution is not known, given the annotated labels of y from the training dataset we can approximate it. Typically, the regularized conditional likelihood maximization is employed for the probabilistic training, which means that our discriminative parametric model will be learned with Conditional Random Fields [Hazan and Urtasun, 2010]. This differs from the original generative proposal of [Geiger et al., 2011a]. Hence, we are only interested in modeling the distribution of the conditional probability of all the variables y given the observations x. The notation that introduces the probabilistic training with CRFs is formally stated in Equations 3.3 to 3.5.

Firstly, let us consider the observed data (occupancy grids) vectors $\mathbf{x}$ and the output vector of random variables $\mathbf{y}$. Besides, let us define the conditional probability as the product of unitary

[1]The feasibility of the brute-force solution depends on the size of the graphical model and the size of the random variables. While doing this statement, we assume complex problems that can not be solved using brute-force, or would required many resources.

ψ_i (dependence on only one random variable) and higher order ψ_α terms (dependence on several random variables) in Eq. 3.3.

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_p(\mathbf{x},\mathbf{y})} \prod_i \psi_i(y_i,\mathbf{x}) \prod_\alpha \psi_\alpha(\mathbf{y}_\alpha,\mathbf{x}) \tag{3.3}$$

$$Z_p(\mathbf{x},\mathbf{y}) = \sum_{\mathbf{y}} \psi(\mathbf{y},\mathbf{x}) \tag{3.4}$$

$$\psi(\mathbf{y},\mathbf{x}) = exp\left\{-\boldsymbol{\theta}^T \cdot \phi(\mathbf{y},\mathbf{x})\right\} \tag{3.5}$$

Comparing the equations 3.1 and 3.5, the term $\boldsymbol{\theta}^T \cdot \phi(\mathbf{y},\mathbf{x})$ is the energy function E that was previously defined. Then, the conditional probability can be rewritten as a summation on the exponential:

$$p(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) = \frac{1}{Z_p(\mathbf{x},\mathbf{y})} exp - \left\{\sum_i \boldsymbol{\theta}_i^T \cdot \phi(y_i,\mathbf{x}) + \sum_\alpha \boldsymbol{\theta}_\alpha^T \cdot \phi(\mathbf{y}_\alpha,\mathbf{x})\right\} \tag{3.6}$$

To fully characterize this conditional probability given a training dataset Ds with pairs $(\mathbf{x}_k,\mathbf{y}_k)$, we need to estimate the parameter vector $\hat{\boldsymbol{\theta}}$. This is carried out employing the regularized conditional likelihood maximization in Equations 3.7 to 3.9, which define the soft-max function proposed by [Hazan and Urtasun, 2010].

$$\hat{\boldsymbol{\theta}} = \underset{\theta}{argmin} \left\{\sum_{Ds} lnZ_\epsilon(\mathbf{x}_k,\mathbf{y}_k) - \boldsymbol{\theta}^T \cdot \mathbf{d} + \frac{C}{p} \cdot \parallel \boldsymbol{\theta} \parallel_p^p\right\} \tag{3.7}$$

$$\mathbf{d} = \sum_{Ds} \phi(\mathbf{y}_k,\mathbf{x}_k) \tag{3.8}$$

$$lnZ_\epsilon(\mathbf{x}_k,\mathbf{y}_k) = \epsilon \cdot ln \sum_{\hat{y}\in\mathcal{Y}} exp\left(\frac{\Delta(\mathbf{y},\hat{\mathbf{y}}) + \boldsymbol{\theta}^T\phi(\mathbf{y},\mathbf{x})}{\epsilon}\right) \tag{3.9}$$

In the equations above, Z_ϵ is the soft-max partition function, which extends the partition function of log-linear models to norms with the aim of preventing overfitting. The vector $\mathbf{d}$ corresponds to the empirical means, C and p are constants to control the regularization term and $\Delta(\mathbf{y},\hat{\mathbf{y}})$ is the loss function that quantifies the error between predicted and ground-truth labels. Lastly, the parameter ϵ performs the commutation between CRF and SSVMs as described in [Hazan and Urtasun, 2010]. In particular, for CRFs training, $\epsilon = 1$.

3.1.3. Goals

Once the problem has been introduced and formulated in terms of structured prediction theory and CRFs in particular, we enumerate below the main goals of this chapter with a top-down view of the scene layout challenge.

- Learn and infer the geometric layout of road intersections from stereo-camera sequences captured by a moving vehicle.

- Implement and test a discriminative approach for this task, as an alternative to the seminar generative approach in [Geiger et al., 2011a].
- Assuming the rectilinear intersection topology is known, define a parameterization and a graphical model for straight and 4 intersecting roads.
- Compute the potentials of the proposed factor graphs given the parameterization and the observed data in the form of 2D occupancy grids.
- Define a loss function, learn the parameters for each topology and perform inference tests on synthetic data built from ground truth and also on the challenging real data.
- Discuss the main conclusions and derive further considerations after the experiments.

3.2. Modelling 1 straight road.

Let us consider a basic case where there is only one straight road contained in the BeP image as can be seen in Fig. 3.2. This means that the road segment, where the ego-vehicle is driving through, is prolongated and no crossing streets are measured.

(a) Last frame of the sequence (b) Measured 2D grid map

Figure 3.2: Last frame and BeP images of a straight road sample [Geiger et al., 2011a].

Due to the process of BeP creation, naturally, the starting position of the ego-vehicle path will always be on the bottom boundary of the image. This constraints the hypotheses space, i.e. there are no horizontal roads crossing the BeP images, and simplifies the definition of the model for 1 straight road. Considering this premise, 3 random variables are proposed that fully describe the road in the BeP. Assuming that street line boundaries are parallel, the three random variables are the fixation points p_1 and p_2 and the angle α, as it is depicted in Fig. 3.3. In general, the domain of p_1 and p_2 is not limited to the image width and they can also have negative values, which is denoted by the dashed line on the bottom part of the figure. This will allow to describe roads that do not start on image bottom, but they cut left or right BeP image sides. However, in the specific case of 1-straight road, the domain is restricted to a narrower range because of the presence of the ego-vehicle in the street.

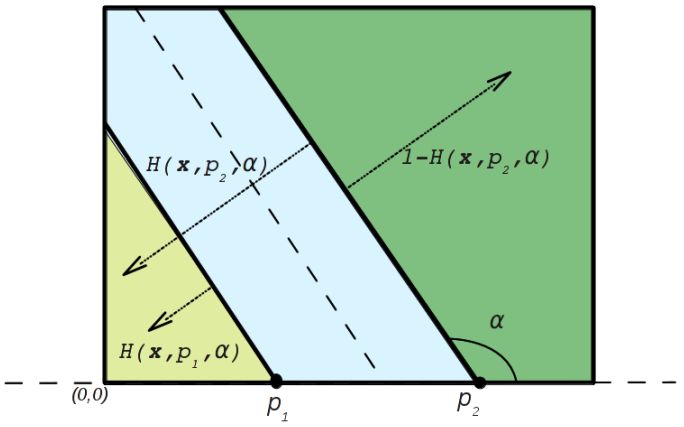


Figure 3.3: Parameterization for 1 straight road in the BeP and pictorial representation of the proposed feature function $H(\mathbf{x}, p_i, \alpha)$ with the domain areas inside the grid map. We assume that the features are normalized over the whole image dividing by the total number of grids of each type. Thus, the green region corresponds to the subtraction $1 - H(\mathbf{x}, p_2, \alpha)$.

As stated in Section 3.1.2, we formulate the road layout inference as a structured prediction problem, in which the potentials of Eq. 3.2 have to be defined. Our proposed model for 1 straight road is represented by the factor graph in Fig. 3.4, which implicitly shows the dependencies between the random variables and the factorization of the conditional probability previously defined in Eq. 3.6. As can be observed, we only propose to use pairwise potentials and they are mathematically defined in Eq. 3.10. Three of them account for visual features on the BeP image and the latter one corresponds to a constraint that enforces $p_1 < p_2$.

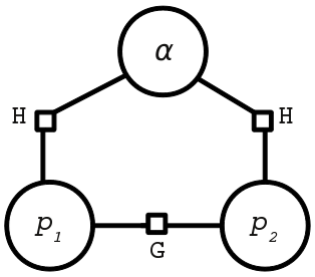


Figure 3.4: Factor graph for 1 road.

$$log\psi(\mathbf{y}, \mathbf{x}) = \sum_{r \in \Upsilon} \boldsymbol{\theta}_r^T \cdot \phi_r(\mathbf{x}, p_1, p_2, \alpha) + \theta_g \cdot \phi_g(p_1, p_2) \tag{3.10}$$

- $r \in \Upsilon = \{nry, rd, nrg\}$ | nry =non-road in yellow, nrg =non-road in green and rd =road.
- $\phi_{nry}(\mathbf{x}, p_1, p_2, \alpha) = H(\mathbf{x}, p_1, \alpha)$.
- $\phi_{rd}(\mathbf{x}, p_1, p_2, \alpha) = H(\mathbf{x}, p_2, \alpha) - H(\mathbf{x}, p_1, \alpha)$.
- $\phi_{nrg}(\mathbf{x}, p_1, p_2, \alpha) = 1 - H(\mathbf{x}, p_2, \alpha)$.
- $\phi_g(p_1, p_2) = G(p_1, p_2) \begin{cases} 0 & \text{if } p_1 < p_2 \\ -\infty & \text{if } p_1 \geq p_2 \end{cases}$

The first three potentials are decomposed as functions that measure a set of features in the BeP image. In particular, we propose to extract information by counting the different grids on the image, i.e. the white pixels (*Occupied* grids) and the black pixels (*Free* grids). Thus, we define the 2D visual features as $[O, F]$, where O accumulates the white pixels and F the black ones in a selected image region. These features are normalized with respect to the whole

image. Besides, the decomposition is carried out following the premise of lowering down the order of the potentials to reduce computational complexity during inference. Consequently, a pairwise function[2] $H(\mathbf{x}, p_i, \alpha)$ is proposed, which computes the $[O, F]$ vectors for each image patch created as the left half region of a BeP image, which has been split by a line hypothesis $line(p_i, \alpha)$, as represented in Fig. 3.3.

In total, there are three measured regions that have been colored in yellow, blue and green. More specifically, the yellow and green ones correspond to non-road areas, which would be ideally filled with white pixels. Due to the normalization, the visual features on the green area can be obtained by subtracting $H(\mathbf{x}, p_2, \alpha)$ from 1 (representing the total number of visual features). Additionally, the blue region overlaps with the yellow one, such that their subtraction $H(\mathbf{x}, p_2, \alpha)$ - $H(\mathbf{x}, p_1, \alpha)$ corresponds to the road area (ideally filled with black pixels). In fact, this is the trick to decompose the 3-tuple problem defined by $\phi(p_1, p_2, \alpha)$ into a linear combination of 2-tuple $H(p_i, \alpha)$ functions.

3.2.1. Loss definition for learning

The parameter learning is carried out with CRFs as exposed in Section 3.1.2 based on [Hazan and Urtasun, 2010, Schwing et al., 2011]. The parameter vector is composed of 7 elements coming from the 3 potentials that measure 2 features on each region, and the additional pairwise constraint, i. e. $\boldsymbol{\theta} = [\theta^O_{nry}, \theta^F_{nry}, \theta^O_{rd}, \theta^F_{rd}, \theta^O_{nrg}, \theta^F_{nrg}, \theta_g]$. It must be noted that in theory, the learned parameter vector $\hat{\boldsymbol{\theta}}$ must present the same signs as the following mask $\tilde{\boldsymbol{\theta}} = [+, -, -, +, +, -, +]$ in order to correctly add the votes of each feature during inference. Indeed, we want to maximize (see Eq. 3.2) the energy related to occupied grids (bounding urban structure) in the non-road regions and the energy of the free grids in the road areas. Hence, the features multiplied by negative weights will be minimized because they correspond to "outliers" or noise on each region, i.e. white pixels on road hypotheses and black pixels on non-road hypotheses.

The learning process can be guided by the definition of the following loss function, which accounts for the pixel-wise error and it is decomposed like the image features. The error is computed as the percentage of pixels that are wrongly predicted on each of the areas of interest for every road hypothesis. Keeping in mind the pairwise potentials defined for inference, the loss function decomposition is represented with the factor graph in Fig. 3.5 and Eq. 3.11.

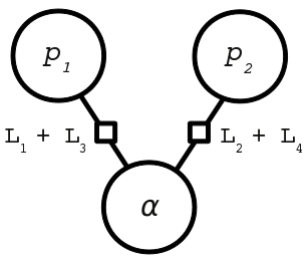


Figure 3.5: Factor graph of the loss for 1 road.

[2]This function is considered as "pairwise" because it depends on two output random variables: p_i and α. The vector $\mathbf{x}$ represents the observable 2D grid map.

$$\Delta(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{r \in \Upsilon} \phi_{loss,r}(\mathbf{x}_{gt}, p_1, p_2, \alpha) \tag{3.11}$$

- $\mathbf{x}_{gt} = [\mathbf{x}, p_1^{gt}, p_2^{gt}, \alpha^{gt}]$ represents the observed data and the ground-truth labels for the defined random variables.
- $r \in \Upsilon = \{nry, rd, nrg\}$ | nry =non-road in yellow, nrg =non-road in green and rd =road.
- $\phi_{loss,nry}(\mathbf{x_{gt}}, p_1, p_2, \alpha) = L_1(\mathbf{x_{gt}}, p_1, \alpha) = H^F(p_1, \alpha)$.
- $\phi_{loss,rd}(\mathbf{x_{gt}}, p_1, p_2, \alpha) = L_3(\mathbf{x_{gt}}, p_1, \alpha) + L_4(\mathbf{x_{gt}}, p_2, \alpha) = -H^O(p_1, \alpha) + H^O(p_2, \alpha)$.
- $\phi_{loss,nrg}(\mathbf{x_{gt}}, p_1, p_2, \alpha) = L_2(\mathbf{x_{gt}}, p_2, \alpha) = 1 - H^F(p_2, \alpha)$.

Each of the factors L_j defined above depend on H, which is the accumulator function in Fig. 3.3, but in this case particularized for the features F or (*Free* grids) O (*Occupied* grids), as indicated in the superscripts. Fig. 3.6 provides a graphical representation for a better understanding.

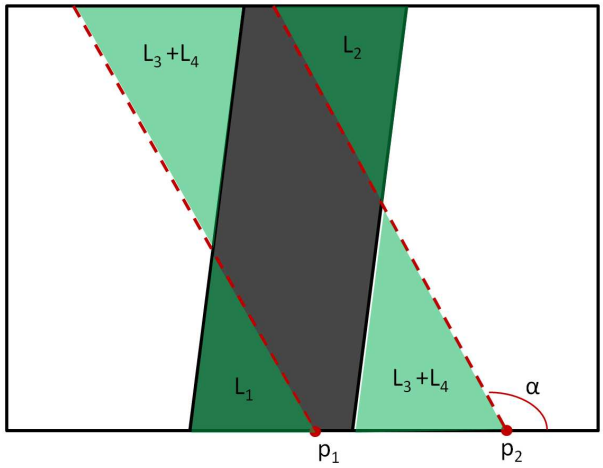


Figure 3.6: Example of the loss computation for 1 straight road. Considering an ideal synthetic image plotted in black (road) and white (occupied grids) and, the road hypothesis in red dashed lines. The pixelwise error is calculated as the sum of the areas where the hypothesis does not match with the synthetic ground truth. These areas are overlaid in green.

3.3. Modelling 4 intersecting roads.

Once reviewed the model for the basic case of 1 straight road on the BeP image, this section describes the more general case of four intersecting roads. Again, we assume that the road on the bottom of the BeP image contains the ego-vehicle with the stereo camera. Fig. 3.7.b displays the measured 2D grid map of an intersection with four arms. As can be observed, inferring the scene layout from this BeP image is a very challenging problem due to the intrinsic artifacts associated to the stereo reconstruction and other undesired objects or obstacles that could be present in the 3D scene. Therefore, generating synthetic images from the ground-truth labels

helps for the task of model validation. Fig. 3.7.c represents an ideal BeP image (road area in black) corresponding to the real scene on the left image. This synthetic grid map has been built from a discretized ground truth.

(a) Last frame of the sequence

(b) Measured 2D grid map

(c) Synthetic image from discretized ground truth

Figure 3.7: Last frame and BeP image samples of 4 intersecting roads [Geiger et al., 2011a].

Similar to the previous section, we seek the definition of the smallest possible set of random variables that completely characterizes the 4-road intersection while also having low-order potentials in the factor graph. First of all, let us propose a smart splitting of the scene inspired by the room layout in [Schwing et al., 2012]. In our case, we do not rely on a set of detected vanishing points in the scene, but on the unknown intersecting points between street boundaries. We divide the intersection in five main regions of interest that are depicted in Fig. 3.8. *A, B, C* and *D* zones represent urban structure that bounds the free or navigable area *E* of the road.

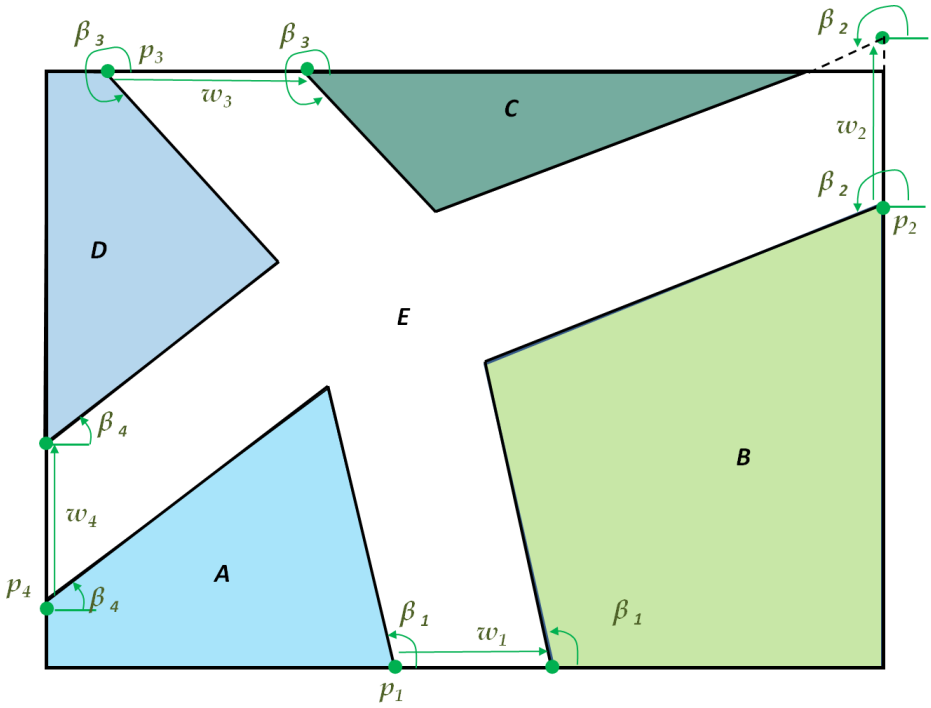


Figure 3.8: Generic layout and proposed parameterization of 4 intersecting roads. There are five regions of interest in the BeP in which road/streets area is painted in white for convenience and non-road regions in a different color each one. It also depicts the physical meaning of the 12 proposed random variables that fully characterize the intersection layout.

Additionally, the figure above contains the designed parameterization, which consists in 12 random variables that fully characterize 4-roads intersections. There are 4 angles β_i, 4 points p_i on image borders and 4 widths w_i measured from every point. Hence, every road is described by one point, one width and one angle. The points can be placed out of the image dimensions to allow modeling all the possible intersection layouts. An example can be observed in Fig. 3.8 for $p_2 + w_2$ in region C.

It must be noted that other parameterizations have been thoroughly analyzed and tested as part of a long research work before yielding this final configuration. They are described in detail in Appendix A. However, in the remainder of this chapter, we well consider the parameterization in Fig. 3.8, for best performance.

As explained for 1 straight road, the 2D visual features $[O, F]$ are also counted on each hypothesis region of Fig. 3.8. This involves that a function for computing the features on each region depends on several random variables, which vary from 4 to 6. Consequently, the structured prediction problem would require high order potentials, which increases the complexity for learning and inference tasks. Although every region of interest could be decomposed in two halves, as proposed in other approaches (Appendix A), we introduce here a different strategy to restrict ourselves to pairwise potentials. In particular, we describe each road with only one random variable y_i that encodes three sub-elements: p_i, w_i and β_i, as it is formally stated below.

$$y_i = f(\mathbf{p}_i, \mathbf{w}_i, \boldsymbol{\beta}_i) \Rightarrow < (\mathbf{p}_i + \mathbf{w}_i(j))_{\beta_i(k)} > \quad \forall i = 1, ..., 4; \quad j = 1, ..., N_{w_i} \quad k = 1, ..., N_{\beta_i} \tag{3.12}$$

Therefore, we reduce the order of the potentials at the cost of increasing the cardinality of every variable y_i. Although this approach increases computational times and may limit the hypotheses space, it benefits the convergence during inference, so we also save time when searching for the optimal solution. Fig. 3.9 illustrates the proposed encoding.

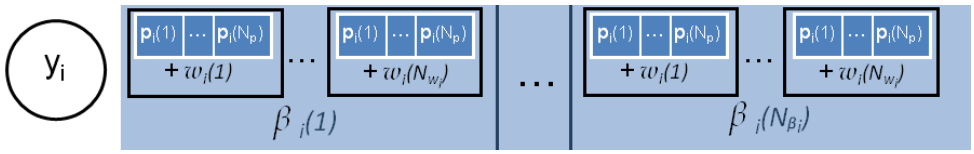


Figure 3.9: Illustration of the variables encoding. Each node in the factor graph encodes 3 variables: p_i, w_i and β_i. Each of them will have N_{pi}, N_{wi} and $N_{\beta i}$ number of states, respectively. Hence, their product gives the total number of states for y_i.

3.3.1. Factorization and *Branch and Bound* inference

Fig. 3.10 shows the factor graph, in which every factor corresponds to one of the regions of interest in Fig. 3.8. In particular, $RegionB \mapsto F_1$, $RegionC \mapsto F_2$, $RegionD \mapsto F_3$ and $RegionA \mapsto F_4$. According to this factorization, we present the following structured prediction problem in Eq. 3.13, where ϕ_{ABCD} decomposes on pairwise potentials that correspond to the factors F_i.

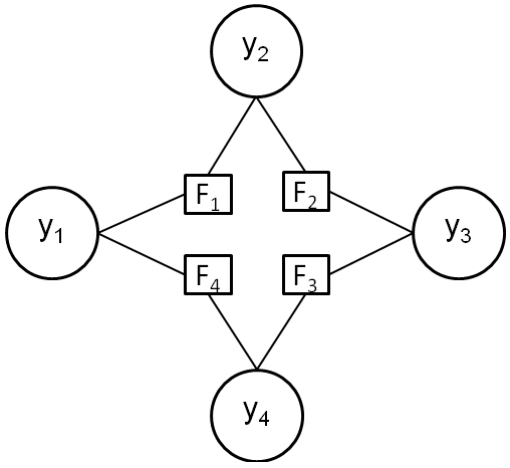


Figure 3.10: Proposed graphical model with pairwise factors for inferring the layout of 4 intersecting roads.

$$log\psi(\mathbf{y},\mathbf{x}) = \boldsymbol{\theta}^T_{ABCD} \cdot \phi_{ABCD}(\mathbf{x}, y_1, y_2, y_3, y_4) + \boldsymbol{\theta}^T_{E} \cdot \phi_{E}(\mathbf{x}, y_1, y_2, y_3, y_4) \tag{3.13}$$

- $\phi_{ABCD}(\mathbf{x}, y_1, y_2, y_3, y_4) = \sum_{i=1}^{4} F_i(\mathbf{x}, y_j, y_k)$.
- $\phi_E(\mathbf{x}, y_1, y_2, y_3, y_4) = \phi(\mathbf{x}) - \phi_{ABCD}(\mathbf{x}, y_1, y_2, y_3, y_4)$.
- $\phi(\mathbf{x}) \Rightarrow$ constant function that represents all the features in the image.

After the first explored proposals in Appendix A and the detected issues while doing approximate inference with the message-passing algorithm [Schwing et al., 2011], we propose a different approach based on exact inference. In particular, we extend the ESS algorithm of [Lampert et al., 2009] to the intersection layout problem in a similar fashion to the indoor room layout inference of [Schwing and Urtasun, 2012]. Basically, it consists in a Branch and Bound technique that splits the hypotheses space into two halves and evaluates a bounding function, which imposes some constraints and upper-bounds the search for the global maximum objective. The process of finding the optimal labels for the random variables defining the scene layout can be seen as a best-first search, in which a priority queue holds the hypotheses with the highest scores. The *branch* step splits the hypotheses set located at the top of queue into two disjoint subsets and the *bound* step evaluates an upper bound for the scores of each subset. Finally, these subsets are inserted in the queue with their corresponding score and this is repeated until a single hypothesis is found at the top of the queue and can not be split further.

Therefore, we need to describe the discrete random variables as intervals representing sets of hypotheses, such that for each variable there are lower and upper limits that contain the set of hypotheses: $Y_i = [y_{i,l}, y_{i,u}]$. These bounding candidates correspond to the smallest and largest regions that can be represented by the encoded variables in Fig. 3.9. Table 3.1 specifies the limits for each region of Fig. 3.8 in terms of the three encoded variables (p_i, w_i, β_i). As can be observed, the interval of every region is determined from the pairwise combination of its constituent nodes in the factor graph. The subindexes u and l correspond to the upper and lower limits of every random variable.

Table 3.1: Interval specification on each region of our model for BB inference.

Node	lower limit $y_{i,l}$	upper limit $y_{i,u}$	Region	$[smallest; largest]$
y_1	$< p_{1,l}, w_{1,l}, \beta_{1,u} >$	$< p_{1,u}, w_{1,u}, \beta_{1,l} >$	A	$[(y_{1,l}, y_{4,l}); (y_{1,u}, y_{4,u})]$
y_2	$< p_{2,l}, w_{2,l}, \beta_{2,u} >$	$< p_{2,u}, w_{2,u}, \beta_{2,l} >$	B	$[(y_{1,u}, y_{2,l}); (y_{1,l}, y_{2,u})]$
y_3	$< p_{3,l}, w_{3,l}, \beta_{3,l} >$	$< p_{3,u}, w_{3,u}, \beta_{3,u} >$	C	$[(y_{2,u}, y_{3,u}); (y_{2,l}, y_{3,l})]$
y_4	$< p_{4,l}, w_{4,l}, \beta_{4,l} >$	$< p_{4,u}, w_{4,u}, \beta_{4,u} >$	D	$[(y_{3,l}, y_{4,u}); (y_{3,u}, y_{4,l})]$

Given the ESS as reference framework, the Branch and Bound solution applied to our problem can be formulated with the following structured prediction objective:

$$\hat{\mathbf{y}} = \underset{y}{argmax} \left\{ \hat{\boldsymbol{\theta}}^T \cdot \phi(\mathbf{x}, \mathbf{y}) \right\} = \underset{y}{argmax} \left\{ f^+(\mathbf{x}, \mathbf{y}) + f^-(\mathbf{x}, \mathbf{y}) \right\} \tag{3.14}$$

where the "quality function" $f = f^+ + f^-$ assigns a score to every hypothesis of the search space defined by our proposed parameterization (Fig. 3.8). Besides, following the decomposition of the ESS method when applied to a linear classification problem with object bounding boxes [Lampert et al., 2009], the quality function is split in positive (f^+) and negative (f^+) summands. In our case, the sign is given by the elements of the parameter vector $\boldsymbol{\theta}$, because our 2D visual features ($[O, F]$) are always positive, thus, all the factors in Fig. 3.10 and Eq. 3.13 are positive, too. In particular, we aim at maximizing the grids of type O inside the occupied regions A, B, C and D, and the grids of type F on the road region E. Then, the parameter vector is four-dimensional: $\boldsymbol{\theta} = [\theta^O_{ABCD}, \theta^F_{ABCD}, \theta^O_E, \theta^F_E]$ and ideally, when this vector is learned, it should present the signs $s = [+, -, -, +]$ in order to maximize F in road areas and O in the occupied ones. Therefore, we can formally describe the positive and negative summands in the next expressions:

$$f^+(\mathbf{x}, \mathbf{y}) = \theta^O_{ABCD} \cdot \sum_{i=1}^{4} F^O_i(\mathbf{x}, y_j, y_k) + \theta^F_E \cdot \left\{ \phi(\mathbf{x}) - \sum_{i=1}^{4} F^F_i(\mathbf{x}, y_j, y_k) \right\}$$

$$f^-(\mathbf{x}, \mathbf{y}) = \theta^F_{ABCD} \cdot \sum_{i=1}^{4} F^F_i(\mathbf{x}, y_j, y_k) + \theta^O_E \cdot \left\{ \phi(\mathbf{x}) - \sum_{i=1}^{4} F^O_i(\mathbf{x}, y_j, y_k) \right\}$$

Furthermore, a bounding function $\hat{f}$ has to be determined, which fulfills the two properties stated in [Lampert et al., 2009]: (1) $\hat{f}$ has to be an upper-bound on f and, (2) it has to guarantee the convergence to optimal solution. However, we can not directly apply the union and intersection of rectangles proposed as bounding function by [Lampert et al., 2009]. The first reason is that our regions are not colinear with the image boundaries and do not form rectangles, but more general convex polygons (triangles, quadrilaterals, ...). The second one is that our problem decomposes in pairwise factors, which can be treated separately as individual components of the intersection layout and, by definition, they do not intersect or merge. Consequently, we arrive to the same modeling as in [Schwing and Urtasun, 2012]. The bound of our quality function is obtained from the smallest and largest areas that every region of our proposed intersection layout can achieve (previously defined in Table 3.1).

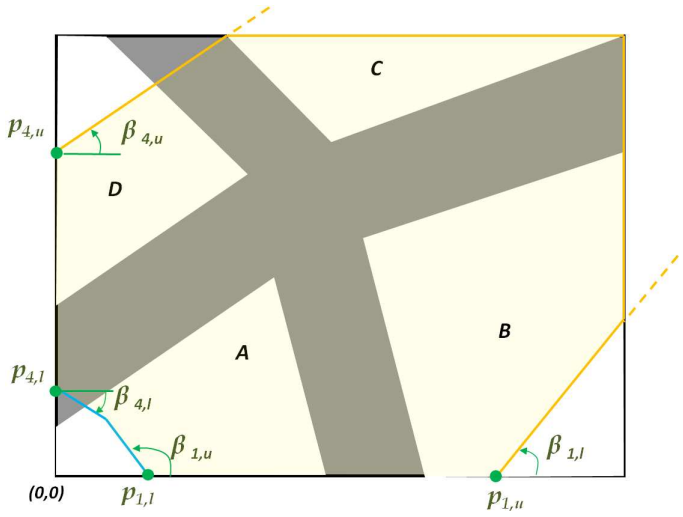


Figure 3.11: Smallest (blue lines) and largest (orange lines) bounds for Region A. The yellow overlaid area represents the hypotheses set between these bounds. The width of the roads w_i has not been included because it does not affect for the evaluation of region A.

Consequently, the bounding function $\hat{f}$ for 1 region is formulated as the notation in Eq. 3.15.

$$\hat{f}(Y_1, Y_2, Y_3, Y_4) = \sum_{r \in \Upsilon} \hat{f}_r(Y_i, Y_j) \tag{3.15}$$

$$\hat{f}_r(Y_i, Y_j) = f_r^+(\mathbf{x}, \mathcal{I}_{largest}(y_i, y_j)) + f_r^-(\mathbf{x}, \mathcal{I}_{smallest}(y_i, y_j)) \tag{3.16}$$

In the equations above, $r \in \Upsilon = \{A, B, C, D\}$ and $\mathcal{I}$ denotes the labels of the interval affecting the variables y_i and y_j for the smallest or largest areas of region r. Figure 3.11 depicts these areas for region A in blue and orange bounding lines respectively. This illustration represents an example of a 4-road intersection and the bounds for region A that has been exposed in Table 3.1. The space of hypotheses for region A is overlaid in yellow. Hence, branching will start by dividing this set in two halves, then, bounding will evaluate $\hat{f}$ on each of them in order to get the scores that are employed to sort the candidates in the priority queue.

In the next section, we explain how to efficiently compute the features from the 2D grid map (or known as BeP image) given a region hypothesis.

3.3.2. Feature computation based on Integral Geometry

The *integral image* representation was originally proposed by [Viola and Jones, 2001b] to efficiently evaluate features that have to be computed multiple times (e.g. at different scales or locations). More recently, [Schwing et al., 2012] extended this concept to *Integral Geometry (IG)* in order to compute the visual features in semantically labeled images of house rooms. The vanishing points of the scene were employed as ray generators that formed a non-rectangular grid over the image, in which the *integral image* algorithm was applied to obtain the sums of the accumulated features in a given region of the grid. Inspired by this work, we also propose an IG approach to compute the features contained inside the defined *A, B, C, D* regions in Fig. 3.8. Each of them is bounded by two segments corresponding to two streets/roads limit. Thus, in

our case, the generator of rays over the BeP image is the intersecting point on each region. Indeed, all the possible configurations between these two rays can be viewed as lines crossing the left and bottom image boundaries such that they draw an imaginary non-rectangular grid. An example for region A is shown in Fig. 3.12. We obtain the number of visual features in the blue region A accumulating the values in the cells from the red point towards the origin. For the remaining regions B, C and D, the same approach is easily deployed by using previous vertical and/or horizontal flips of the BeP image.

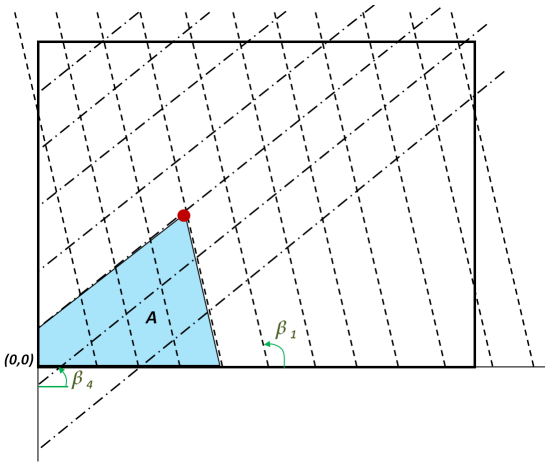


Figure 3.12: Space of hypotheses for 1 region of interest with fixed β_1 and β_4 values. It can be viewed as two rays crossing the left and bottom boundaries of the image that intersect in a point inside the measured 2D grid map. Then, the occupied/free grids can be accumulated by using the "integral image" algorithm by [Viola and Jones, 2001b].

This graphical representation can be viewed as a matrix of 2D bins. Each of them accumulates the number of visual features ([O,F]) delimited by the physical non-squared cells. To efficiently compute the cell membership of every measured grid (BeP image pixel), we make use of trigonometry rules. A descriptive diagram is shown in Fig. 3.13.

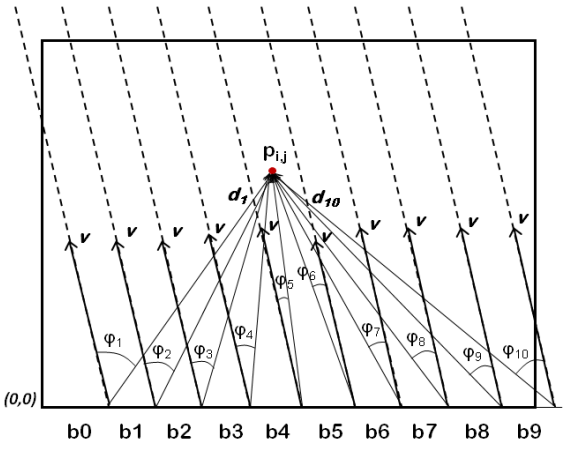


Figure 3.13: Efficient computation of the bin membership for every pixel on the BeP image. For clarity, only one dimension is represented in this diagram. From the vector of angles $\boldsymbol{\varphi}$, the bin index can be easily obtained by detecting a sign change.

Instead of directly calculating the angles φ_i, we will only evaluate their sign to detect the bin membership of every pixel in the image. In fact, we will determine the rotation between the vectors $\mathbf{v}$ and $\mathbf{d}_i$ at every discretized position in the image. From Eq. 3.17, the *sine* can be isolated in order to get the angle sign. Therefore, the computation of Eq. 3.18 is not needed, as we can directly evaluate the sign from the numerator (Eq. 3.19). Then, having the vector $\mathbf{s}$, the bin index is obtained as the vector position in which a step from 0 to 1 is triggered. Similarly, this can be extended to the vertical dimension to identify the second index of the cell in the 2D grid of Fig. 3.12.

$$\mathbf{d}_i^T = R(\phi_i) \cdot \mathbf{v}^T \Rightarrow \begin{bmatrix} \mathbf{d}_i(x) \\ \mathbf{d}_i(y) \end{bmatrix} = \begin{pmatrix} cos\varphi_i & -sin\varphi_i \\ sin\varphi_i & cos\varphi_i \end{pmatrix} \begin{bmatrix} \mathbf{v}(x) \\ \mathbf{v}(y) \end{bmatrix} \tag{3.17}$$

$$sin\varphi = \frac{\mathbf{v}(x) \cdot \mathbf{d}_i(y) - \mathbf{v}(y) \cdot \mathbf{d}_i(x)}{\mathbf{v}(x)^2 + \mathbf{v}(y)^2} \tag{3.18}$$

$$s_i = (\mathbf{v}(x) \cdot \mathbf{d}_i(y) - \mathbf{v}(y) \cdot \mathbf{d}_i(x)) > 0 \tag{3.19}$$

3.3.3. Loss definition for learning

From Eq. 3.13, we derived that the parameter vector is $\boldsymbol{\theta} = [\theta^O_{ABCD}, \theta^F_{ABCD}, \theta^O_E, \theta^F_E]$. Then, only four weights have to be estimated during learning, which is performed based on Section 3.1.2 and [Hazan and Urtasun, 2010, Schwing and Urtasun, 2012]. For the computation of the loss, we assume the same graphical model in Fig. 3.10 and count the pixel-wise error in a very similar way as exemplified for 1 straight road in Fig. 3.6, but in this case extended for the regions proposed in Fig. 3.8. In addition, the loss can be also scaled by an empirical factor if needed, together with some constraints like reported in Appendix A.

3.4. Experimental results

Firstly, the dataset of BeP images is presented, next, the evaluation protocol is reviewed in detail and afterwards, the experimental results for straight roads and 4 intersecting roads are reported. Finally, a discussion summarizes the main derivations from the experiments.

3.4.1. Intersections dataset

In Section 3.1.1 we provided an overview of the BeP images that are employed in this Thesis for inferring the layout of intersections in urban scenes. They come from a voxelization of the 3D scene reconstructed from stereo sequences [Geiger et al., 2011a]. The videos have been recorded by the robotic autonomous platform *AnnieWay* and processed with ELAS library [Geiger et al., 2010]. As a result, a dataset of 113 BeP images is available to download from [Geiger, 2011]. Every image is a sparse 2D occupancy grid map of the environment ahead of the ego-vehicle. Fig. 3.14 displays all of them with the ground-truth road boundaries overlaid in green. The ground truth was obtained from Google maps, as stated by [Geiger et al., 2011a]. Additionally, Fig. 3.15 displays the corresponding synthetic images that have been built directly from the ground-truth labels. They are conceived as "ideal" BeP images, with the road area in black and the occupied area in white and will be employed for validating the proposed methodology.

In total, there are 22 images depicting a straight road, 18 containing 3 intersecting roads and the remaining 73 representing 4 intersecting roads. It must be remarked that we have not included here any special treatment for 3-way intersections, which is a future research goal. Hence, for practical purposes they are considered as 4 intersecting roads in this Thesis.

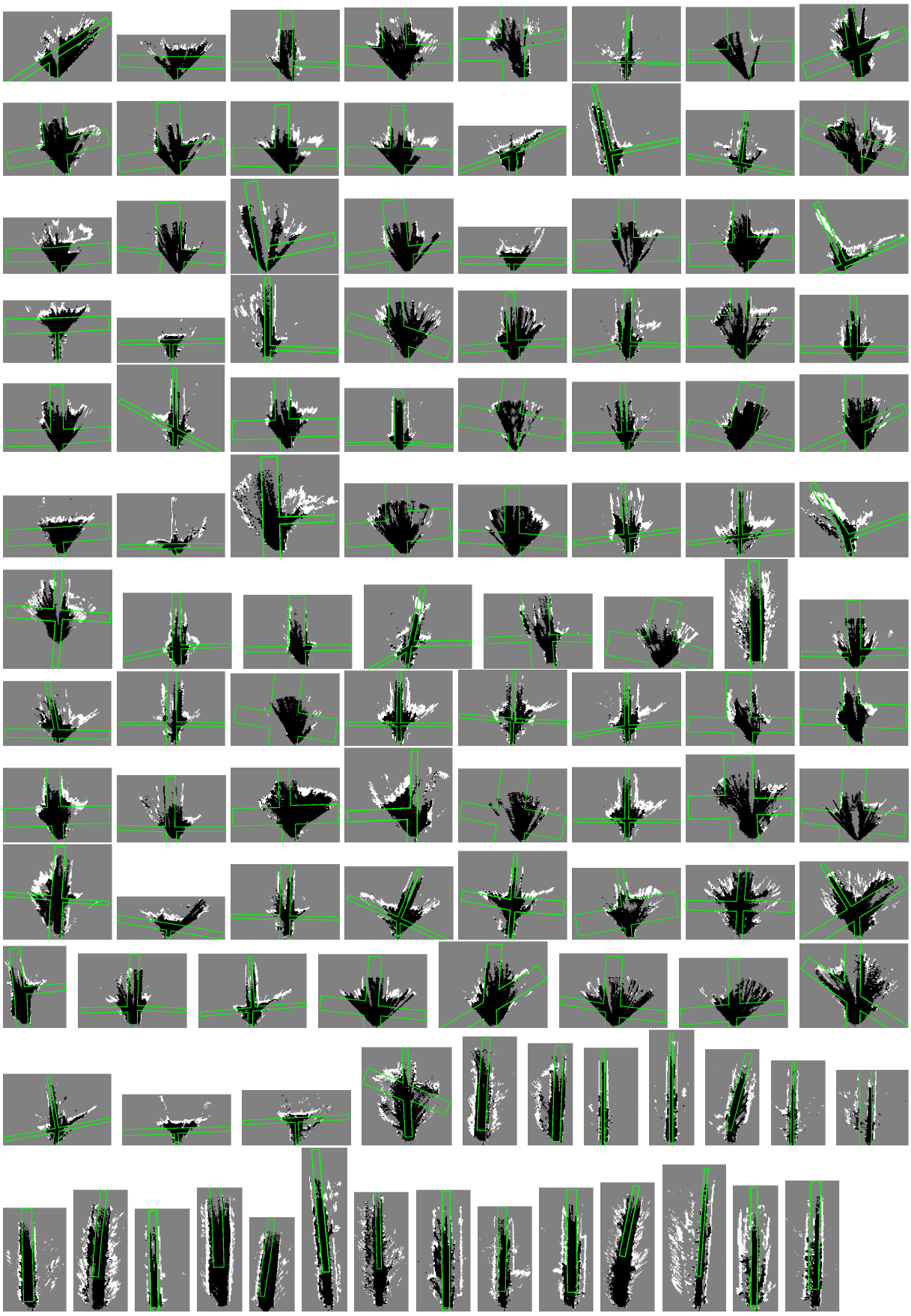


Figure 3.14: Bird's eye perspective images with ground-truth roads overlaid in green.

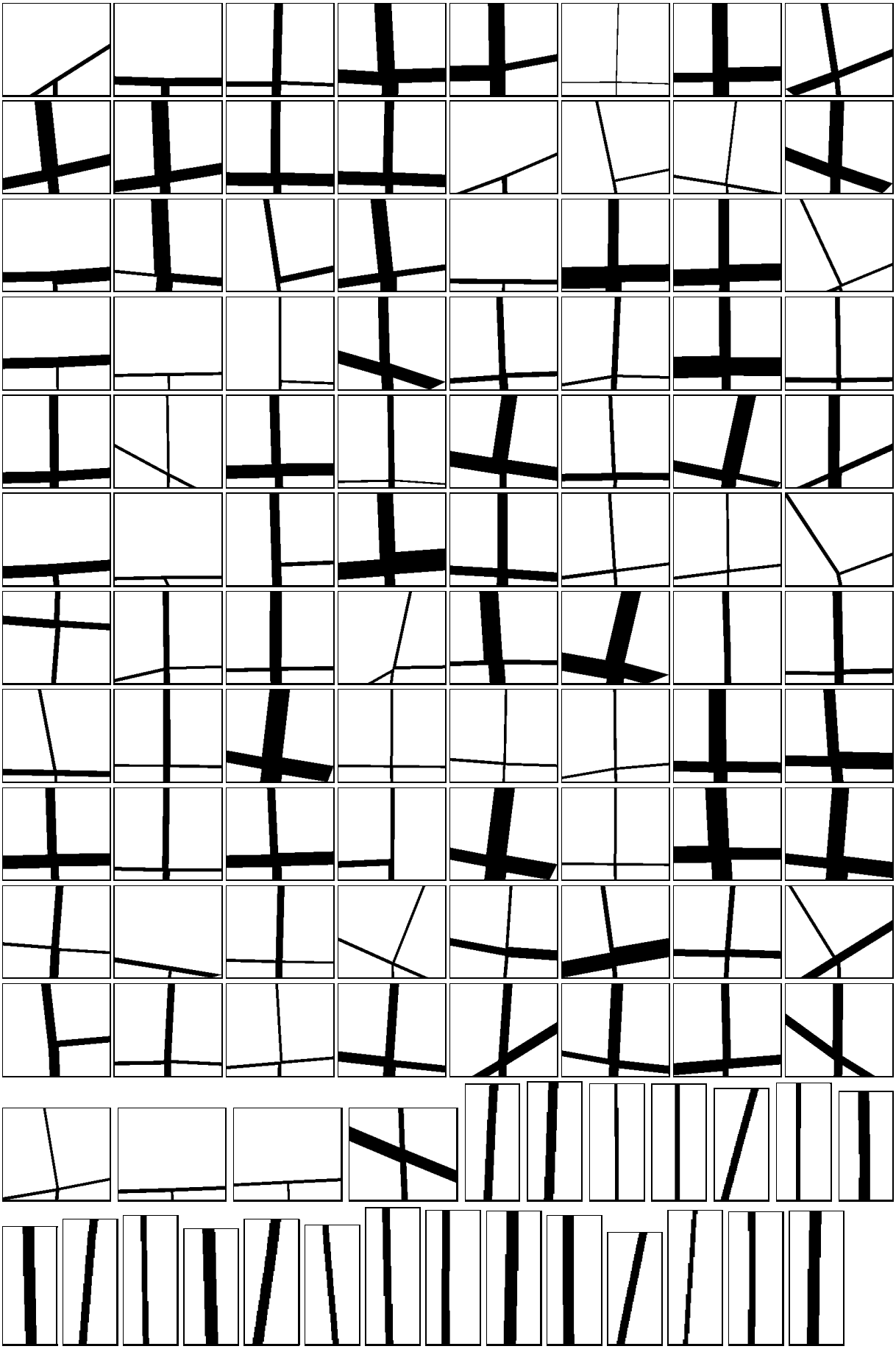


Figure 3.15: Bird's eye perspective ideal images. Synthetically built from the ground truth.

3.4.2. Evaluation protocol

Due to the reduced size of the dataset, we propose to carry out *leave-one-out* cross-validation experiments in which one sample is kept for validation and the others are employed for training the model. Then, a different vector of parameters $\hat{\boldsymbol{\theta}}$ is learned on each round and used for prediction on the validation sample. This process is repeated N times, where N equals the number of roads of each type, i.e. straight or 4 intersecting roads. After it, the results from the validation of each individual sample are averaged to yield a final error measure, as we will describe next.

On the other hand, the learning process is based on the theoretic concepts introduced in Section 3.1.2 and implemented in the works [Hazan and Urtasun, 2010, Schwing et al., 2011]. According to them and reminding Eq. 3.7, we define the following setup for all the experiments: $p = 2$ (L2-norm), $\epsilon = 1$ (CRF learning), a relative duality gap of 10^{-4} (this is a stopping criteria for convergence of the underlying optimization algorithm) and 5 different values of the regularizer C $[10, 1, 0.1, 0.01, 0.001]$. The latter one is for estimating the value that produces the best prediction results from the corresponding learned models. For structured prediction, we employ the dcBP engine [Schwing et al., 2011] and an adaptation of the BB algorithm in [Schwing et al., 2012]. In this way, we test both approximate and exact inference on our proposed models for road layout prediction.

For results evaluation, two measures are proposed below:

1. The pixelwise error computed with the loss function (Δ) defined on each case, i.e. 1 straight road (in Section 3.2) and 4-way intersections (in Section 3.3). It must noted that this error measure also captures the noise in the observed data, thus, when observing the measurements in the next sections we will provide an appropriate interpretation of their values. This pixelwise loss contributes to the model training, but it is employed here as a validation measure after inferring the roads. It must be reminded that the proposed inference algorithms find the solution with the maximum energy and it does not need to match the solution with the minimum loss.

2. The overlapping ratio between the ground-truth layout of the road and the predicted layout. This is measured with the Intersection over Union as it was also done in [Geiger et al., 2011a], which is based on the original measure of the overlap between objects [Everingham et al., 2010].

$$IoU = \frac{area(RL_{pred} \bigcap RL_{gt})}{area(RL_{pred} \bigcup RL_{gt})}$$

   Where RL refers to Road Layout, and suffixes $pred$ and gt represent the predicted and ground-truth layouts, respectively.

Additionally, the experiments have been carried out on an i7 CPU @2.5GHz with 12 GB of RAM. The 4 cores are used during learning and inference.

3.4.3. Inferring straight roads

This section provides the predicted roads for the model presented in Section 3.2. According to dataset priors, the discretization of the model variables p_1, p_2, α has been set to the ranges $[60, 110]$ pixels, $[70, 120]$ pixels and $[70°, 125°]$, respectively. Besides, the cardinality is 25 for the points and 55 for the angle, meaning that the states of the variables are separated 2 pixels and 1° respectively. This makes a search space of $25^2 \cdot 55 = 34,375$ hypotheses.

3.4.3.1. Tests on synthetic BeP images

A preliminary evaluation of the model on synthetic images with different levels of noise is presented in this subsection. For learning the model parameters, the setup is the one mentioned in the section above. For roads prediction, the approximate inference algorithm based on distributed convex Belief Propagation [Schwing et al., 2011] is used. To create random perturbations on the synthetic images, a random noise generator is employed, which is inspired in [Domke, 2010] and defined in Eq. 3.20.

$$I' = I \diamond (1 - T^{\gamma}) + (1 - I) \diamond T^{\gamma} \tag{3.20}$$

where I is a sample image, I' is the resulting image with noise, $T = random(size(I))$ is a matrix of random values with the same size as I, the symbol $\diamond$ refers to the element-wise multiplication of the matrices and γ is a configurable parameter that sets the level of noise. In particular, we have used the values in the first column of Table 3.2, which produces the perturbations shown in the examples of Fig. 3.16. The results on the table summarizes several statistics from the synthetic straight road samples, while the figure illustrates some examples of the predictions.

Table 3.2: Straight roads. Prediction results for different levels of noise on synthetic images

	Pixelwise loss (%)			**IoU (%)**			**Inference time (s)**		
γ	**min**	**max**	**mean**	**min**	**max**	**mean**	**min**	**max**	**mean**
0	0.005	1.5	**0.68**	60.00	100.00	**79.36**	0.83	2.065	**1.183**
0.4	0.0	0.008	**0.002**	3.11	88.89	**71.87**	0.671	2.068	**1.304**
0.6	0.0	0.012	**0.005**	4.11	100.0	**68.70**	0.772	2.094	**1.295**
0.8	0.02	0.07	**0.042**	0.0	82.07	**36.96**	0.713	2.101	**1.208**

Observing the table above, the mean IoU decreases for higher levels of noise, which is the expected behavior, and it is more notable when approaching $\gamma = 0.8$. Indeed, there is a high image degradation for this level of noise, as can be seen in the last row of Fig. 3.16. However, the predictions are still closely approximated to the synthetic roads, due to the adaptation of the model parameters that are learned in the leave-one-out process. On the other hand, the table shows that pixelwise loss is lower for the noisy examples. This is explained by the definition of the loss in Section 3.2.1, which counts the white and black pixels of each hypothesis that are in wrong locations. For the degraded samples, the noise is in the form of white and black pixels randomly distributed on the image. Consequently, both of them are spread over the image, but

not concentrated in any special location. Hence, in the regions highlighted in Fig. 3.6, some of these noisy black and white pixels will not be counted as errors according to our loss modeling. Nevertheless, the loss increases for higher γ values, as it can be checked on Table 3.2.

Furthermore, looking at the first row of Fig. 3.16, one can interpret the IoU values 80%, 66.67% and 85.7% as very low overlapping indicators given the tight fitting of the red predictions to the synthetic images. This is explained by discretization artifacts, because in our experiments, the random variables p_1 and p_2 are quantified in jumps of 2 pixels, thus, 1 row of 1 pixel width on each side of the synthetic road can produce a low overlapping, such as 66.67%.

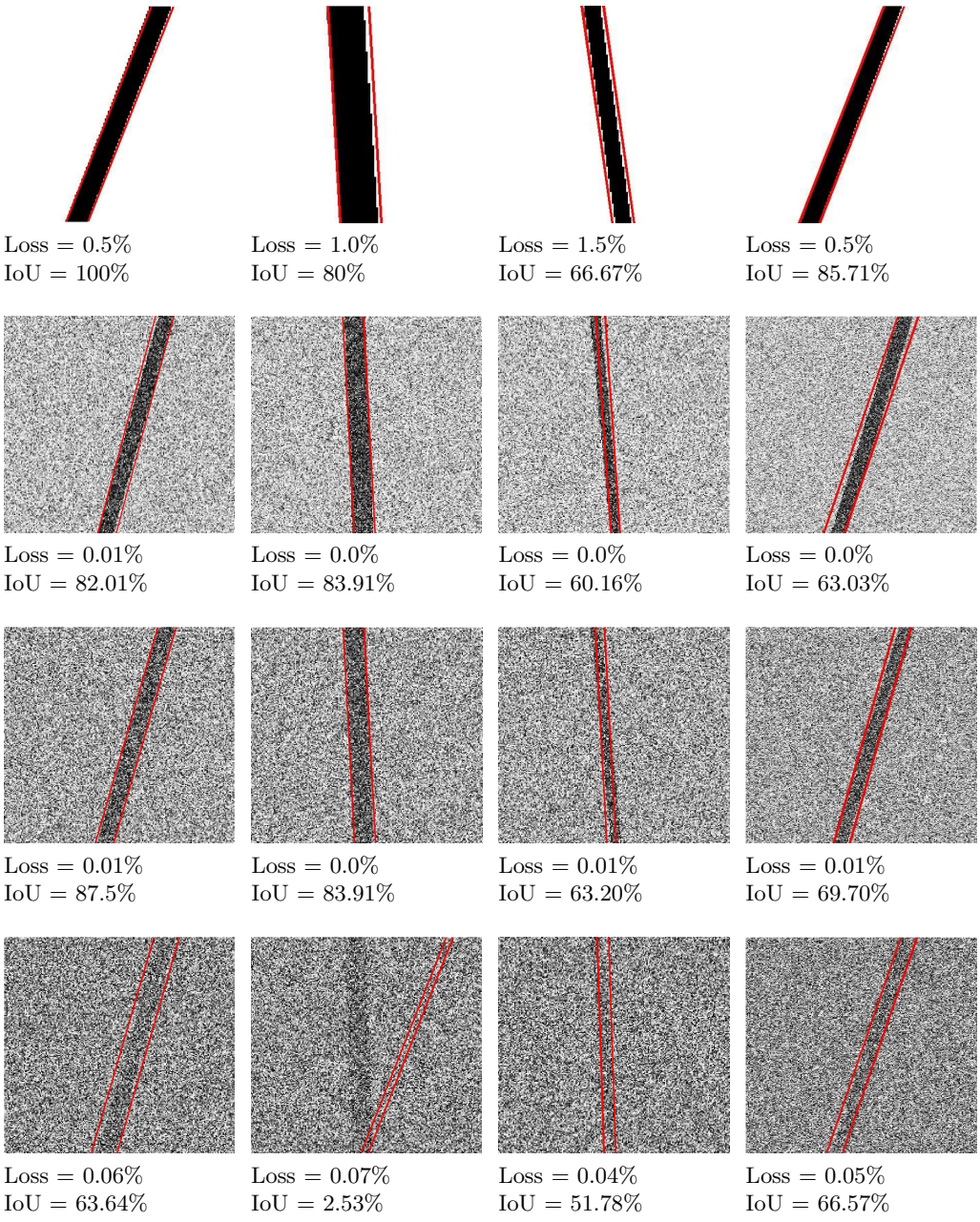


Figure 3.16: Prediction results for straight roads on synthetic images with variable noise. A different synthetic BeP image is on each column. Every row corresponds to different levels of noise (0, 0.4, 0.6 and 0.8). The predicted roads are in red color.

3.4.3.2. Tests on real BeP images

The results attached to this section have been computed on the real BeP images (Fig. 3.14) containing straight roads. For learning the model parameters, the setup is the one mentioned in Section 3.4.2. For inferring the road layouts, three approaches are compared: 1) dcBP engine [Schwing et al., 2011], which is an approximate inference algorithm, 2) an adapted BB approach, which is conceptually similar to the one exposed in Section 3.3 for 4 intersecting roads and, 3) an *exhaustive search* that iteratively navigates all the possible hypotheses to find the maximum. Assuming a learned parameter vector $\hat{\boldsymbol{\theta}}$ on each *leave-one-out* validation round, all of the inference methods yielded exactly the same labels for the random variables p_1, p_2, α (red lines in Fig. 3.17). It must be noted that the computed features/potentials are exactly the same, but the difference is in the optimization method (inference algorithm) that pursues the hypothesis with maximum energy. As a consequence of predicting the same road lines, all of the methods present the same error and IoU measurements for each image, but different timing.

For reporting the results, firstly, Table 3.3 shows the average pixelwise error (Loss), the average *Intersection over Union (IoU)* ratios and the training times for the different values of the regularizer C. As it can be observed, the highest IoU is obtained for C=0.01 and above. However, C=0.001 takes many iterations (around 2500), which is also translated into a large training time. For the values below it, the convergence is typically achieved in few iterations (5-20). Despite the lower pixelwise losses for C=1 and C=0.1, we have checked that the differences regard to noisy grids, that are also added in the loss. Thus, we designate as better prediction performance, the one related to higher IoU. Therefore, considering the best performing training setup (C=0.01), the predicted roads are displayed in red color in Fig. 3.17, where the green parallel lines represent the ground-truth street boundaries.

Table 3.3: Average prediction performance for different values of C during training

C	Avg. Loss (%)	Avg. IoU (%)	training time (ms)
10	3.32	16.70	352
1	0.92	51.32	863
0.1	0.99	52.52	484
0.01	1.01	53.22	978
0.001	1.01	53.22	85147

From the predicted roads in Fig. 3.17, a very low pixelwise error is obtained, which means that the proposed method is working correctly in terms of prediction accuracy given the observed data. Visually, the reader can see very good predictions. However, compared to the ground truth in green, there is a low IoU ratio, below 50%, for many samples. This important difference has a fair explanation in the high number of noisy grids in the BeP images. Nevertheless, our learned models are robust to some level of outliers, i.e. white pixels on road areas and black pixels in non-road ones, but cannot achieve higher overlapping ratios because of the sparse and noisy 3D reconstructions from stereo sequences. Indeed, the behavior of our learned models is correct, because maximizing the energy in Eq. 3.10 tries to separate street boundaries in order to delimit the road area, but at the same time to bring them closer in order to minimize white pixels in the road area. Besides, as it can be seen, the images contain many more black grids

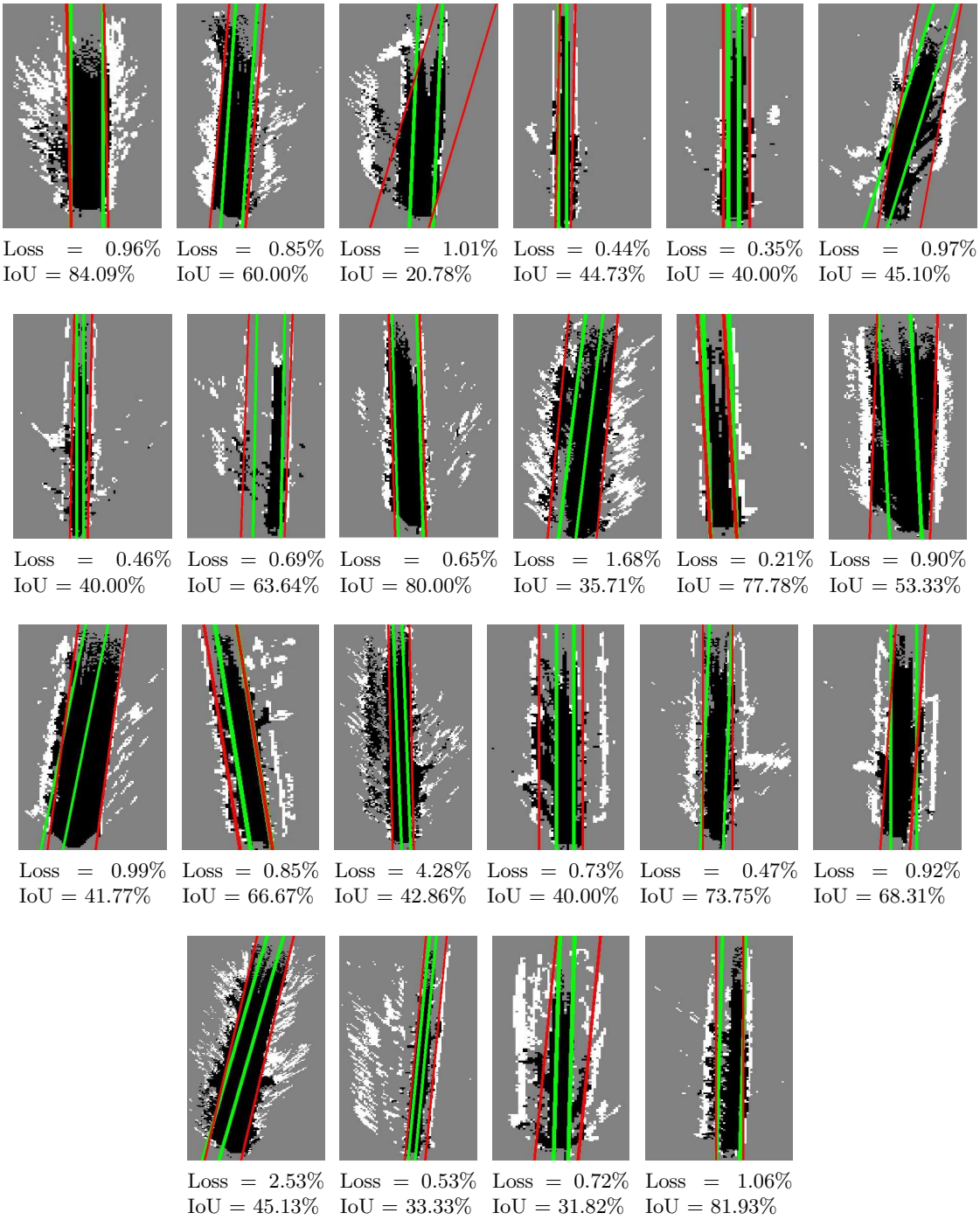


Figure 3.17: Results for inferring 1 straight road. Ground truth street boundaries are depicted in green and predicted roads in red. Besides, the pixelwise error and the intersection over union (IoU), both in percentage, are displayed on the bottom of each sample.

(free/non-occupied areas) than white ones. Consequently, the detected roads are wider than the ground truth. Therefore, it is difficult to achieve a better fitting to the road area with this measurements of the urban scene. The observed data may include wide streets such that the free grids in the 2D occupancy maps may belong to drivable roads and pedestrians sidewalks, which cannot be differentiated. Although some preprocessing step could filter the 2D occupancy grid maps (e.g. dilation or closing operations on the pixels), our guess is that better stereo

reconstruction algorithms, flow vectors and also object detection could support an improved prediction performance due to the addition of better and/or more cues.

Additionally, Table 3.4 summarizes the minimum, maximum, mean and standard deviation times for the three inference methods over the 22 samples of 1 road. Note that dcBP is able to manage several CPU cores, while BB and exhaustive search implementations only use one core.

Table 3.4: Prediction times for 1 straight road using different inference algorithms

Method	**min**	**max**	**mean**	**sdv**
dcBP [Schwing et al., 2011]	152ms	1021ms	944ms	179ms
Branch and Bound	4ms	19ms	7ms	3ms
Exhaustive search	5614ms	5706ms	5655ms	240ms

The table shows impressive prediction times for the exact inference approach of BB. Then, compared to the exhaustive search, it significantly reduces the time for finding the solution with maximum energy, but also outperforms the state-of-the-art approach known as dcBP.

3.4.4. Inferring 4 intersecting roads

This section provides the predicted roads for the model presented in Section 3.3. According to dataset priors, the discretization intervals of the model variables y_i, which were defined in Eq. 3.12, are shown in Table 3.5. Every p_i, w_i and β_i is discretized in 8 values uniformly distributed on each interval. Thus, the cardinality of y_i is $8^3 = 512$ possible states. This makes a search space of 512^4 hypotheses for the problem of 4-roads intersections.

Table 3.5: Domain intervals for the discrete random variables y_i in 4 intersecting roads

$\mathbf{p_1}$	$\mathbf{w_1}$	$\boldsymbol{\beta}_1$	$\mathbf{p_2}$	$\mathbf{w_2}$	$\boldsymbol{\beta}_2$
$[-17, 137]$	$[5, 40]$	$[-35, 35]$	$[55, 125]$	$[5, 40]$	$[60, 110]$
$\mathbf{p_3}$	$\mathbf{w_3}$	$\boldsymbol{\beta}_3$	$\mathbf{p_4}$	$\mathbf{w_4}$	$\boldsymbol{\beta}_4$
$[-6, 150]$	$[5, 40]$	$[152, 212]$	$[65, 100]$	$[5, 40]$	$[250, 280]$

3.4.4.1. Tests on synthetic BeP images

Initially, our proposed model has been tested on synthetic images to check the convergence and feasibility of the learning and inference algorithms given the features and the decomposition of the scene layout in Appendix A. In those trials, the prediction experiments were carried out on individual synthetic samples from Fig. 3.15 and employing the dcBP engine [Schwing et al., 2011]. However, as it is briefly commented at the end of the appendix, we found convergence issues for the inference algorithm. Our intuition is that, despite the different tested constraints and function shapes, the global energy function $E = \boldsymbol{\theta} \cdot \phi(x, y)$ may have discontinuities and smoothness problems, which prevents the the message-passing approximate inference algorithm (dcBP) to converge to a global maximum. Then, intermediate solutions, i.e. wrong layout configurations are returned as the output predictions.

On the contrary, we are able to predict the layout of 4-roads with the BB approach presented in this chapter, when validating the model on synthetic images. As already shown in Fig. 3.14, the real data set is very sparse and noisy. Thus, before the experiments on the real BeP images, we have carried out a set of experiments with an incremental level of random noise (Eq. 3.20) to test our proposed model under difficult observed 2D occupancy maps. We have used the values in the first column of Table 3.6, which produces the perturbations shown in the examples of Fig. 3.18 and 3.19. The results on the table summarizes several statistics from the 4-roads samples on the synthetic dataset, while the figures illustrate some examples of the predictions.

Table 3.6: 4 roads. Prediction results for different levels of noise on synthetic images

	Pixelwise loss (%)			**IoU (%)**			**Inference time (s)**		
γ	**min**	**max**	**mean**	**min**	**max**	**mean**	**min**	**max**	**mean**
0	0.005	34.03	**4.77**	60.63	99.99	**94.08**	0.071	160.96	**34.37**
0.4	0.18	31.57	**7.35**	59.90	99.80	**91.15**	0.049	124.28	**33.37**
0.6	0.27	37.14	**8.55**	58.62	99.70	**89.59**	0.037	108.91	**43.41**
0.8	4.29	38.63	**17.29**	51.44	95.49	**79.60**	0.80	118.74	**69.25**

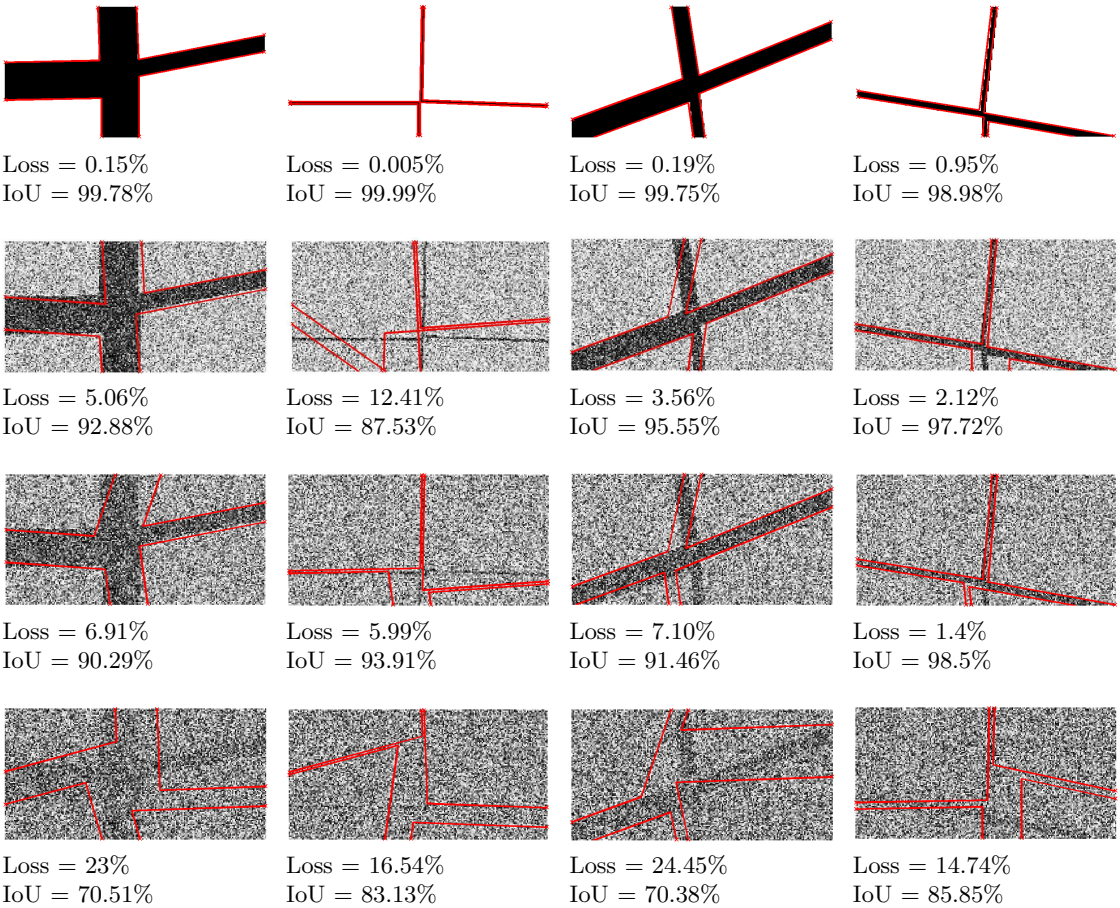


Figure 3.18: Predictions results for 4 intersecting roads on synthetic images with variable noise. A different synthetic BeP image is on each column. Every row corresponds to different levels of noise (0, 0.4, 0.6 and 0.8). The predicted intersection layouts are in red color.

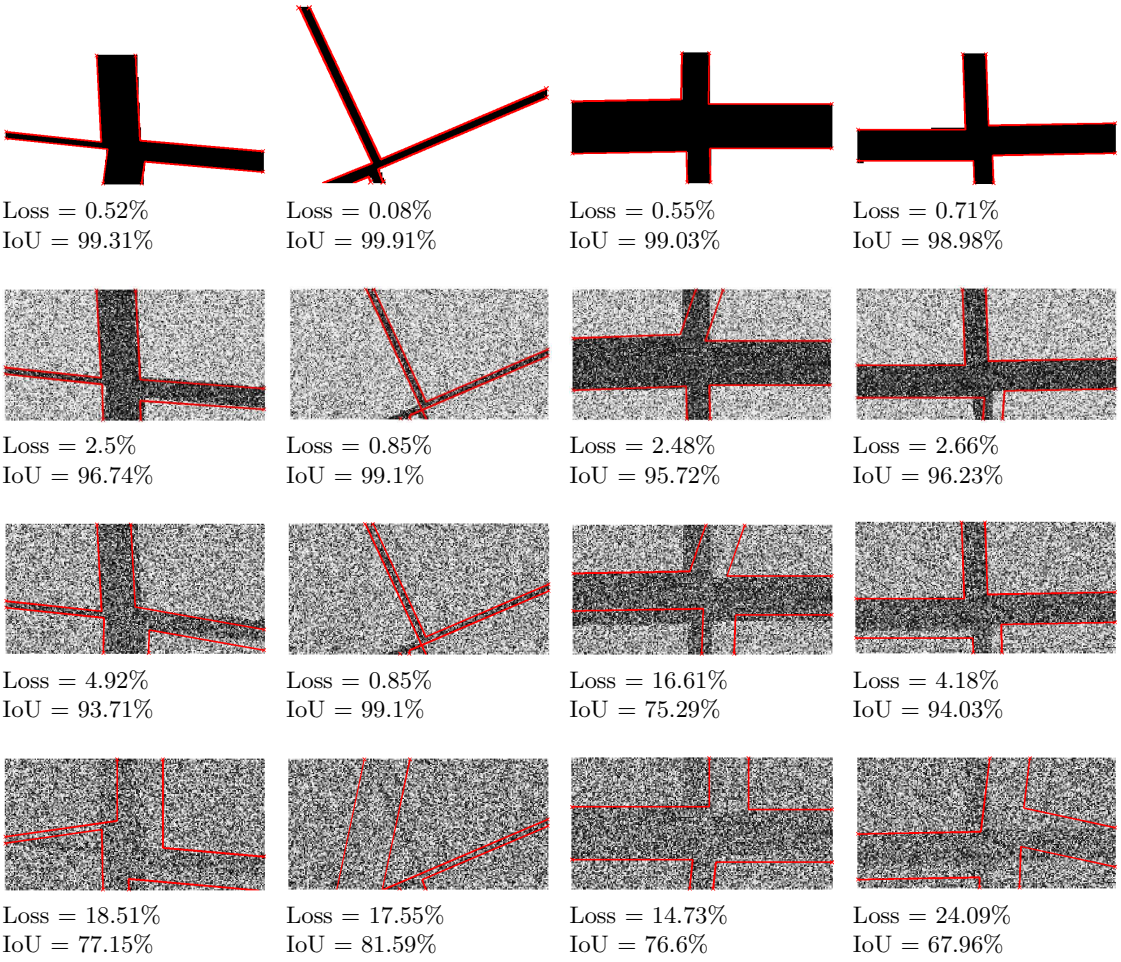


Figure 3.19: Continuation of Fig. 3.18

From the results in the table and figures, it can be observed that on average the pixelwise error increases for higher levels of random noise and, similarly, the intersection over union ratio decreases. Despite the noisy images for the level 0.8, our model is able to achieve a mean IoU near to 80%. Besides, it can be also seen an increment on the average inference time, which corresponds to the enlargement of the queue in the BB algorithm. In fact, the bounds for each hypotheses of the search space become more similar and less discriminative. Hence, the algorithm has to explore more hypotheses to find the global maximum.

3.4.4.2. Tests on real BeP images

In relation to the real dataset, our proposed approach is not able to infer the 4 intersecting roads as it is displayed in the samples of Fig. 3.20. The main cause is the sparsity of the 2D occupancy grid maps (BeP images). As a matter of fact, it is very difficult to infer the existence of an intersection from a human point of view. Thus, teaching machines to do it without additional cues, is also very challenging. Although road pixels (in black) are more abundant, there is usually few occupied grids (in white) on the left and right sides of the street with the ego-vehicle. Therefore, our model can not deal with missing information, but we already demonstrated that is robust to moderate levels of random noise.

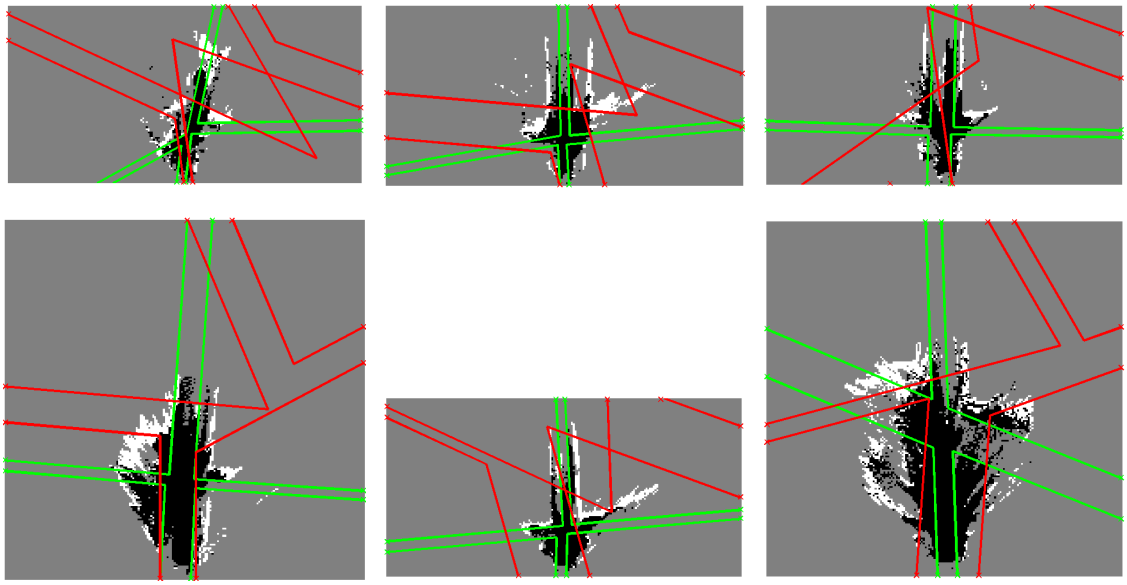


Figure 3.20: Predictions results for 4 intersecting roads on real BeP images. The ground-truth layout is painted in green and the predicted one in red color. It can be observed that even for the least sparse and least noisy 2D grid maps of the dataset, inferring the layout is very complicated and the algorithm is not able to find the correct solution.

Moreover, with the aim of building enhanced BeP images from the real data, we have carried out a set of experiments adding random pixels from the ground-truth synthetic images. The next Fig. 3.21 depicts the predicted intersection layout for 8 samples and three cases: a) populate the real BeP image with 5% of pixels from white regions in the corresponding synthetic image, b) same as before, but using 10% of random pixels and, c) populate the real BeP image with 5% of pixels from white and black regions in the corresponding synthetic image.

In Fig. 3.21, the rows 3, 4, 5 and 7 show cases where the addition of pixels from the synthetic ground truth clearly help the prediction of the intersection layout. The second row depicts a case with very similar predictions for the three tests configurations, which is explained by the poor occupancy grid map in terms of discriminative features. Indeed, in the original BeP image, there are very few occupied grids (white) and the free-area grids (black) did not visually suggest the existence of an intersection. Thus, the addition of white pixels from the ground truth bounds the intersection geometry. On the other hand, the examples on the first and last rows are gradually approaching to the correct solution as far as more ground-truth data is added (from first to third columns). However, the optimal solution is not predicted due to the strong upper white regions and the lack of them in the bottom of the original grid maps. Finally, the layout of the sample in the sixth row is well approximated when a 10% of white pixels from ground truth are added. However, it does not work well for the configuration in the third column. Basically, there are a big number of free-area grids (black) in the real BeP image, thus, the addition of more 'white cues' is more important than the addition of black and white pixels together.

From these experiments, it can be concluded that enhancing the reconstructed occupancy grid maps is complex, difficult to automate and to generalize for every observed scene. Some cases have a lack of occupied grids, while others have missing information of the navigable area.

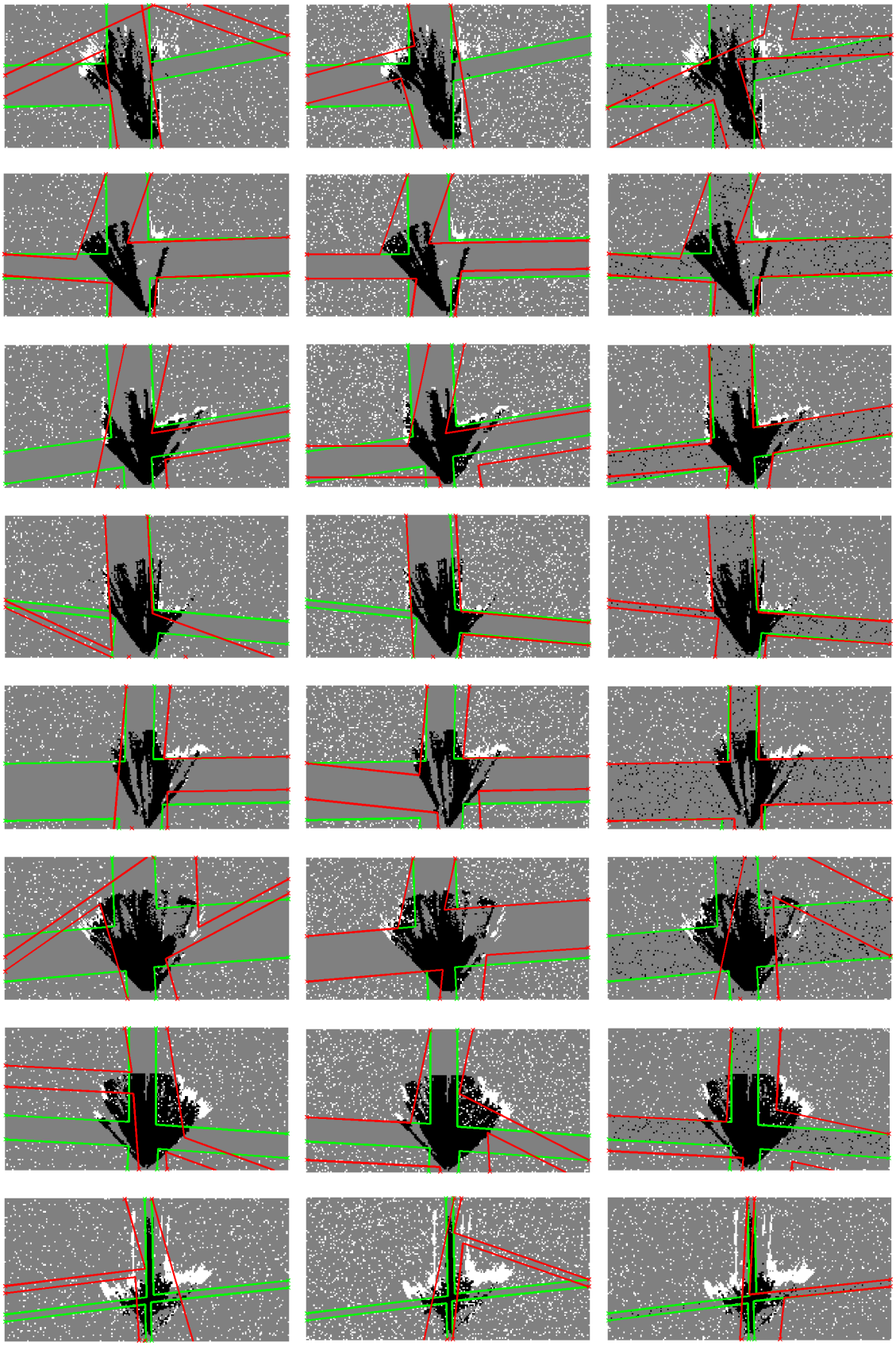


Figure 3.21: Predictions results for 4 intersecting roads on enhanced real BeP images. The ground-truth layout is painted in green and the predicted one in red color. There is a different sample from the dataset on each row. The three columns correspond to the a), b) and c) cases described in the text, which are: the addition of 5% of pixels from white synthetic regions, 10% of pixels from white synthetic regions and 5% of pixels from black and white synthetic regions.

3.5. Conclusions

To sum up, this Chapter has presented the methodology for learning and inferring the road scene layout, in particular the geometry of intersections from occupancy grid maps (or Bird's eye Perspective images) recovered from stereo sequences. To the best of our knowledge, the proposed structured prediction is the first discriminative approach in the state of the art, devised as an alternative to the generative model in [Geiger et al., 2011a]. Besides, we have employed less cues from the sequences, i.e. only the occupancy maps without flow vectors or tracklets [Geiger et al., 2011b], which made the task more complicated for predicting the road boundaries.

The supervised learning of the models has been carried out based on Conditional Random Fields (CRF) [Hazan and Urtasun, 2010], whilst inference has been tested with two different algorithms: an approximate inference approach based on distributed convex Belief Propagation (dcBP) [Schwing et al., 2011] and an exact inference approach based upon Branch and Bound (BB) [Schwing and Urtasun, 2012]. Several parameterizations have been studied in order to find a suitable model for inferring the intersections geometry. Moreover, a set of *leave-one-out* cross-validation experiments have been carried out on synthetic and real data to evaluate the proposed graphical models. In fact, the model for straight roads (Section 3.2) has demonstrated good layout predictions on synthetic and real BeP images and we have shown the runtime benefits of using BB vs dcBP.

With regard to 4 intersecting roads, the approaches in Appendix A, while tested with the largely employed CRF of [Hazan and Urtasun, 2010] and the dcBP inference engine [Schwing et al., 2011], they have yielded poor prediction results. Among the main problems that were found, we can mention: wrong learned parameters in the form of swapped signs and unbalanced weights, no reaching agreement during inference, wrong predictions after convergence of the inference task and high pixel-wise errors employing the synthetic images. Consequently, it has been difficult to extend the model for the real dataset given the noisy BeP images reconstructed from stereo.

However, the BB approach in Section 3.3 has successfully worked on synthetic BeP images of 4-roads. We have conducted a set of experiments with artificially generated noise over these images achieving low average pixelwise loss (<9%) and high IoU ratios (90%) for moderate levels of noise. It must be remarked the dimensionality of the problem (514^4 different 4-roads layout hypotheses) and, despite the grids sparsity, our proposed model has been able to obtain good approximations to intersection layouts with low levels (5-10%) of leading data randomly selected from ground-truth synthetic images. Besides, some of the predictions in Fig. 3.21 suggest the adaptation capacity of our approach to infer 3-roads intersections, which is an extension for the future.

Therefore, these experiments using synthetic images have shown the feasibility of our proposal when more accurate measurements and better reconstructed occupancy grid maps are available.

Chapter 4

Supervised learning of object classes from 2.5D appearance

As already pointed out in Chapter 2, nowadays, 3D image understanding is of great interest in the area of autonomous robotics platforms, e.g. autonomous/intelligent vehicles. In previous chapter, we proposed a 3D scene understanding approach for inferring roads geometry, i.e. intersections layout, based on stereo sequences. Complementary, this chapter focus is on predicting relevant object instances contained in the road scenes from color and disparity information. Although geometry and objects are treated separately in this Thesis, their joint labeling is a follow-up path to leverage the 3D urban scene understanding [Geiger et al., 2014].

Object detection from images has been under active research since 1970s. However, inferring the location and orientation of objects for autonomous robotic platforms is still an open problem [ICCV Workshop, 2013]. Indeed, there is a strong research interest on the object classes 'car', 'pedestrian' and 'cyclist' [Geiger et al., 2012] in order to provide more accurate predictions of these object instances in complex, dynamic and naturalistic urban scenarios. On the other hand, part-based detectors [Felzenszwalb et al., 2010b] have been successfully tested on image classification, segmentation and retrieval tasks [Everingham et al., 2010]. Thus, they are good candidates to be extended for the 3D urban scene understanding challenges. More importantly, the existence of public datasets and common evaluation metrics are the key for advancing the performance of visual recognition systems in this context.

This chapter tackles three main lines with the aim of increasing the accuracy of the bounding box predictions for cars, pedestrians and cyclists in urban road scenes: (1) an extension of DTPBM to account for 2.5D information extracted from stereo disparity maps, (2) a detailed analysis of the supervised parameter learning and (3) additional approaches with the aim of improving object detection. Furthermore, Chapter 5 deeply analyzes two state-of-the-art evaluation protocols ([Everingham et al., 2010, Geiger et al., 2012]), showing subtleties that are very relevant for results assessment and comparison.

Firstly, the problem description and the DTPBM framework are introduced, then the following sections deal with all the research carried out and the new contributed 3D-aware features.

4.1. Introduction

4.1.1. Problem description

Similar to the scene layout approach in previous chapter, let us consider a moving observer (e.g. robot, mobile platform, vehicle) with a stereo camera on it, which is navigating through structured and non-open spaces, such as streets inside cities or interurban roads. In this context, we are interested in the *object detection and orientation estimation* challenge proposed in [Geiger et al., 2012]. In particular for the object classes 'car', 'pedestrian' and 'cyclist'. The KITTI Vision Benchmark Suite [KITTI, 2012] is publicly available and the website provides a common panel for research and for comparison of results. The dataset has been collected in urban and inter-urban naturalistic environments employing an autonomous driving platform, which provides stereo images in color and gray scale, positioning data from an inertial navigation system (GPS/IMU) and dense 3D point clouds from a Velodyne Laserscanner. Although the Velodyne provides dense point clouds that can be used for scene understanding in conjunction with appearance data from the images, it is a more expensive solution for integration in autonomous vehicles. Thus, we trust on the benefits of deploying stereo cameras in the vehicles for visual recognition. Basically, with two cameras one can have access to color and depth data of the scene, which is usually known as *2.5D data*. Supporting this, the 3DV commercial system has recently emerged as a complete stereo solution ready for research labs and automotive industry [Vislab.it, 2014]. Thus, this grants a promising future for the research and applications on 3D outdoor scene understanding from 2.5D measurements.

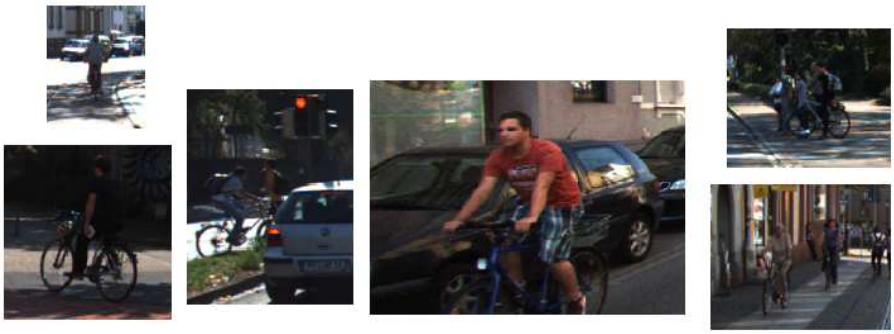
Therefore, we will make use of the stereo color images from KITTI, which consist in a large set of stereo rectified frames that have been randomly picked from several video sequences. They are divided into training and testing subsets and Fig. 4.1 shows some examples of the left-camera images. A thorough review of the dataset characteristics is in Section 5.1.

Figure 4.1: Left-camera image samples from KITTI dataset. Upper images from the training with ground-truth labels and lower images from the test set.

Many challenges arise from the displayed images, i.e. object detection under occlusion and background clutter, orientation estimation from different viewpoints, detection at different scales, object truncation, varying illumination conditions, shadows, color differences between objects of the same class, etc. Fig. 4.2 illustrates some examples of these cases for each category.

(a) cars

(b) pedestrians



(c) cyclists

Figure 4.2: Examples of challenging object instances in the KITTI dataset.

To approach the object detection and orientation estimation challenge, few methods have reported results on [KITTI, 2012] since the publication of the dataset [Geiger et al., 2012]. Excluding *mBoW* [Behley et al., 2013], which employed laser data, the remaining proposals up to April 2014[1] relied only on visual appearance from color [Pepik et al., 2013, Geiger et al., 2011b]. This Thesis contributes with the addition of 3D cues from the stereo images to improve object prediction ratios, being the first approach based on stereo data and published in KITTI website.

In order to obtain 3D data, disparity maps have to be computed, but they are not provided in [KITTI, 2012]. Although *LIBELAS* [Geiger et al., 2010] method could be employed for doing

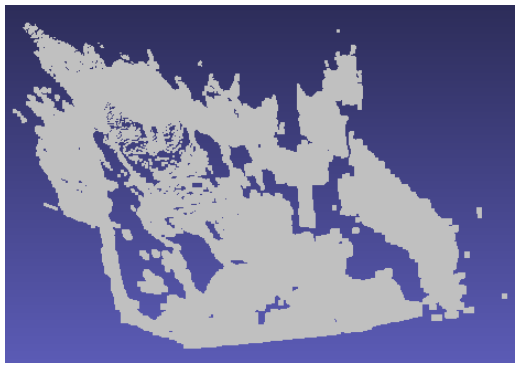
[1]Several works have been showing results in KITTI website in the first half of 2014 and they have been marked as anonymous submissions because they are pending to publication in different conferences or journals. Hence, we can only mention their reported names: SubCat, SVM-Res, DA-DPM. Further details in Section 5.3.4.

it, the dataset is composed of a random selection of frames without temporal sorting. Thus, computing disparity from all the video sequences and then cross-referencing every frame is out of the scope of prediction tasks in this Thesis. Consequently, we opt to calculate disparity maps from each pair of left-right images based upon the well-known *Semi-Global Matching (SGM)* method [Hirschmuller, 2008], which provides a good average performance according to the ranking in the stereo benchmark [KITTI, 2012]. Future trends in the state of the art base their proposals on optical flow [Yamaguchi et al., 2013], which can be considered as a further enhancement for the future.

(a) Left-camera image

(b) Right-camera image

(c) Disparity map



(d) Recovered 3D point cloud

Figure 4.3: 3D reconstruction of a sample urban scene from KITTI.

On the one hand, we could approach the addition of 3D cues from the 3D reconstructed scene, as it is depicted in Fig. 4.3, where the bottom representation is the recovered point cloud up to an unknown scale, given the calibration parameters provided in KITTI. Intuitively, only some parts of scene structure and some perspective projection effects can be visually perceived. On the other hand, cars and cyclists can be clearly identified from the disparity image (in the center). As a matter of fact, carrying out 3D reasoning directly on the point clouds from these sparse images is a complex task. Searching for a 3D cuboid that surrounds every object requires a computationally demanding learning and inference processes plus the addition of assumptions and tight constraints. Some approaches have faced it directly from monocular

images [Fidler et al., 2012], based upon CAD prior models [Pepik et al., 2012b] or using dense laser data [Behley et al., 2013].

In order to evaluate the feasibility of 3D object instances from the point clouds recovered from stereo, we have carried out a tedious manual segmentation in some training images. As a result, we were able to obtain some recognizable 3D objects (Fig. 4.4) for the closer and most contrasted instances. However, the vast majority of the dataset comprises noisy samples as the ones depicted in Fig. 4.5. This is due to the sparsity and small errors from disparity, which cause large depth estimations ($Z_d \propto \frac{1}{D}$). Hence, automatically picking the 3D points corresponding to the 2D bounding box ground truth adds many noisy 3D points, as it is demonstrated on the unfiltered point clouds in Fig. 4.4.

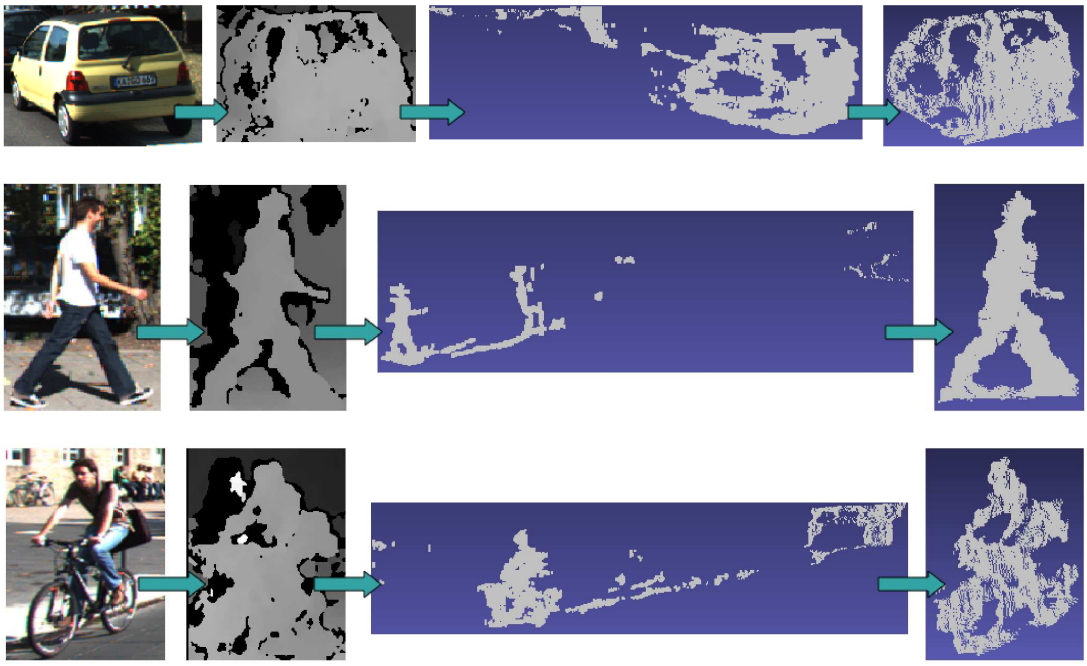


Figure 4.4: Manually segmented objects from the 3D reconstructed scenes. From left to right: Color image patches, disparity map regions in gray scale, original point clouds recovered from them and manually segmented objects. The point clouds are directly obtained with the reprojected pixels of the object bounding box and manually filtered afterwards. As can be seen, the 3D point clouds before manually filtering contain large depth deviations associated to small errors in disparity. Therefore, collecting a clean training dataset cannot be carried out by simply reprojecting the image pixels inside the ground-truth boxes.

Consequently, our intuition is that, for the target goal of improving accuracy in object detection and orientation estimation, 3D point clouds may add more noise. Then, disparity is preferred over the clouds because it carries the same information about objects, the errors do not generate large deviations and the gradients can be discriminative features. Some works have already demonstrated the benefits of adding disparity in visual recognition tasks to increase detection performance [Walk et al., 2010, Makris et al., 2013]. Therefore, our main aim is to learn better and richer models employing 2.5D data that can yield more accurate predictions of the objects in stereo scenes.

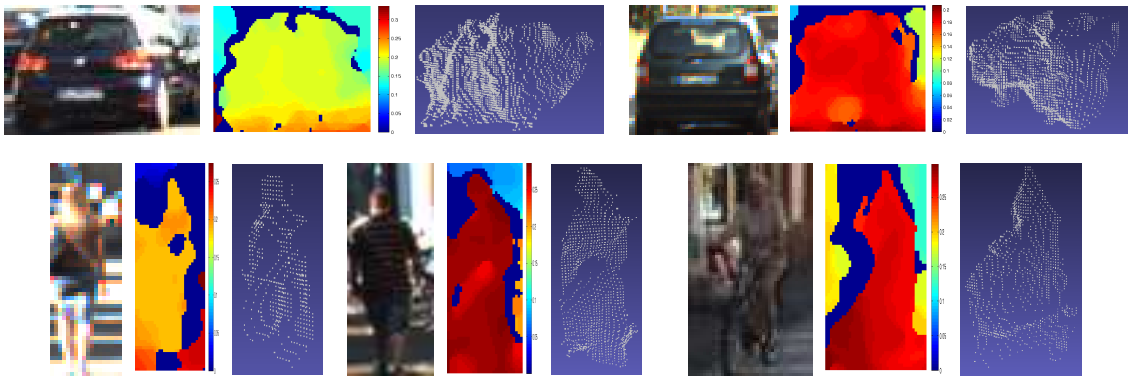


Figure 4.5: Manually segmented objects from the 3D reconstructed scenes. These examples depict sparse point clouds that have been filtered from the noisy clouds recovered from disparity. These cases are very common within the labeled objects of KITTI dataset. For reference, each instance is shown in three representations: the color image patch, the related disparity in a color scale and the reprojected 3D point cloud. If we think on disparity gradients, it is more discriminative to employ the disparity patches than the reprojected 3D point clouds.

4.1.2. Problem formulation: DPM framework

Since the successful application of SVM to pedestrian detection in the feature space of HOG [Dalal and Triggs, 2005], several approaches have been proposed for the discriminative task of object recognition. As already mentioned in Section 2.3, pictorial structures were devised in the 1970s, but in terms of performance, they were not demonstrated in practice until the Discriminatively Trained Part-Based Models [Felzenszwalb et al., 2010b]. The merits of this work were already introduced in Section 2.3. In fact, part-based models have been successfully applied on generic datasets not limited to road scenarios like [PASCAL VOC, 2012], but also in KITTI [Geiger et al., 2011b, Geiger et al., 2012]. Another advantage is that they only need ground-truth labels of the objects bounding boxes during training, thus, not requiring costly annotations for the object parts. Actually, human manual selection of the relevant parts, may not correspond to the most discriminative parts. Thus, this is a job for machine learning.

Hence, our baseline will be the DTPBM or commonly referred to as DPM in the literature. Most of the recently published works in [KITTI, 2012] rely on modifications on top of it, but they have not exploited the use of 2.5D data and they have not provided detailed considerations on the supervised training for different setups. This chapter of the Thesis, in conjunction with the large set of experiments in Chapter 5, fill this gap.

DPM classifies and locates objects at different scales based on a pyramid of appearance features at different resolutions, i.e. a scale pyramid of modified HOG descriptors [Dalal and Triggs, 2005]. Besides, it has been successfully tested not only in categorization, but also in segmentation, person layout and action classification tasks. It also provides open source code for the general public and this Thesis has carried out a deep analysis to understand the underlying mechanism and to contribute with several modifications on top of its release 4 [Felzenszwalb et al., 2010a]. Consequently, we will provide here a mid-sized review of DPM [Felzenszwalb et al., 2010b] that will facilitate the understanding of the contributions.

Following the notation previously introduced in Chapter 3 for CRF and the structured

prediction approach in [Pepik et al., 2012b], the DPM can be viewed as a mixture of CRF models (one for every object viewpoint), where each of them presents a star topology as depicted in Fig. 4.6 and the pictorial structure in Fig. 4.7. Therefore, the model of an object consists in several parts, which are formally defined as the discrete random variables p_i.

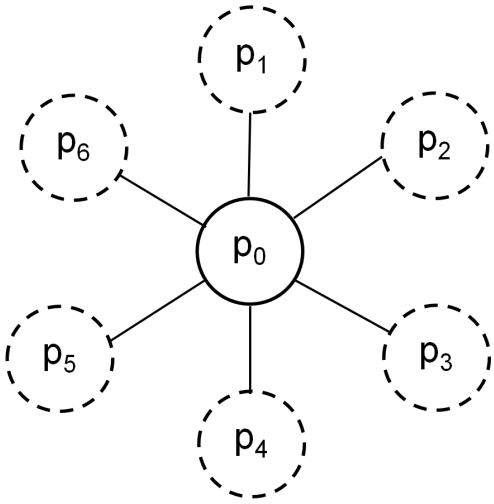


Figure 4.6: Undirected graphical model of one object viewpoint in DPM. This is a sample graph with 6 object parts. The *root* part (p_0) of the object is defined as the 2D bounding box around it. The remaining 2D parts (p_i, $i = 1, .., 6$) are hidden/latent variables (dashed nodes) determined with a maximization goal. This star topology represents the spring-like connections between the parts and the principal bounding box. They are dependent on elasticity constraints that allow the models to adapt to intraclass variation.

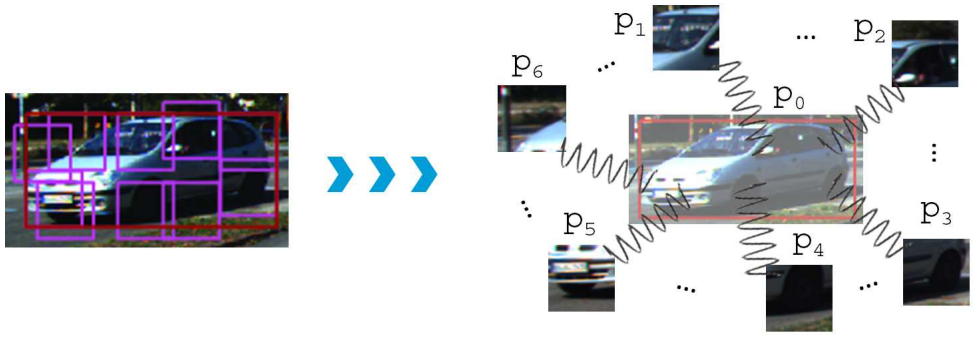


Figure 4.7: Pictorial representation of the spring-like connections between the root and the remaining parts of the object model. In the drawing, only 6 parts are shown for simplicity.

Every object part is defined as hidden because they have not been annotated in the dataset, i.e., we only know the ground-truth labels for the 2D bounding boxes around the object. Every part is defined as $p_i = (u_i, v_i, l_i)$, which corresponds to the upper-left corner coordinates (u, v) in pixels that locates the part in the image and the scale level l in the feature pyramid, which is built as explained in [Felzenszwalb et al., 2010b]. The location, scale and size of p_0 are given by the ground truth during training, but they are predicted when searching for the objects in the test images. The number of parts and their size are fixed during initialization. Besides, due to the latent/hidden nature of the parts, (u_i, v_i, l_i) have to be estimated during both learning and inference. Mathematically, they are determined by exploring the energy in Eq. 4.1.

$$score_{\boldsymbol{\theta}}(p_0) = \max_{z \in Z(\mathbf{x})} \boldsymbol{\theta}^T \psi(\mathbf{x}, z) \tag{4.1}$$

In the equation above, $\mathbf{x}$ denotes an input image, $\boldsymbol{\theta}$ is the parameter vector, ψ is the potential with the visual features of hypothesis $z = (p_0, ..., p_n)$ computed on $\mathbf{x}$ and their product resembles

an energy function, like in Chapter 3. Particularly, Eq. 4.1 is the formulation of the *Latent Support Vector Machine (LSVM)* proposed in [Felzenszwalb et al., 2010b]. From the set of possible object parts configurations $Z(\mathbf{x})$, the selected hypothesis $z = (p_0, \hat{p}_1, ..., \hat{p}_n)$, exhibiting the maximum energy, provides the score for a 2D patch (p_0) in the image.

Next, the energy of the conditional distribution ψ can be separated into unary and pairwise potentials as already introduced in Eq. 3.6. In particular, we can establish the following general formulation for the DPM approach:

$$\boldsymbol{\theta}^T \psi(\mathbf{x}, z) = \boldsymbol{\theta}_i^T \phi_i(\mathbf{x}, z) + \boldsymbol{\theta}_\alpha^T \phi_\alpha(\tilde{z}) \tag{4.2}$$

The unary terms ϕ_i describe the appearance of each object part using HOG features computed on the image $\mathbf{x}$. They can be seen as the concatenation of the HOG features for the subwindows and pyramid scales indicated by each hypothesis z. Similarly, the parameter vector $\boldsymbol{\theta}_i$ can be seen as the concatenation of the learned filters for every part. Besides, the pairwise potentials ϕ_α (previously represented in Fig. 4.6) encode the 2D distances of the parts with respect to the root p_0, which are obtained as a descriptor of four elements $< du_i, dv_i, du_i^2, dv_i^2 >$ (check [Felzenszwalb et al., 2010b] for further details). Next, we refer to $\tilde{z}$ as the pairwise hypotheses (p_0, p_i) that connects each part to an anchor in the root window. Then, the parameters $\boldsymbol{\theta}_\alpha$ contain the four learned weights for every part spring.

If we dive deeper in the formulation of the DPM method, we will review here the scoring function in [Felzenszwalb et al., 2010b] that defines the energy exposed above. Eq. 4.3 presents the score that is computed for every hypothesis $z \in Z(\mathbf{x})$, considering only 1 component of the mixture of CRFs, being n the total number of objects parts p_i.

$$s(z) = \sum_{i=0}^{n} F_i \cdot \phi_i(\mathbf{x}, p_i) - \sum_{i=1}^{n} \mathbf{d}_i \cdot \phi_\alpha(du_i, dv_i) + bias \tag{4.3}$$

F_i represents all the learned weights of the root and part 2D filters, which are concatenated as the unary parameters $\boldsymbol{\theta_i} = (F_0, ..., F_n)$. Besides, $\mathbf{d}_i$ are the learned deformation weights, which are concatenated as the parameters $\boldsymbol{\theta_\alpha} = (\mathbf{d}_1, ..., \mathbf{d}_n)$. Therefore, the total parameter vector consists in $\boldsymbol{\theta} = (\boldsymbol{\theta_i}, \boldsymbol{\theta_\alpha}, bias)$. It must be noted that the part filters $F_i \setminus i \neq 0$ and their relative displacements (du_i, dv_i) are estimated at twice the resolution of the root scale in the pyramid, as originally constrained in [Felzenszwalb et al., 2010b]. In fact, we can imagine the root filter as a 2D coarse sketch of the object and the part filters like a finer picture of the object (see Fig. 4.8 for an example).

Therefore, every 2D bounding box hypothesis p_0 (given by the ground truth during learning or by an image search algorithm, e.g. sliding window, during object detection) will have an associated score obtained from the best configuration of parts with respect to the root. This is given by Eq. 4.4, which is internally computed employing dynamic programming and *generalized distance transforms* [Felzenszwalb et al., 2010b].

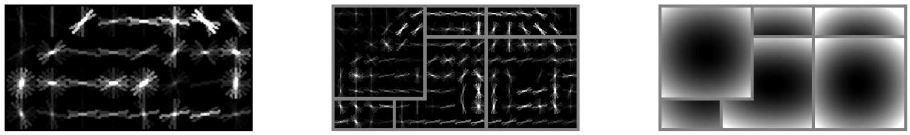
$$score(p_0) = \max_{p_1,...,p_n} s(p_0, ..., p_n) \tag{4.4}$$

Regarding to the orientation estimation of the objects, DPM has been originally conceived as a mixture of models, such that an additional discrete random variable c is defined to account for the component of the mixture. Therefore, extending the part-based detector to object orientation prediction is direct. In fact, each component is trained for one different object viewpoint, which is an approach that has been also followed by [López-Sastre et al., 2011, Geiger et al., 2012, Pepik et al., 2012b].

As a consequence of this model extension, the dimensionality is approximately[2] multiplied by a factor equal to the number of states of c. This increases the computational complexity of both tasks, learning and inference. Basically, the total parameter vector can be denoted as $\boldsymbol{\beta} = (\boldsymbol{\theta}^1, ..., \boldsymbol{\theta}^{nc})$, but we can still represent the scoring function in Eq. 4.1 as a product, particularizing for one component $\boldsymbol{\beta}_c = \boldsymbol{\theta}^c$. Hence, we will obtain scores for each new hypothesis defined as $z' = (c, p_0, ..., p_n)$ and ψ being the concatenation of visual and displacement features.

$$score(c, p_0) = \max_{p_1,...,p_n} s'(c, p_0, ..., p_n) = \boldsymbol{\beta}_c^T \cdot \psi(\mathbf{x}, z') \tag{4.5}$$

Figure 4.8 shows three examples of the models learned with DPM. The learned appearance parameters F_i are represented as the positive and normalized oriented gradients (HOG descriptors [Dalal and Triggs, 2005]). Besides, the displacement parameters $\mathbf{d}_i$ are depicted as 2D energies that votes the possible locations of the parts with respect to the anchors in the root window. These energies can be seen as elasticity constraints.



(a) Cars from viewpoint $5\pi/8$ rad

(b) Pedestrians from viewing angle $3\pi/4$ rad

(c) Cyclists with orientation $-\pi/4$ rad

Figure 4.8: Examples of the learned weights. From left to right: root filter F_0, part filters F_i at twice resolution and the cost of parts placement relative to anchors in the root window.

[2]Each component has distinct number of parameters due to different aspect ratio and dimensions for the root filter F_0^c, which may also be the case for the remaining object parts, depending on initialization.

Once defined the model of an object and the scoring function that votes every hypothesis in the DPM framework, we will briefly introduce the two main phases of any classification problem: 1) model learning and 2) object inference. The schematic representations in Fig. 4.9 and Fig. 4.10 illustrate the two phases when employing 2.5D data as proposed in this Thesis.

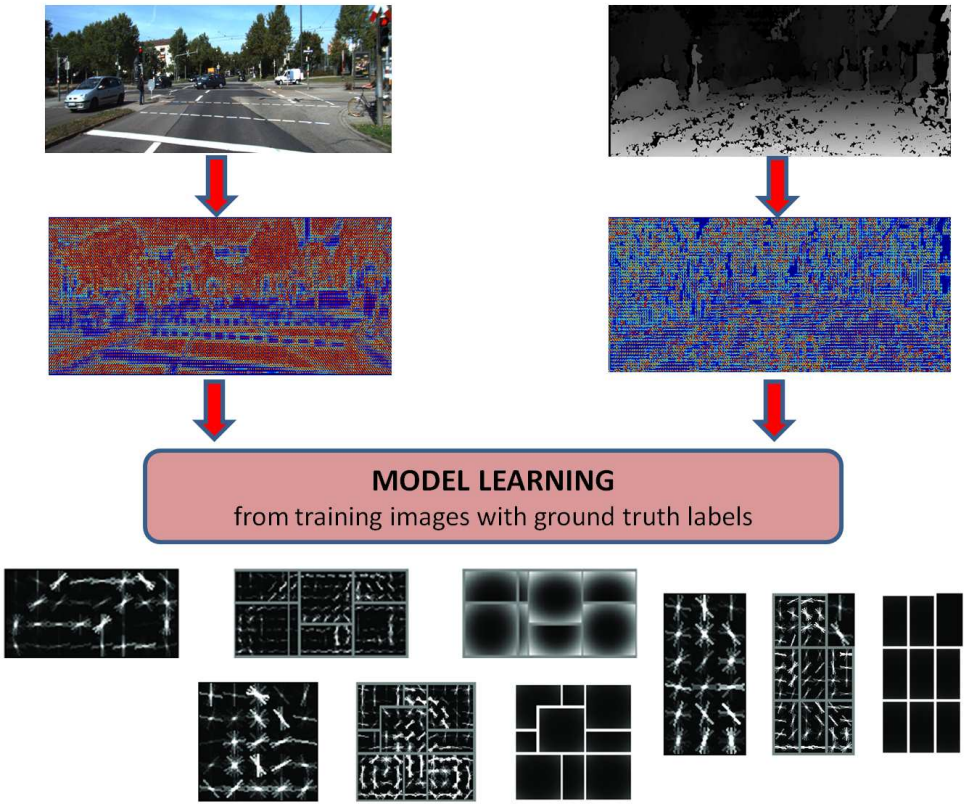


Figure 4.9: Scheme of the learning phase when using 2.5D data.

Learning. In DPM, the parameter vector $\boldsymbol{\beta}$ is learned by training a LSVM classifier, where the latent variables z consists of the model component c (object viewpoint) and the location (u_i, v_i) and scale l_i of the compositional object parts. This entails a non-convex optimization problem that is converted to convex exploiting two ideas: the semiconvexity of the SVM hinge loss for the negative training samples and the restriction to a single possible latent value for each positive sample. The optimization is solved through many iterations that are split in two steps: 1. the coordinate-descend approach that relabels positive samples and 2. the stochastic gradient-descent algorithm that optimizes the parameter vector [Felzenszwalb et al., 2010b]. As a result, this process yields the highly-dimensional weight vector $\boldsymbol{\beta}$, whose length depends on the number of components, the number and size of part filters F_i and the length of the visual features. For example, for the class 'car', considering 16 mixture components where each of them has 1 root part (variable size depending on aspect ratio), 8 subparts of fixed size (6×6) and a normalized gradient descriptor of 32 dimensions, the total number of parameters to be learned is 170,624. An illustration is on Fig. 4.8.a.

Furthermore, a *bootstrapping* strategy is followed for data-mining hard negative samples during the optimization of $\boldsymbol{\beta}$. In few words, a first set of negatives (image subwindows not

labeled in the ground truth as object instances) is selected for training. Then, on each new model adjustment, the negative samples that are incorrectly classified are collected to form the subset of hard negatives for re-training the model on the next iteration.

In relation to the feature extraction process, a scale pyramid of HOG descriptors is computed for every positive and negative sample for the detection and scoring of the latent variables z. The number of octaves, i.e. scales between a given size and half its resolution, is configurable but we have kept the default $\lambda = 5$.

Attending to the implementation details, 4 main stages can be differentiated while training:

I.- Model initialization (number of parts, root filters size, etc.) and individual components training based only on root parts.. The positive samples are warped to fit the size of the initialized filters and several negatives are randomly chosen from negative images. Algorithm 1 shows this stage in pseudocode.

II.- When using bilateral symmetry for an object class, this second stage trains the left instances vs the right instances of the root filters. However, we omit further details because we will not employ it, as appointed in next Section 4.3.

III.- The individual model components are merged together into the mixture. Then, all of them contribute during the parameter learning. Moreover, it considers latent detections for the object bounding boxes, with the aim of providing a certain degree of flexibility in the location and size of the root parts with respect to the given ground-truth labels. It also collects hard negatives from negative images. This is reflected in Algorithm 2.

IV.- Finally, the parts are added and initialized and the model is re-trained considering the latent variables and the bootstrapping algorithm. Algorithm 3 provides a schematic view of this stage, but further details can be found in [Felzenszwalb et al., 2010b].

Algorithm 1 DPM training stage I. Initializing and learning individual model components.

Input: P set of positive images with ground-truth bounding boxes.
Input: N set of negative images.
Output: $\boldsymbol{\beta}_1, ...\boldsymbol{\beta}_{Nc}$ set of parameter vectors for each component of the mixture model.
Algorithm:

```
S_p := ∅; S_n := ∅;                          ▷ Positive and negative HOG feature samples
for c in Nc do                               ▷ Iterate through a fixed number of viewpoints
    β_c := initmodel(P);                     ▷ Init size of root filters p_0^c and the model structure
    S_p := poswarp(P, β_c);                  ▷ Warp positive patches to the filter size and compute features
    S_n := negrandom(N, β_c);                ▷ Collect random background samples and compute features
    β_c := StochasticGradientDescent(S_p, S_n, β_c);
end for
```

Algorithm 2 DPM training stage III. Merge individual models and train with latent root detections and hard negatives.

Input: P set of positive images with ground-truth bounding boxes.
Input: N set of negative images.
Input: $\boldsymbol{\beta}_1, ...\boldsymbol{\beta}_{Nc}$ set of parameter vectors for every model component.
Output: $\boldsymbol{\beta}_m$ parameter vector of the initialized mixture model.
Algorithm:

```
S'_p := ∅; S'_n := ∅;                ▷ Positive and negative HOG features in the scaled pyramid
β_m := mergemodels(β_1, ..., β_Nc);
S'_p := poslatent(P, β_m);           ▷ Assuming p_0 as latent variable. Detect high scoring
                                     ▷ 2D boxes that overlap positive samples and compute features
for j = 1 → Ndm do                   ▷ Hard-negative data mining iterations
    S'_n := neghard(N, β_m);         ▷ Collect hard negatives and compute features
    β_m := StochasticGradientDescent(S'_p, S'_n, β_m);
end for
```

Algorithm 3 DPM training stage IV. Add latent parts and learn the final mixture model.

Input: P set of positive images with bounding boxes from ground truth.
Input: N set of negative images.
Input: $\boldsymbol{\beta}_m$ parameter vector from previous stage.
Output: $\boldsymbol{\beta}$ final mixture model.
Algorithm:

```
S'_p := ∅; S'_n := ∅;                ▷ Positive and negative HOG features in the scaled pyramid
β = β_m;
for c in Nc do
    β := addparts(β);                ▷ Add parts and initialize their location and size
end for
for i = 1 → Ncd do                   ▷ Coordinate-descent iterations for positives relabeling
    S'_p = poslatent(P, β);          ▷ Search for latent parts location (see Eq. 4.5)
    for j = 1 → Ndm do               ▷ Hard-negative data mining iterations
        S'_n := neghard(N, β);       ▷ Collect hard negatives and compute features
        β := StochasticGradientDescent(S'_p, S'_n, β);
    end for
end for
```

Inference. For predicting the 2D bounding boxes around the objects (cars, pedestrians or cyclists) in unseen images/scenes, a feature scale pyramid is built and walked through to generate the set of hypotheses, as depicted in Fig. 4.10. Then, the score of every hypothesis is obtained from Eq. 4.5 and applying the matching process in [Felzenszwalb et al., 2010b]. In fact, this process can be seen as a convolution between filters F_i and the image features at different scales. Additionally, a minimum threshold rejects the predictions with lower confidence[3]. This detection process usually generates many hits around the same object, but with slight changes in scale and location. Consequently, a maximum suppression filter sorts the scores of the candidates boxes in decreasing order and removes the overlapping candidates that does not fulfill a maximum overlap requirement (e.g. 50%).

[3]This threshold is automatically estimated after training but can be also empirically adjusted. It only affects the final results in terms of false positives reduction. In fact, the threshold reduces the number of false positives per image at high recalls, which corresponds to the last part of the precision-recall curve where precision plummets.

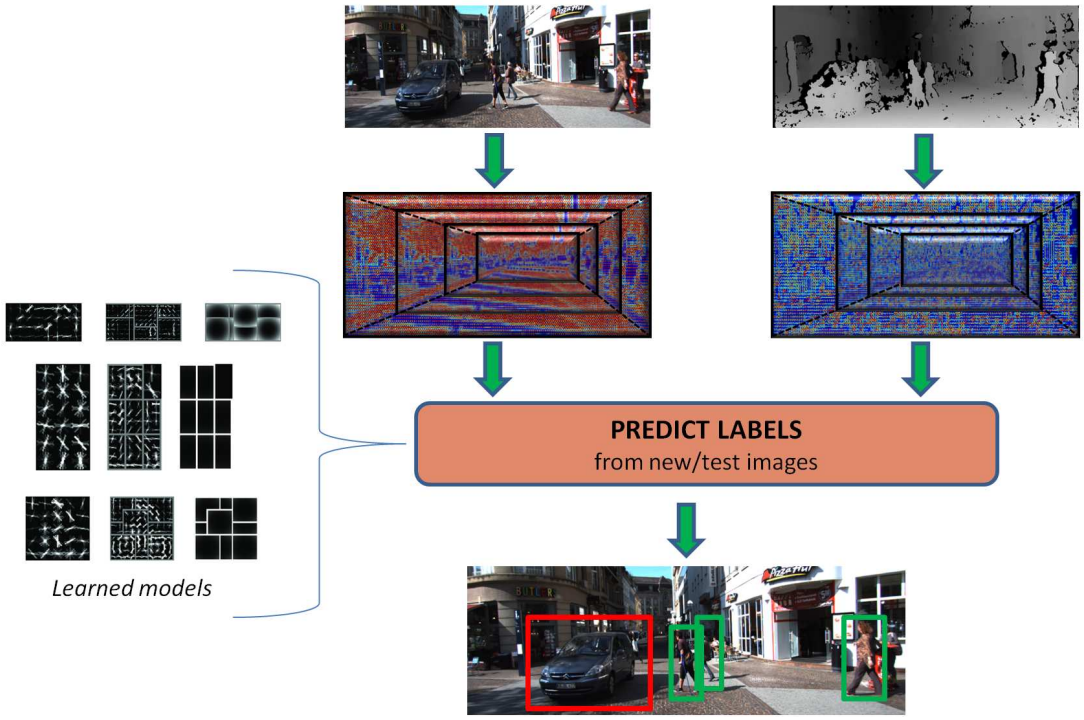


Figure 4.10: Scheme of the inference phase when using 2.5D data.

4.1.3. Goals

As introduced in previous Section 4.1.1, this chapter of the Thesis tackles the object detection and orientation estimation challenge defined in [Geiger et al., 2012, KITTI, 2012]. This involves the following specific goals:

- Study and analysis of the DPM framework to jointly solve the tasks of 2D bounding box prediction and orientation estimation of objects in urban environments, particularized for the categories 'car', 'pedestrian' and 'cyclist'.

- Addition of 3D cues from stereo images that carry information from appearance and depth of objects parts in the scene. New contributed 3D-aware features that capture 2.5D data.

- Supervised learning of richer models from 2.5D data (color and disparity) that generalize well to unseen images. Tuning of the DPM training pipeline as a key process during supervised learning, based on cross-validation rounds to prevent overfitting.

- Proposal and analysis of two additional approaches on top of DPM: whitening and stereo consistency check.

- Improvements in the precision-recall curves that measure the performance of the trained models, employing first the validation subset and then, the testset for publishing the results in the KITTI website and for ranking our method among the state-of-the-art works.

4.2. 3D-aware features

The previous section introduced the DPM framework where we briefly talked about HOG descriptors [Dalal and Triggs, 2005] as the underlying visual feature. On the other hand, one of the principal aims of this Thesis is the employment of 2.5D data to augment the accuracy of object predictions as stated in Section 4.1.1. Therefore, we propose to add new features to the DPM pipeline in order to improve visual recognition for the challenge on object detection and orientation estimation [Geiger et al., 2012].

Firstly, we will review the modified HOG features proposed in [Felzenszwalb et al., 2010b]. As depicted in Fig. 4.11, an input image patch of size $M \times N$ is divided into squared HOG cells of 8×8 pixels. A margin of 1 cell is left on each image border, such that the gradients are computed in the remaining cells of the image. Each computed visual feature consists in a histogram of length $d = 32$ elements. As a result, a cube of dimensions $(h_c \times w_c \times d)$ describes the input image patch. In Fig. 4.11, the cube frontal face is overlaid with a pictorial representation of the HOG descriptors. It draws every vector as oriented segments weighted by the corresponding element of the histogram, such that the stronger gradients are the ones showing whiter segments.

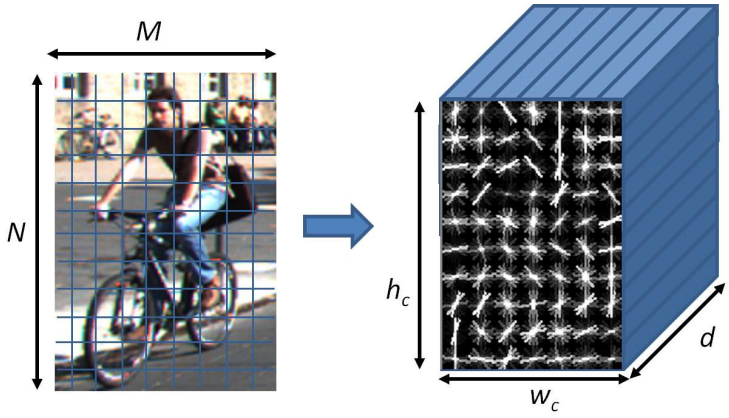


Figure 4.11: Example of the HOG descriptor proposed in [Felzenszwalb et al., 2010b]. From left to right: input image patch of 73x93 pixels, visualization of the 7x10x32 cube with the internal structure of the computed descriptor and the pictorial representation of the gradients on every squared cell. Every cell in the grid corresponds to a block of 8x8 pixels. Vectorizing the descriptor, the total length is 2,240.

In relation to features construction, Fig. 4.12 illustrates the generation process as a concatenation of contrast-sensitive (*B1*) and -insensitive (*B2*) gradients and 4 different normalizations of the histogram. The "contrast-sensitiveness" regards the number of orientations for the discretization of the gradients. *B1* are 18 bins in the range $[0, 2\pi]$, while *B2* are 9 bins reduced to $[0, \pi]$, which is obtained by folding *B1* on two halves and adding up its elements.

Once the reference color-based features have been described, the newly devised 3D-aware features will be presented. Attending to the problem formulation in terms of a scoring function (see Eq. 4.3), we propose to add 3D-aware parameters and potentials that capture the disparity gradients of the objects, providing 3D cues of the scene as already explained in the last paragraph of Section 4.1.1. This is formally stated in Eq. 4.6.

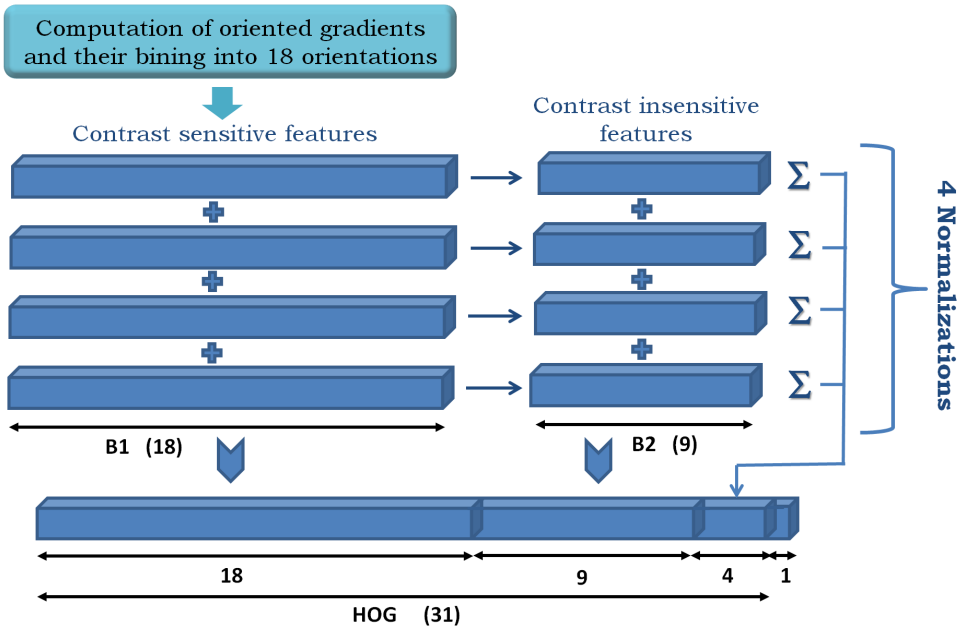


Figure 4.12: This diagram illustrates the generation process of the HOG descriptor proposed in [Felzenszwalb et al., 2010b]. The histograms of oriented gradients are firstly computed on 18 discretized orientations in $[0, 2\pi]$ and they are normalized with 4 rules [Dalal and Triggs, 2005]. Then, these histograms are collapsed to the range $[0, \pi]$ and finally, four accumulators related to each normalization are concatenated together to form a 31-dimensional descriptor, which is truncated by 1 additional element for memory alignment purposes.

$$s(z) = \sum_{i=0}^{n} F_i \cdot \phi_c(\mathbf{x}_c, p_i) + \sum_{i=0}^{n} G_i \cdot \phi_d(\mathbf{x}_d, p_i) - \sum_{i=1}^{n} \mathbf{d}_i \cdot \phi_\alpha(du_i, dv_i) + bias \tag{4.6}$$

In the equation above, G_i are the new disparity filters and ϕ_d are the disparity features for each latent hypothesis computed on the input disparity image $\mathbf{x}_d$.

Our approach can be also viewed as the concatenation of features from color images and disparity maps, which forms the 2.5D measurements for the part-based detector. Hence, during training we learn richer models of the visual and depth appearances of the objects, while for detection, additional measurement data is available from the stereo images. Although the disparity maps are not provided in [KITTI, 2012], we compute them from each pair of left-right images employing the SGM [Hirschmuller, 2008] method, as already indicated in Section 4.1.1.

This Thesis proposes several 3D-aware features that combine color and disparity information. In Chapter 5, we demonstrate from a set of initial experiments that the gradient information from the disparity maps can obtain object prediction ratios close to those ones produced with gradients on color images. Consequently, the semantic information contained in the scenes is preserved in the disparity images, thus, the DPM framework was able to learn discriminative models. However, disparity alone is not able to achieve the detection performance yielded by modeling color appearance. Therefore, the 3D-aware features contributed in this Thesis incorporate 2.5D data into DPM framework. The diagrams in Fig. 4.13 and 4.14 show how they are built and a large set of related experiments are included in Chapter 5 and Appendix B.

- ***C2***. Concatenation of HOG features from color (HOG_c) and disparity (HOG_d) images producing a descriptor of length 64.
- ***C3***. Concatenation of C2 features with an element-wise product (HOG_p) of the contrast-sensitive and contrast-insensitive histograms. Addition of 1 element at the end for truncation and memory alignment. This is the longest feature tested with 92 elements.
- ***C4***. Concatenation of the element-wise product (HOG_p) and the 4 normalization blocks from color images (HOG_{c-N}) plus the truncation element. Total length of 32.
- ***C5***. Concatenation of HOG features from color (HOG_c) and the element-wise product (HOG_p). Total length 60 dimensions.

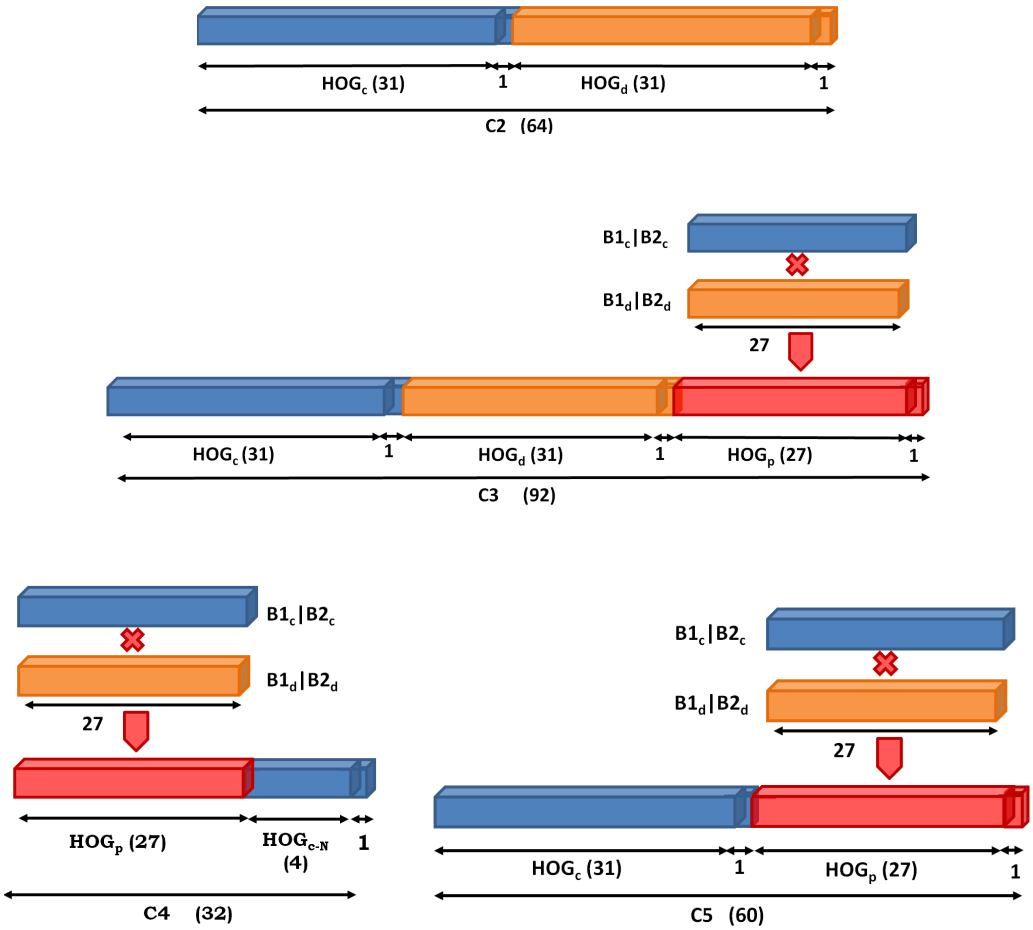


Figure 4.13: 3D-aware features as different combinations of HOG descriptors computed on 2.5D data. HOG_c refers to the HOG descriptor as defined in [Felzenszwalb et al., 2010b] computed on the color image, while HOG_d is calculated on the disparity map. Besides, HOG_p is the element-wise product of the 27 contrast-sensitive (B1) and -insensitive (B2) features. Finally, HOG_{c-N} represents the 4 normalization accumulators for the color image described in Fig. 4.12.

- ***C6***. This is a special case in which the color feature (HOG_c) is followed by 4 statistics computed on the disparity image for every cell defined by HOG_c. This is inspired on the *DispStat* feature of [Walk et al., 2010] employed for pedestrian detection. In particular we concatenate the *max*, *min*, *mean* and *median* over the cell. For the normalization of this

new feature we propose a *whitening* algorithm, which is described in Section 4.4.1. The final length of this vector is 36.

- ***C7***. This feature is conceived as the intersection between HOG_c and HOG_d computed with the element-wise minimum operation, yielding a descriptor of the same length (32).
- ***C8B1***. The last two features focus the analysis on the importance of contrast-sensitive (B1) vs contrast-insensitive (B2) histograms on the disparity. They have been devised after the first experiments and results analysis on previous features. More details can be found in Chapter 5. The first one, *C8B1* accounts for gradients in disparity in the range $[0, 2\pi]$ and the histogram is discretized in 8 bins instead of 9 because of memory alignment purposes during the convolution of an image with the learned filters.
- ***C8B2***. Similarly, this feature is reduced to the range $[0, \pi]$ for contrast-insensitiveness.

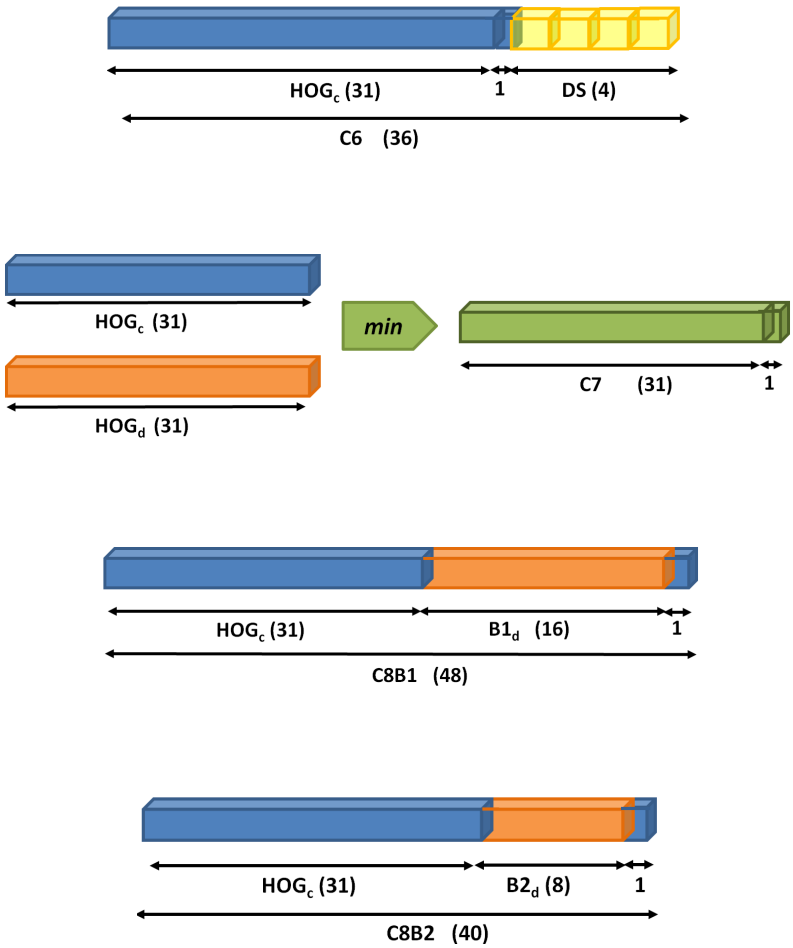


Figure 4.14: 3D-aware features as different combinations of HOG descriptors computed on 2.5D data. In addition to Fig. 4.13, here *DS* refers to 4 disparity statistics on every HOG cell (max, min, mean median). Besides, $B1_d$ and $B2_d$ corresponds to contrast-sensitive and -insensitive features, but discretized into 8 gradient orientations.

The next Fig. 4.15 depicts a set of object instances and the corresponding features computed from color and disparity. The displayed cars, pedestrians and cyclists have been automatically detected on test images employing the learned models from the KITTI training dataset. These

sample images are included here to illustrate the 3D-aware features contributed in this Thesis. They are plotted as small glyphs of the positive weights of the descriptors and represented on a color scale. As it can be observed, the gradients from color images can be visually recognized more easily. However, the gradients on the disparity provide complementary information about the objects depth, which leads to an enhanced description of these object instances in the scene.

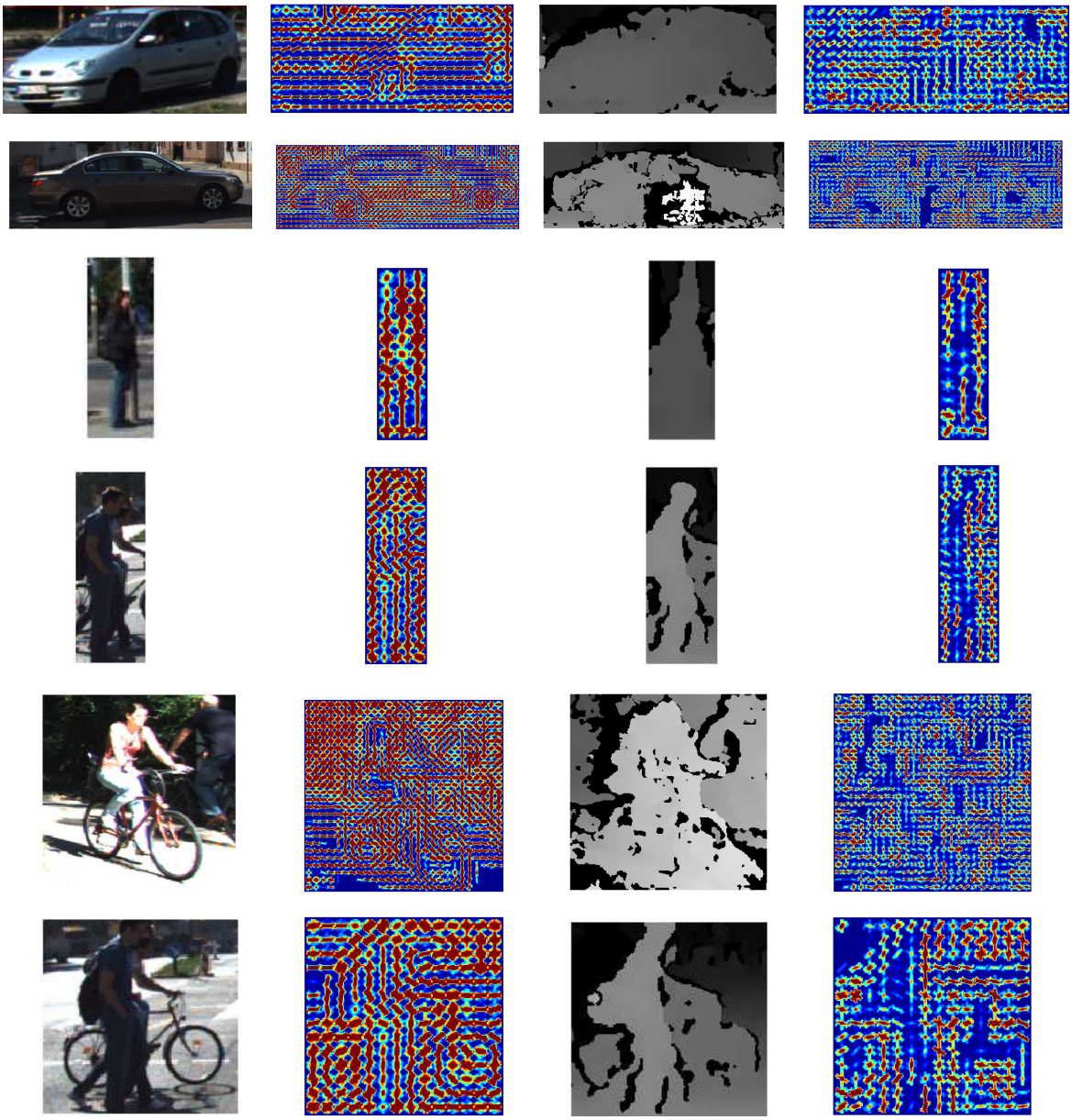


Figure 4.15: Object instances and their 3D-aware features, in particular *C8B1*. The first two columns display the original color image and the gradients related to color, the next two columns show the disparity patch and its gradients.

4.2.1. Scaling disparity

When dealing with disparity features, another important issue, which has been pointed out in [Walk et al., 2010, Helmer and Lowe, 2010], is the scaling of disparity values to accommodate to the objects height variances depending on their depth in the scene. In the first citation, disparity is scaled depending on the ratio of the current object hypothesis and a reference height for the object class. The second work is applied to indoor environments to detect small objects for mobile robotics. It employs a disparity prior that accounts for depth and scale agreement of the bounding box, leading to a reduction in the false positives and increased scores for the correct detections.

In particular, [Walk et al., 2010] exposed that the ratio of disparity (D) and the observed height (h_o) is inversely proportional to the object height (H_o) in the 3D real scene. In different words, this is equivalent to say that the relation between the measured disparity and the object height in pixels are linear-dependent upon the baseline of the stereo rig (B) and the reference height (H_o), which can be considered as constants. Hence, $D = \frac{B}{H_o} \cdot h_o$.

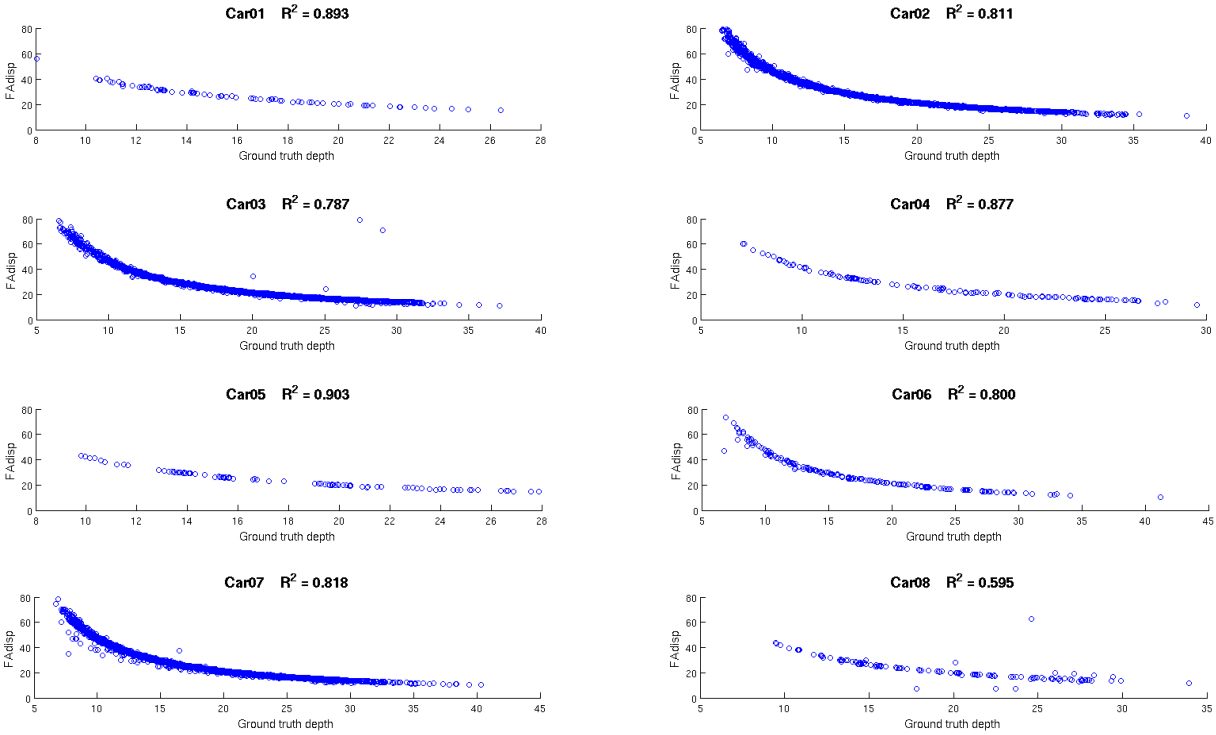


Figure 4.16: Plotting the *FAdisp* measurement vs the ground-truth depth (in meters) for the class 'car' of KITTI training dataset. The class is clustered in 8 different viewpoints, having a different aspect ratio each one. Excluding a very low number of outliers, all the samples fulfill that $D \propto \frac{1}{Z_d}$.

The proportionality between h_o and D was studied for pedestrians in [Walk et al., 2010] employing different features computed on the disparity map, but we add here further tests for the class 'car'. Let us design a basic measurement of the disparity in a 2D bounding box around ground-truth objects. In particular, we employ a filtered and averaged disparity value (*FAdisp*), which is computed from the histogram of the 2D patch. The disparity is quantified in 80 levels

and the first 5 bins are discarded because we empirically observed that they correspond to very low disparities related to errors. Then, the maximum mode of the histogram is found in order to compute a weighted average of the disparity in the peak surroundings (at -12dB of decay). The effectiveness of *FAdisp* is shown in previous Fig. 4.16, where there are several plots for the class 'car' clustered in 8 different viewpoints, related to 8 different aspect ratios, too. As it can be appreciated, all the curves correspond to the function shape of $f = 1/x$, which denotes the inverse proportionality between the disparity D and depth Z_p ($D = \frac{f \cdot B}{Z_p}$). Moreover, the *goodness of fit* (R^2) quantifies how well these plots fit to function f.

Consequently, we can also demonstrate the linear relation between the disparity and the height in pixels for an object in the scene ($D \propto h_o$). Fig. 4.17 shows 8 plots for the clustered cars and the R^2 measurement, which is over 90% for most of the cases. The same conclusions were obtained for the classes 'pedestrian' and 'cyclist' of [KITTI, 2012], but we have omitted them to limit the extension of this chapter.

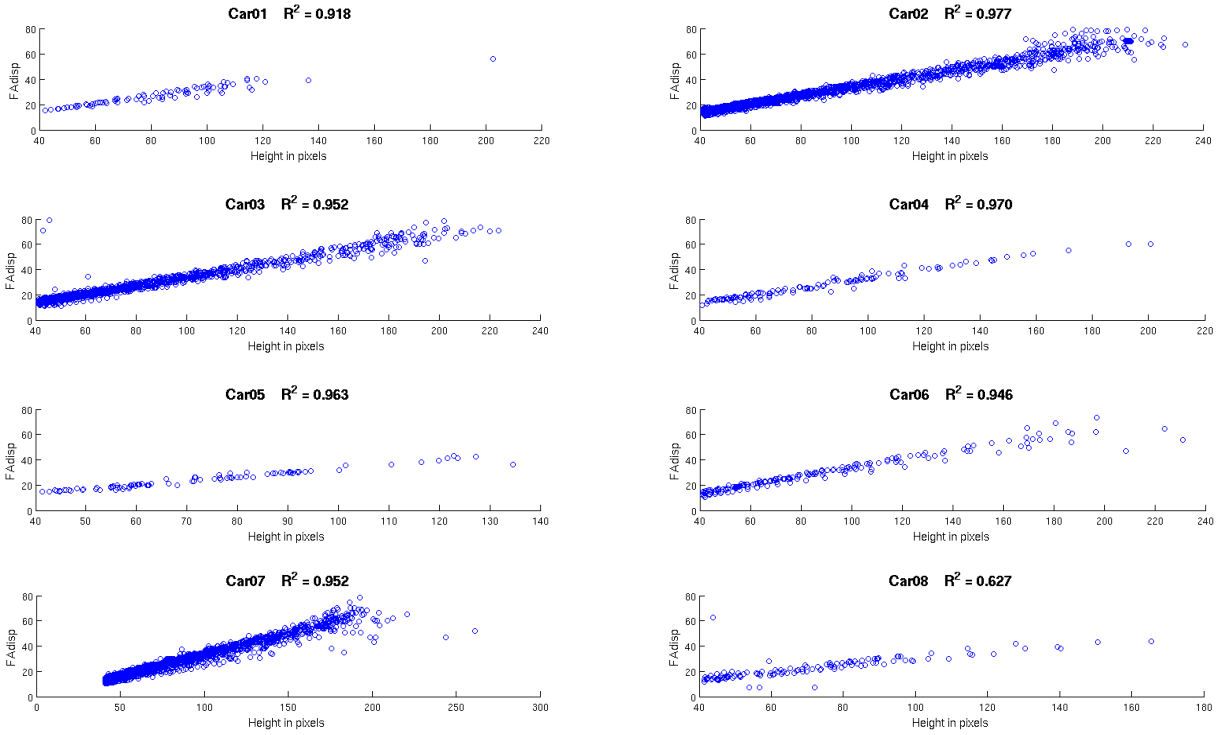


Figure 4.17: Disparity vs cars height in pixels. The plots proof the linear relation between the measured disparity D on the 2D patch and the pixel height h_o of the cars, given a reference height H_o of the objects in meters. This real height (H_o) is an intrinsic parameter carried out in the class because all mid-sized cars present very similar heights, excluding some outliers for small urban vehicles (SUV).

As stated in the beginning of this section, the disparity invariance can be employed to improve the object detection. More specifically, in the DPM framework, we propose to scale the disparity according to the following expression: $D' = \frac{D}{h_o/h_g}$. This is done during training, on the first stage of model initialization (see Algorithm 1). In fact, the filters G_i in Eq. 4.6 are initialized to a fixed size based on the size and aspect ratio of the training samples. Then, the positive samples are warped to this size. As a consequence, the disparity must be updated with a new value D' that accounts for the scale change in pixels from h_o to the new h_g of the filter.

On the other hand, this update is not performed during feature scale pyramid computation or during object detection, because, as opposed to [Walk et al., 2010], DPM does not employ a sliding window approach. DPM carries out a convolution of the filters at each level of the feature pyramid, thus, the height of the searching window is already fixed by the filter height. Indeed, we can view the process of resizing images through the pyramid like looking a landscape from binoculars. Zooming in causes a narrow field of view, while zooming out enlarges it. Hence, different features are computed, but the object captured in the image is always at the same distance from the camera in the real scene, so that disparity does not need to be updated in the feature pyramid.

4.3. Supervised parameter learning in DPM

The discriminative learning method employed in DPM was introduced in Section 4.1.2 and it is deeply described in the seminar paper [Felzenszwalb et al., 2010b]. The method employs several ML techniques to train an object category model in a supervised fashion. This means the ground-truth labels are available in the dataset and can be used during training to learn and optimize the vector of parameters $\boldsymbol{\beta}$. However, we also understand this supervision process as the natural behavior of building further improvements on top of the existing DPM framework. Therefore, this Thesis contributes with new ideas to tune DPM and reviews some already applied approaches to learn more discriminative models, also providing a large set of experiments for comparison on the KITTI dataset [Geiger et al., 2012], as it will be shown in Chapter 5. Then, in addition to the contributed 3D-aware features in previous section, the next paragraphs present a set of subtleties for supervised parameter learning in DPM.

Orientation estimation of objects. [KITTI, 2012] launches two challenges: predict 2D bounding boxes of cars, pedestrians and cyclists and estimate their viewpoint angle with respect to the stereo camera in the vehicle. The first one is naturally solved by DPM as an object detector, but the second one requires that every component of the mixture of CRF models corresponds to one orientation. Originally, DPM performs an unsupervised clustering of the ground-truth bounding boxes in different aspect ratios. However, we initialize these clusters depending on the orientation label of the training samples. In particular, we consider the same discretization made by [KITTI, 2012]: 16 angles for cars, 8 for pedestrians and 4 for cyclists, because there are more available training samples for cars and due to the low intraclass variability when modeling the appearance of pedestrians and cyclists into many distinct viewing angles. Figures 4.19-4.21 illustrate the learned mixture models when using *C8B1* features.

The orientation, viewpoint or observed angle, as defined by [Geiger et al., 2012] considers the camera coordinate system and the vector from the camera center to the object center. For example, this angle is zero only when the object is located along the Z-axis of the camera (which is in the direction of the optical axis). To cluster every labeled object in one model component, i.e. discretizing in the orientations, we employ the angle quantification depicted in Fig. 4.18.

Moreover, bilateral symmetry is not employed in our work because the orientation ground-truth labels are provided in KITTI and every component of the mixture does not need to

differentiate between the left/right versions of the object, as this is already implicit in the viewpoint modeling.

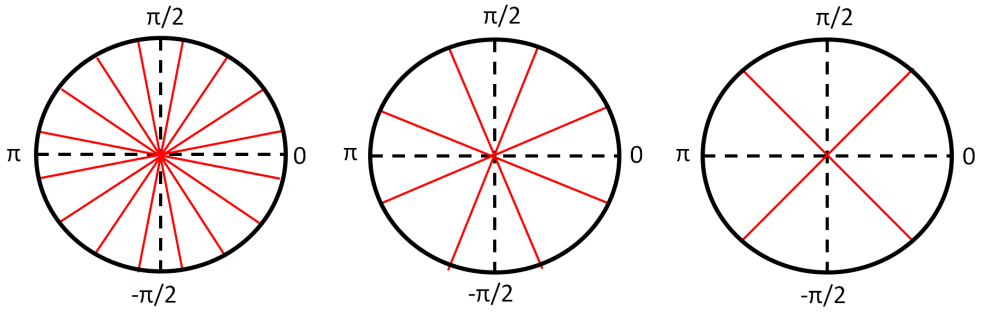


Figure 4.18: Discretization of the objects orientation in the range $[-\pi, \pi]$. Every component of our learned mixture models in DPM framework corresponds to one sector delimited by the red lines. During detection, every predicted viewpoint is labeled as the angle of the bisector. From left to right, each circle is related to car, pedestrian and cyclist categories respectively.

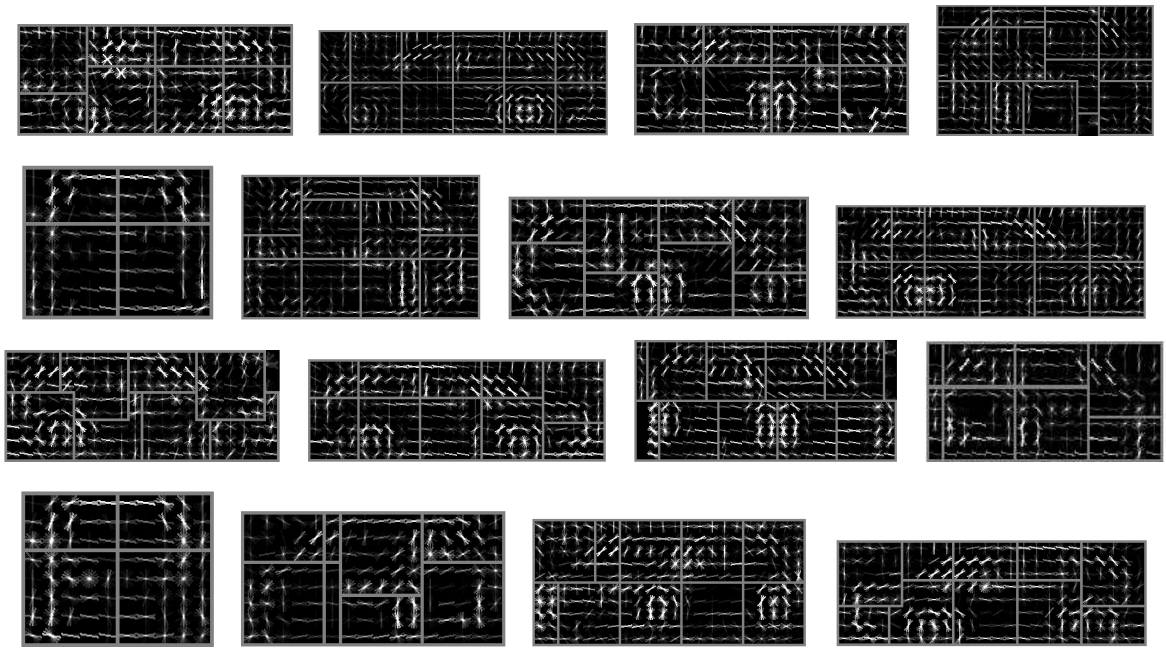


Figure 4.19: Mixture model example with the different viewpoints for the class 'Car'. Particularly, these are the object parts at twice resolution of the root filter. The 3D-aware features *C8B1* were employed to train this model.

Training data selection. Supervised training also regards the selection of the training samples such that, the cleaner data the better model learning. However, it also depends on the complexity grade that the model is designed to represent [Zhu et al., 2012]. DPM is able to model an object category at multiple scales, under small partial occlusions, illumination changes and it is relatively flexible to intraclass variability. Hence, to account for the performance variability, we have carried out a set of experiments (check Chapter 5) increasing the difficulty level of the training samples in terms of truncation, occlusion and minimum pixel height, according to the labels of the ground truth.

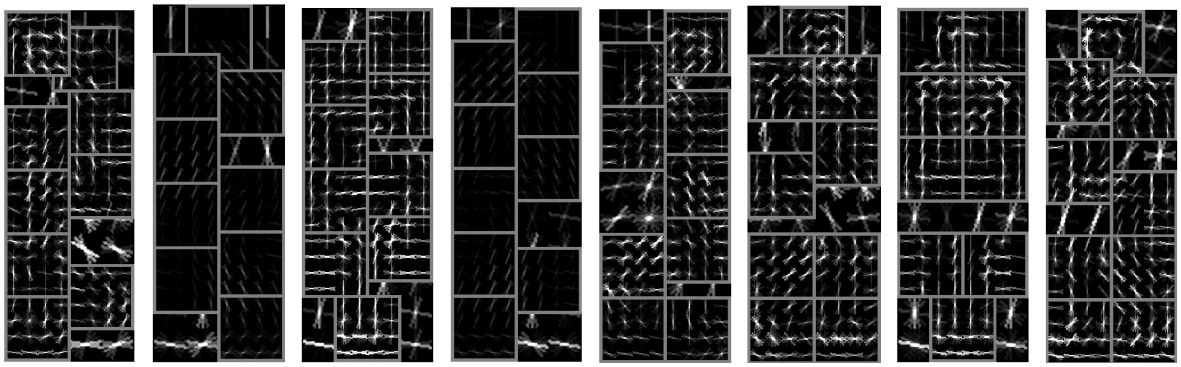


Figure 4.20: Mixture model example with the different viewpoints for the class 'Pedestrian'. Particularly, these are the object parts at twice resolution of the root filter. The 3D-aware features *C8B1* were employed to train this model.

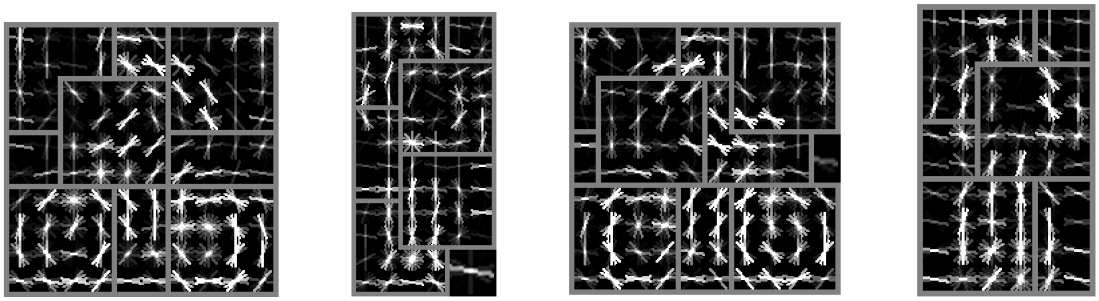


Figure 4.21: Mixture model example with the different viewpoints for the class 'Cyclist'. Particularly, these are the object parts at twice resolution of the root filter. The 3D-aware features *C8B1* were employed to train this model.

Furthermore, we also propose to enlarge the training dataset by mirroring the positive samples (ground-truth bounding boxes) and relabeling them into the corresponding viewing angle with respect to $\pi/2$ and $-\pi/2$, which act as the axes of reflection. This compensates for unbalanced number of samples in the different object orientations and increases the training data to support the mixture model learning. Overfitting is prevented with 5-fold cross-validation, as it will be shown in Section 5.2.

On the other hand, the KITTI dataset contains many difficult samples at far distances, which present a small size in pixels. In this context, we upsample the ground-truth image patches that have a lower size than the model templates. Thus, these positive images also contribute during learning, instead of being discarded as it is originally done in DPM. This modification was also mentioned in the LSVM-MDPM-sv approach published in [KITTI, 2012].

Root filters initialization. To initialize the size and aspect ratio of every model template, DPM picks the 20 percentile area from the distribution of the labeled bounding boxes for each model component. Hence, in our approach, a different root filter size and aspect ratio will be set for every object orientation subcategory. In addition, we modify the minimum and maximum area bounds for these root filters for a better initialization in Algorithm 1.

Selection of negative samples. The number of negative samples for training is immense, i. e. there are loads of subwindows in the dataset images that belong to the background, in "positive images" (those ones containing some annotated object) and "negative images" (without

any annotated object). Consequently, DPM constructs the training data from positive instances and from *hard negative* instances in its data-mining approach, which collects the negatives from strictly "negative images" (see Algorithm 3). However, in KITTI dataset, for the class 'car', this is a problem. In fact, this urban dataset contains cars in almost all images. Then, there are few "negative images" for hard-negatives data-mining. Therefore, inspired on MDPM-LSVM-sv [KITTI, 2012], we make two distinctions: 1) In the first training stage of DPM (Algorithm 1) we pick random negative samples from strictly negative images and, 2) during latent search and bootstrapping (Algorithms 2 and 3), the hard negatives are collected from positive images, for better modeling of the background class.

Overlap requirement. During results evaluation, there exists a minimum overlap requirement to consider a bounding box prediction as a correct matching with the ground truth. However, the DPM training pipeline also imposes a minimum overlap requirement during the latent positives search, such that the latent candidates must overlap at least a 70% with the ground-truth bounding box. This requisite constraints the learning process for a better fit of the models. We provide several experiments while modifying this parameter around the default value.

LSVM regularization. The loss function definition for LSVM training [Felzenszwalb et al., 2010b] presents the regularization parameter C, which is similar in the hinge loss of the *Support Vector Machine (SVM)* classifier. This parameter influences the scale of the learned vector $\boldsymbol{\beta}$, but it also affects the generalization of the model to unseen data and, hence, the final performance during detection. Indeed, there is a dependency on the training dataset size and feature length [Zhu et al., 2012] and typically, several training rounds are carried out to select the most appropriate value for C. Due to the long training times required by DPM, we trust on the default value provided by DPM and do not carry extensive tests in this aspect. Nevertheless, we have observed performance differences while cross-validating our 3D-aware features (of different lengths) and employing two or three different values for C. The experiments on this regard are included in Chapter 5.

Latent viewpoint. The overlap requirement described above does not discriminate between latent candidates belonging to different model components. Then, the viewpoint is considered a latent variable, such that the latent candidate z presenting the maximum score (see Eq. 4.5) will be selected as the leading hypothesis during training. However, inspired by [López-Sastre et al., 2011] we opt for fixing the latent viewpoint with ground-truth information, i.e. we restrict the hypotheses space to the latent positives presenting the same model component (orientation) that is annotated in the ground truth. We enforce this only on the second stage of the training pipeline (Algorithm 2), when the model components are merged. Thus, we guide the learning process, but also keep the viewpoint as a hidden variable during the last stage of latent parts training.

Adaptive object parts. The part-based approach in DPM proposes a rigid object model with certain degree of flexibility to adapt to intraclass variation. However, the number of parts and their shape is left unchanged for all the mixture components. Differently, based on prior information from the dataset, we enrich the model parts initialization in Algorithm 3 with variable aspect ratios and sizes for the object parts on every viewpoint. This provides further model adaptation to the intraclass variability attending to the viewing angle subcategories.

4.4. Additional approaches

Previous sections have exposed the contributed 3D-aware features and several aspects to be considered for the supervised learning of the mixtures of CRFs in DPM framework. The aim of this section is to introduce two new ideas with the aim of obtaining better object detection. The first one is *whitening*, which is employed in the parameter learning process to normalize the gradient features before the stochastic gradient descent that optimizes $\boldsymbol{\beta}$. The second one is a *stereo consistency check* that enforces epipolar geometry constraints during object detection.

4.4.1. Whitening

The HOG features employed in DPM are normalized, so that every dimension of the descriptor have a comparable scale. However, in this Thesis we have proposed 3D-aware features that are feed to the DPM pipeline and do not have any additional normalization. In fact, these features are histograms from color and disparity data that are individually normalized with respect to the image blocks in the vicinity as defined in [Felzenszwalb et al., 2010b]. More specifically, the lack of a joint normalization after the histograms concatenation (as presented in Section 4.2) can cause slower convergence during learning and weaker learned parameters. This comes from the fact that the stochastic gradient descent algorithm, which optimizes the vector of parameters $\boldsymbol{\beta}$ during learning, is sensitive to anisotropically distributed samples. Hence, *whitening* the feature space enforces an isotropic distribution which can help to better learn the linear LSVM classifier. Whitening is equivalent to feature equalization and it can remove the implicit correlations in the features that do not correspond to discriminative information. Therefore, decorrelating the elements inside the descriptors will lead the training, such that a stronger focus will be on learning interesting regularities in the dataset for every object class.

The need of this normalization process is more obvious for some of our proposed features, i.e. the ones based on a product of descriptors (*C3, C4 and C5*) and the one adding the 4 statistics of the disparity cells (*C6*). Next, we will describe in detail the proposed whitening process.

According to [Gharbi et al., 2012], the *whitening* can be defined as a standard technique in signal processing and machine learning, in which the normalized samples are obtained by transforming the space by the inverse square root of the covariance matrix, ($\Sigma^{-1/2}$). Similarly, [Hariharan et al., 2012] devised the *Whitened Histograms of Orientations (WHO)* features, which are transformed HOG vectors with isotropic covariance matrix. Then, given these reference works, we formulate the required normalization process in Eq. 4.7.

$$\hat{\mathbf{x}} = \Sigma^{-1/2}(\mathbf{x} - \boldsymbol{\mu}) \tag{4.7}$$

where $\mathbf{x}$ represents the feature vector and $\hat{\mathbf{x}}$, its transformed version, being $\boldsymbol{\mu}$ the empirical mean computed over all feature samples in the dataset and Σ the corresponding covariance matrix. It must be noted that whitening is related to *Principal Component Analysis (PCA)* and *Singular Value Decomposition (SVD)* [Bishop, 2006], but we will not perform dimensionality reduction on the descriptors length.

Three problems arise from Eq. 4.7 in order to compute the unknown whitening matrix and mean vector.

1. In DPM, every model template (F_i and G_i) associated to each object viewpoint has a different size. Thus, every feature vector $\mathbf{x}$ also presents a different length. Consequently, a different mean vector and covariance matrix are required for each model component and object category.

2. The dimensionality of $\mathbf{x}$ is high, typically from 1,000 - 3,500 from the experiments carried out in this Thesis. This involves that the covariance matrix computation will require a large number of samples of the order 10^6. However, this amount of training samples is not available for every model component. Despite the big size of KITTI dataset, the scale pyramid on each image and our mirroring of positive samples, the number of items are around $2 \cdot 10^5$ during initialization, but much lower and also unbalanced during latent parts training. Therefore, the covariance matrices can not be reliably estimated before every learning step.

3. For the computation of the whitening matrix, the inverse square root of Eq. 4.7 can not be directly computed. Hence, the covariance matrix has to be decomposed in order to estimate the whitening parameters $W = \Sigma^{-1/2}$.

To deal with the last point, we propose to employ the *Zero Components Analysis (ZCA)* [Krizhevsky, 2009], which decomposes the covariance matrix in its eigenvalues and eigenvectors. For the remaining issues, we generalize a single covariance matrix and mean vector for all the model components and object categories, which is inspired in the discriminative decorrelation of [Hariharan et al., 2012].

Next, the whole process for obtaining W is described in detail. Firstly, let us consider a 3D-aware feature of size $h_f \times w_f \times d$, such that h_f and w_f define the height and width of an image patch in HOG cells [Dalal and Triggs, 2005] and d is the number of color and disparity features per cell, which is fixed depending on the feature type (see Section 4.2). Hence, every data sample $\mathbf{x}$ can be described as the concatenation of the features on all the cells, resulting in a vector of length $N_f d$, being N_f the number of cells ($h_f \cdot w_f$). Then, attending to the approach in [Hariharan et al., 2012], we can compute a reference mean vector $\boldsymbol{\mu}_0$ of length d and a generic spatial autocorrelation matrix Γ of size $N_l d \times N_l d$ to represent the feature space. N_l is the largest number of HOG cells, which correspond to the largest template size of a model component in DPM.

The mean is obtained with Eq. 4.8 by averaging over all data samples (M_{ds}) and cells (N_f).

$$\boldsymbol{\mu}_0 = \frac{1}{N_f \cdot M_{ds}} \sum_{n,m} \mathbf{x}_\mathbf{n}^\mathbf{m} \tag{4.8}$$

Besides, the covariance in Eq. 4.9 is computed per block $b(j,k)$ of size $d \times d$, where j and k are index intervals of the matrix.

$$\Sigma_{b(j,k)} = \Gamma_{u,v} = E[\bar{\mathbf{x}}_\mathbf{u}, \bar{\mathbf{x}}_\mathbf{v}^T] = \frac{1}{M_{ds}} \sum_m (\mathbf{x}_\mathbf{u}^\mathbf{m} - \boldsymbol{\mu}_0)(\mathbf{x}_\mathbf{v}^\mathbf{m} - \boldsymbol{\mu}_0)^T \tag{4.9}$$

In the equation above, the spatial autocorrelation function Γ is evaluated for each pair of vector segments $\bar{\mathbf{x}}_\mathbf{u}$ and $\bar{\mathbf{x}}_\mathbf{v}$ with zero mean, for the HOG cells u and $v \in 1, ..., N_f$ on each data sample m.

The data samples $\mathbf{x}$ are collected from the training images without ground-truth labels. In the particular case of the KITTI dataset employed in this Thesis, we randomly pick 12 subwindows of fixed size (128x72 pixels) at 8 different resolutions of the training images (7,480). This is to achieve invariance to translation and scale. Finally, we obtain a dataset of 718,080 features ($\mathbf{x}$) that are feed to Algorithm 4, which performs an incremental computation [Knuth, 1998] of equations 4.8 and 4.9 with the aim of saving memory.

Algorithm 4 Incremental computation of the mean and spatial autocorrelation.

Input: X matrix of data samples ordered in columns.
Input: N_l number of cells of a data sample $\mathbf{x}$.
Output: $\boldsymbol{\mu}_0$ mean vector of length d.
Output: Γ spatial autocorrelation matrix of size $N_l d \times N_l d$.
Algorithm:

```
n = 0; m = 0; zeros(μ_0); zeros(Γ);
for x in X do
    m = m + 1;
    for v = 1 → N_l do
        n = n + 1;
        ρ = x_v − μ_0;
        for u = v → N_l do
            Γ_{u,v} = Γ_{u,v} + ρ ∗ (x_u − μ_0)^T;
        end for
        μ_0 = μ_0 + ρ/n;
    end for
end for
Γ = Γ/(m − 1);
```

Once the parameters $\boldsymbol{\mu}_0$ and Γ are obtained, they are employed during the initialization stage of DPM training to build the covariance matrices Σ_c and mean vectors $\boldsymbol{\mu_c}$ of every model component c. In particular, as suggested by [Hariharan et al., 2012], for a given model template size with N_c cells, $\boldsymbol{\mu_c}$ is built as a concatenation of the reference vector $\boldsymbol{\mu}_0$, N_c times. Similarly, Σ_c is the submatrix of size $N_c d \times N_c d$ extracted from the blocks of Γ. After this particularization for every model template, each whitening matrix W_c is computed with the ZCA approach [Krizhevsky, 2009] plus a regularization of the eigenvalues. In mathematical notation, the symmetric and orthogonally diagonalizable matrix Σ_c can be decomposed in its diagonal matrix of eigenvalues (Λ) and the unitary matrix of eigenvectors (U): $\Sigma_c = U\Lambda U^T$. As a result, calculating the inverse square root of the covariance can be done as denoted in Eq. 4.10.

$$W_c = \Sigma_c^{-1/2} = U\Lambda^{-1/2}U^T \tag{4.10}$$

In fact, Λ is a diagonal matrix with all its elements real valued by definition. Hence, calculating their inverse square root is straightforward. However, in order to remove the lowest eigenvalues corresponding to noise, we carry out a regularization such that we keep those ones

presenting the 99% of the total variance and we set the remaining ones to ϵ. More specifically, we obtain k that holds Eq. 4.11 and set $\epsilon = \lambda_k$.

$$\sum_{i=1}^{k} \lambda_i \geq 0.99 \cdot \sum_{t=1}^{N_c d} \lambda_t \tag{4.11}$$

4.4.2. Stereo consistency check

With the aim of reducing the number of false positives during object detection, we introduce in this section a *stereo consistency check* that employs 2.5D data. In addition to the 3D-aware features that were described in Section 4.2, we propose a married matching between the detections in both stereo views, which basically consists in applying the learned DPM models on the left and right images and match the bounding boxes based on epipolar geometry constraints.

Some works in the literature have researched similar ideas, but they have not been deployed as presented in this Thesis. In particular, [Bao et al., 2012] introduced the "object co-detection" approach to match one object instance in multiple views by measuring appearance and geometry consistency, but without employing disparity information. Their results on stereo images yielded increased average precision compared to DPM. Another related approach, presented the concept of 3D scene-consistency for the stereo matching [Bleyer et al., 2012], applied to the pixel-wise labeling task in [PASCAL VOC, 2012]. In our case, the 3D scene consistency is enforced with epipolar geometry constraints, assuming that KITTI images are already rectified and the calibration parameters also provided in the dataset.

Then, given the set of 2D bounding box hypotheses before the *Non-Maximum Suppression (NMS)* during DPM object prediction, we approach the stereo consistency check as a search of candidates over the space defined by $S(x_l, y_l, h_l, c_l, x_r, y_r, h_r, c_r)$. These variables represent the (x, y) image coordinates of the center of the predicted bounding boxes, their heights h in pixels and their predicted orientation or model component c, in both left and right views. To reduce this large search space (8 degrees of freedom), a set of physical constraints are proposed on the left and right stereo-rectified images. The first two constraints are due to epipolar geometry and the third one is based upon matching the bounding boxes size.

1. A point in the left image has its matching point on the corresponding epipolar line in the right image: $y = y_l = y_r$. In fact, we allow some flexibility searching on $y_r \pm 1$.

2. The bounding box candidate on the right image that matches a prediction on the left image should present a horizontal translation in pixels, which is equivalent to the disparity. Thus, we measure the mean disparity value d_l on a 11x11 block around the center of the 2D predictions in the left image. The size of the block is the same one employed for the generation of the disparity maps with the SGM method [Hirschmuller, 2008]. Therefore, $x_r = x_l - d \pm 1$.

3. Lastly, due to multiple overlapping detections centered at the same object in the scene, which is a normal behavior of the DPM detector, we enforce $h = h_l = h_r$. However, we

have empirically observed the same detection ratios when employing the aspect ratio or the width of the bounding boxes for constraining.

On the other hand, we allow some flexibility by not matching detections with equal orientations, i.e. in general it is allowed that $c_l \neq c_r$, because this can be related to the small viewpoint change when observing the object from the left or right camera. Indeed, this restriction would not reduce the false positives, but the orientation prediction power. Therefore, in Eq. 4.12 we define the re-scoring function for the left-image detections that have matching candidates in the right image. After that, the candidates are sorted in descending order and filtered with the NMS method. It must be noted that new detections from the right image are not added. Besides, no-matched detections in the left image are not downweighted or penalized either. Indeed, we tested these approaches and the detection ratios always plummeted.

$$f(x_l, y, d_l, h, c_l, c_r) = (score(x_l, y, h, c_l) + score(x_l - d_l, y, h, c_r))/2 \tag{4.12}$$

The function *score* that appears above refers to the output scalar obtained from Eq. 4.6, which is related to every predicted bounding box. Obtaining the scores for the detections on both stereo images, involves computing the two disparity maps in order to detect objects in left and right views when employing our 3D-aware features. Indeed, we obtain the left-disparity image (D_l) (referred to the left-image coordinate system) from SGM algorithm, which internally includes a cross-check between left and right images for increasing the accuracy of the disparity map. Then, we estimate the right-disparity map (D_r) (referred to the right-image coordinate system) from the equivalence in Eq. 4.13. Moreover, we apply a posterior dilation process to compensate for discontinuities.

$$D_l(i,j) = D_r(i, j-d) \quad | \quad d = D_l(i,j) \tag{4.13}$$

$$D_r(i,j) = D_l(i, j+d) \quad | \quad d = D_r(i,j) \tag{4.14}$$

4.5. Conclusions

This Chapter has presented the methodology for the joint object detection and orientation estimation in road scenes. In particular, for the object categories 'car', 'pedestrian' and 'cyclist', contributing to the visual recognition challenge known as KITTI [Geiger et al., 2012], which a state-of-the-art and public dataset. The successful object detector known as Discriminative Part-based Models (DPM) [Felzenszwalb et al., 2010b] has been revisited, because it is the baseline framework for the KITTI challenge. Moreover, a set of modifications have been proposed on top of DPM to incorporate 2.5D data (color and disparity). In fact, our research work is the first proposal using stereo data that have been published in the KITTI website [KITTI, 2012].

Firstly, the object detection has been presented as a 3D challenge using the point clouds reconstructed from stereo. However, we have depicted the inherent difficulties to perform 3D reasoning and modeling from the sparse and noisy point clouds in naturalistic urban environments. Then, several 3D-aware features have been proposed, which measure color and disparity

gradients on the KITTI road scene images. In Chapter 5, 5-fold cross-validation experiments evaluate and compare them. Besides, subsection 4.2.1 has shown the relationship between the object height in pixels and an estimated object disparity, proposing an image patch resize of the ground-truth samples during the first stage of DPM training procedure.

Furthermore, section 4.3 has reviewed and proposed several modifications for supervised learning of the mixture models in DPM. Their influence in the object detection and orientation estimation performances are reported in Chapter 5 and additional validation rounds are attached in Appendix B. Apart from the modifications to the DPM training pipeline, two additional approaches have been devised with the aim of improving detection ratios. The 'feature whitening', for normalization/equalization of the 3D-aware features in order to benefit the stochastic gradient descent algorithm, which is part of the linear LSVM inside DPM. The second one is the 'stereo consistency check', which matches detections on both stereo views to reduce the number of false positives. Experimental results and conclusions on these proposals are included in the next Chapter.

Chapter 5

Experiments for KITTI object evaluation challenge

This section describes the evaluation protocol and principal experiments for the object detection and orientation estimation challenge [KITTI, 2012] as explained in Chapter 4. In first place, a larger view on the KITTI dataset [Geiger et al., 2012] and the evaluation protocol for results assessment are reviewed in detail. Then, Section 5.3 presents the experiments on supervised learning and inference with DPM. This section includes the validation tests for DPM tuning using color-based features, for the contributed 3D-aware features and for the whitening and stereo-consistency check. Besides, we also provide the object prediction performances on the KITTI test images with ranking positions compared to other state-of-the-art works and published in [KITTI, 2012]. Finally, a brief discussion on the results closes the chapter.

5.1. KITTI dataset

The KITTI Vision Benchmark has been possible due to the collaboration work between the *Karlsruhe Institute of Technology* and the *Toyota Technological Institute at Chicago*. All the sensor data was captured by the autonomous driving platform *Annieway*, driving around the city of Karlsruhe (Germany) and some nearby rural areas and highways. Several annotators and software tools were employed for labeling multiple objects in the image sequences. Furthermore, a development kit with utility functions in Matlab/C++ is also provided to manage the dataset, which was already introduced in Section 4.1.1, where some images from the training and testing subsets can be found in Fig. 4.1.

Particularly, the object dataset consists of 7,481 training images and 7,518 test images, which are doubled if we consider both views of the stereo camera. In fact, the images are already rectified and the corresponding calibration matrices with intrinsic and extrinsic parameters are also available. The images are panoramic with a resolution around 1240×375 pixels depending on the rectification process. The dataset comprises a total of 80,526 labeled objects (cars, vans, trucks, pedestrians, cyclists, etc.). Besides, several image patches are marked as 'DontCare' regions, which correspond to far objects that are not counted when evaluating the results, such

that detectors performance is not penalized, as we will explain later. In addition to the 2D object bounding boxes and category labels, the ground truth provides: the occlusion level; the truncation percentage on the image borders; the observation angle with respect to the camera (from a bird's eye view) and the 3D cuboid in meters. Although 8 different classes were annotated, only 3 have enough instances for a comprehensive evaluation: cars, pedestrians and cyclists, which are also the categories studied in this Thesis.

Moreover, the discriminative and supervised learning approach exposed in Chapter 4 requires the discretization on object classes and mixture components, i.e. the subclasses associated to each observation angle. Therefore, every object category will be discretized in 16, 8 and 4 orientations, respectively. The reasons are: 1) for a direct comparison with the LSVM-MDPM baseline results in [KITTI, 2012], which employs the same numbers and, 2) due to the dependency on the number of training samples for each class. Particularly, the greater number of samples for the class 'car' makes more easy to learn discriminative models for a higher number of orientations. In total, there are 28,742 cars, 4,487 pedestrians and 1,627 cyclists in the training subset. Obviously, the ground truth is not provided for the test samples and we do not have access to statistics of the testing subset.

Tables 5.1-5.5 present some statistics based on the ground truth of training objects. Indeed, the information summarized on every table corresponds to the dataset characteristics that are relevant (occlusion, object height, orientation, etc.) during the supervised learning and, which also define the difficulty evaluation levels in KITTI.

Table 5.1: Number of samples per occlusion type and object category

Category	Fully visible	Partly occluded	Largely occluded	Unknown
Car	13457	8184	6173	928
Pedestrian	2667	1095	671	54
Cyclist	1010	255	80	282

Table 5.2: Number of samples per truncation percentage and object category

Category	[0-10]%	[10-20]%	[20-60]%	[60-100]%
Car	25044	376	1236	2086
Pedestrian	4205	73	137	72
Cyclist	1511	14	47	55

Table 5.3: Number of samples per pixel height and object category

Category	$<$ 25pix	[25-40]pix	$>$ 40 pix
Car	4108	7552	17082
Pedestrian	82	339	4066
Cyclist	97	334	1196

From the tables above, it can be observed that over the 75% of the samples for all the classes are fully visible or partly occluded, around the 90% present a truncation below 10% with the image borders and there are a few samples of low height (<25pix), i.e. 14%, 2%

Table 5.4: Number of samples per area in pixels and object category

Category	$< 10^3$	$[1-2]10^3$	$[2-4]10^3$	$[4-6]10^3$	$> 6 \cdot 10^3$
Car	4557	5003	5358	2759	11065
Pedestrian	950	843	779	431	1484
Cyclist	437	301	349	121	419

and 6% respectively for every object category. The latter numbers reflect also the aspect ratio differences: cars are wider, whilst pedestrian and cyclists are taller. In fact, the number of them with an area below 1,000 pixels is greater than the values in the first column of Table 5.3. In particular, the percentages are 16%, 21% and 26%, respectively.

Table 5.5: Number of samples per viewpoint and object category

Ref. labels	Viewpoint (rad)	Car	Pedestrian	Cyclist
left side	$-\pi = \pi$	550	435	162
	$-7\pi/8$	1155	-	-
	$-3\pi/4$	1154	475	-
	$-5\pi/8$	2365	-	-
back/rear	$-\pi/2$	7012	955	679
	$-3\pi/8$	2046	-	-
	$-\pi/4$	1232	222	-
	$-\pi/8$	555	-	-
right side	0	404	392	147
	$\pi/8$	718	-	-
	$\pi/4$	354	705	-
	$3\pi/8$	331	-	-
frontal	$\pi/2$	3416	775	639
	$5\pi/8$	5483	-	-
	$3\pi/4$	1386	528	-
	$7\pi/8$	581	-	-

In relation to the observation angle, the number of training samples is unbalanced along the subclasses. As can be seen in Table 5.5, the number of cars is greater around $-\pi/2$ (back view) and $5\pi/8$ (near front view). Besides, the interval $[-7\pi/8, -\pi/4]$ contains more vehicles than its positive counterpart. This can be explained by the cars ahead of the ego-vehicle on urban and interurban roads and the parked cars on cities, which all of them are seen from the back and nearby angles due to road turns, lanes and perspective image projection. Detecting all these road participants helps for the challenges posed by autonomous navigation in urban environments and can also assist during the intersection layout inference shown in Chapter 3.

Similarly, most of the pedestrians and cyclists are observed from their frontal and back views. However, there are also several pedestrians viewed from the side related to street crossings. It must be noted that detecting frontal and back pedestrians on the sidewalks is very important for autonomous vehicles because the detections can be employed to generate trajectories that may predict pedestrian intention. Actually, prevention is a key factor. For example, in general, German drivers are very respectful towards the pedestrians, stopping or decelerating their cars when people is approaching a pedestrian crossing due to an implicit intention predic-

tion. However, this is totally opposite in Italy, where you need to be brave enough and very fast when crossing streets. These "empirical facts" are actually reflected on European statistics [UNECE, 2010, European Commission, 2014]. With regard to cyclists, in Germany roads, they are usually on bike lanes next to the roads, or in their absence, riding on the pavement. Hence, most of the cases are cyclists in the same or opposite directions of the ego-vehicle.

5.2. Evaluation protocol

The evaluation criteria, both metrics and computation algorithm, is of great importance when comparing different detectors, or even more, when comparing several rounds of supervised training experiments [Yebes et al., 2014]. Typically, the evaluation metrics for classifiers rely on the counting of *True Positives (TP)* (correctly detected samples), *False Positives (FP)* (detections not matching the ground truth), *False Negatives (FN)* (ground-truth annotations that could not be predicted) and *True Negatives (TN)* (correct predictions of the background class). The most widely employed evaluation metrics are the ones enumerated below. From them and for the object detection and orientation estimation tasks, we employ *precision, recall, miss rate* and *FPPI* metrics.

- Hit rate ≡ Sensitivity ≡ TP rate ≡ **Recall** ≡ $\frac{TP}{TP+FN}$
- **Precision** = $\frac{TP}{TP+FP}$
- Specificity ≡ TN rate ≡ $\frac{TN}{TN+FP}$
- FP rate ≡ 'type I error' ≡ 1 - Specificity ≡ $\frac{FP}{FP+TN}$
- **Miss rate** ≡ FN rate ≡ 'type II error' ≡ 1 - Hit rate ≡ $\frac{FN}{TP+FN}$
- ***False Positives Per Image (FPPI)*** ≡ $\frac{FP}{Number-of-images}$

Cross-validation. This method divides the training set in K folds, such that $K-1$ subsets are employed for learning a new model and the other for evaluating the prediction performance. This is repeated K times while changing the validation subset. Typically, 10-fold cross-validation is the most recommended option [Duda et al., 2001] because it tends to provide less biased estimation of the system accuracy while using the 90% of training samples. However, the mixture models of DPM framework explained in Chapter 4 require a large training time for each fold, which usually takes between 15 to 30 hours[1] depending on the features length and the number of samples. Therefore, in this Thesis, every trained model relies on **5-fold cross-validation**, which prevents overfitting and assesses the performance of the 3D-aware features and DPM framework of Chapter 4. Then, all the plots attached to this section show the averaged curves from the cross-validation rounds.

[1]This temporal window has been measured when running the learning on a computer with i7 CPU and 16GB of RAM

Evaluation metrics. [Geiger et al., 2012] employs the *Average Precision (AP)* and proposes the *Average Orientation Similarity (AOS)* as common evaluation metrics for the challenge [KITTI, 2012]. They are based upon PASCAL metrics [Everingham et al., 2010]. Particularly, the predicted bounding boxes are sorted in decreasing order of confidence (scores given by Eq. 4.6 in our case) and precision and recall values are computed from the cumulative distribution of TP, FP and FN. Then, AP and AOS are obtained as the *Area under the Curve (AuC)*. The formulas are reproduced here for clarity:

$$AP = \frac{1}{Npr} \sum_{r \in \{0, 0.1, \ldots, 1\}} \underset{\tilde{r}:\tilde{r} \geq r}{max}\, p(\tilde{r}) \tag{5.1}$$

$$AOS = \frac{1}{Npr} \sum_{r \in \{0, 0.1, \ldots, 1\}} \underset{\tilde{r}:\tilde{r} \geq r}{max}\, s(\tilde{r}) \tag{5.2}$$

$$s(r) = \frac{1}{|D(r)|} \sum_{i \in D(r)} \frac{1 + \cos \Delta_{\alpha}^{(i)}}{2}\, \delta_i \tag{5.3}$$

Npr is the number of sampled recall points, which is 41 in KITTI evaluation and, r and p are the recall and precision values respectively. $D(r)$ corresponds to the set of all object detections at recall r, $\Delta_{\alpha}^{(i)}$ is the angle difference between the predicted and ground-truth orientations for the *ith* detection. In addition, multiple detections are penalized, such that $\delta_i = 0$ when the detection i has not been assigned to a ground-truth bounding box, but $\delta_i = 1$ when there exists the minimum required overlap for the object class.

Accordingly, the experimental results in this Thesis are reported using the precision-recall (p-r) curves and related AP and AOS figures, which assign a single value to each curve. Additionally, attending to the metrics shown in a recent survey on pedestrian detection [Dollár et al., 2012], it is also provided the *Log-Average Miss Rate (LAMR)* from log-log plots of miss rate vs FPPI to evaluate the detection performance. It must be noted that AP and LAMR measure different things. In the first case, the precision is related to the number of FP, such that the lower FP, the higher AP, the better classification approach. On the contrary, miss rate comes from the number of FN (missed detections), such that the lower FN, the lower miss rate and the better classification approach. Hence, in the plots provided later, we seek higher values of AP and lower values of LAMR. The latter one is obtained from the following equation:

$$LAMR = \exp\left(\frac{1}{Nfppi} \sum_{f \in \{10^{-2}, \ldots, 1\}} \log(mr_{interp}(f))\right) \tag{5.4}$$

where $Nfppi$ is the number of FPPI points considered (9 as in [Dollár et al., 2012]) and $mr_{interp}(f)$ is the miss rate interpolated at FPPI value f.

Evaluation algorithm. Despite the common metrics above, counting TP, FP and FN differs from PASCAL [Everingham et al., 2010] to KITTI [Geiger et al., 2012]. In fact, we observed that given a set of different experiments and the corresponding sets of predicted bounding boxes, the gradients in AP between the experiments yielded opposite signs and the AP values differed up to 20 points in KITTI vs PASCAL evaluations. Therefore, there is a high risk of

extracting misleading conclusions from the experiments depending on the evaluation algorithm. Although [Geiger et al., 2012] states that their evaluation relies on the well known measurements of [Everingham et al., 2010], we bring here a detailed analysis of KITTI vs PASCAL evaluation approaches because there is no reference in the literature concerning this issue. Next, the common aspects are presented:

- IoU measures the overlapping area between predicted (BB_{dt}) and ground-truth (BB_{gt}) bounding boxes:

$$IoU = \frac{area(BB_{dt} \bigcap BB_{gt})}{area(BB_{dt} \bigcup BB_{gt})}$$

- Every TP is the highest scoring detection with the highest overlap. The remaining overlapped (multiple) detections are counted as FP.
- AP is obtained as the AuC from the "p-r curve".

Most of the works and datasets on object recognition [Everingham et al., 2010, Dollár et al., 2012] impose a minimum overlap requirement of 50% for the IoU area. In particular,[KITTI, 2012] imposes 70% for cars and 50% for pedestrians and cyclists. For instance, Table 5.6 compares AP and AOS for the same experiment evaluated with two distinct overlaps on the 5th fold of a randomly balanced split of training cars. Furthermore, this split is evaluated in three different subsets of ground-truth samples, i.e. "easy", "moderate" and "hard" as defined in [KITTI, 2012]. One of the experiments employs the pre-trained *LSVM-MDPM-sv* for cars [KITTI, 2012] and the other has been trained using the remaining 4 folds of the cross-validation on a selection of 'easy' samples.

Table 5.6: Evaluating minimum overlap requirement for cars

		70%		50%	
		AP %	AOS %	AP %	AOS %
LSVM-MDPM-sv [KITTI, 2012]	easy	72.02	64.95	98.07	88.45
	mod.	55.95	51.01	78.87	70.70
	hard	40.89	37.47	63.54	56.77
Training on 'easy' samples (ours)	easy	83.56	81.88	98.16	96.06
	mod.	47.79	45.52	66.08	63.80
	hard	35.91	34.89	51.91	49.95

As it can be seen, all cases yielded a boost in precision when reducing the minimum required overlap, which comes from a reduction of FN to a couple of miss-labeled ground-truth 'easy' samples (upper bound of $AP \simeq 98\%$) and also due to a notable decrease in FP for 'moderate' and 'hard' levels (FN is still significant in these categories due to smaller and/or occluded samples). Thus, supervising the evaluation protocols and establishing commonalities greatly influences the possible bias in the conclusions obtained from the results.

Additionally, KITTI follows these premises[2], which are not mentioned in PASCAL challenges.

[2]Some premises of the KITTI evaluation protocol are in a text file inside its development kit [KITTI, 2012], but others are directly in the source code.

- 'DontCare' regions do not count as TP or FP when detected for any object category or as FN when missed. Besides, their overlap is treated differently, dividing by the area of the predicted bounding box instead of the union. This favors partial overlapped predictions around these ground-truth regions of a relatively small size.
- Neighbouring classes (e.g. 'Van' for class 'Car', 'Cyclist' for class 'Pedestrian') do not count as TP, FP or FN.
- The detection performance is assessed in three different difficulty levels, which splits the objects into 'easy', 'moderate' and 'hard' subsets, such that $easy \supset moderate \supset hard$. The detections overlapping ground-truth objects of a difficulty higher than the one under evaluation do not count as TP or FP. Similarly, they do not count as FN when missed.
- The predictions lower than 30 pixels in height are not evaluated because at this scale they are more prone to error, being a source of FP.
- To compute the final "p-r curve", the recall points are approximated to a linear function, being built from a subsampled version of the sorted scores from TP list. By default, KITTI computes 41 points and we observed small variations in AP for higher number of points, thus, it is a good approximation.

Attending to the first three premises, a detector is not rewarded for detecting those labeled objects, but also not penalized. Simply discarding the indicated ground-truth regions, does not count them as TP or FN and will cause an increase in FP. Indeed, these training samples are marked as *ignored* such that predicted bounding boxes fulfilling the minimum overlap constraint on the indicated regions, do not count as FP. This is the main source of variation between the AP estimated by PASCAL vs KITTI. In general, the KITTI evaluation [Geiger et al., 2012] will lead to higher precision estimates because of the FP subtraction. This filtering of ground-truth and predicted bounding boxes is also supported by [Dollár et al., 2012].

5.3. Experiments

This section gathers the main results from the experiments carried out, attending to the previous evaluation protocol and the proposals in Chapter 4. In the subsequent divisions there is a wide range of experimental setups and results, both in training and testing image subsets for different aspects related to supervised learning of complex and high-dimensional models, object detection and orientation estimation.

5.3.1. Data cleanliness

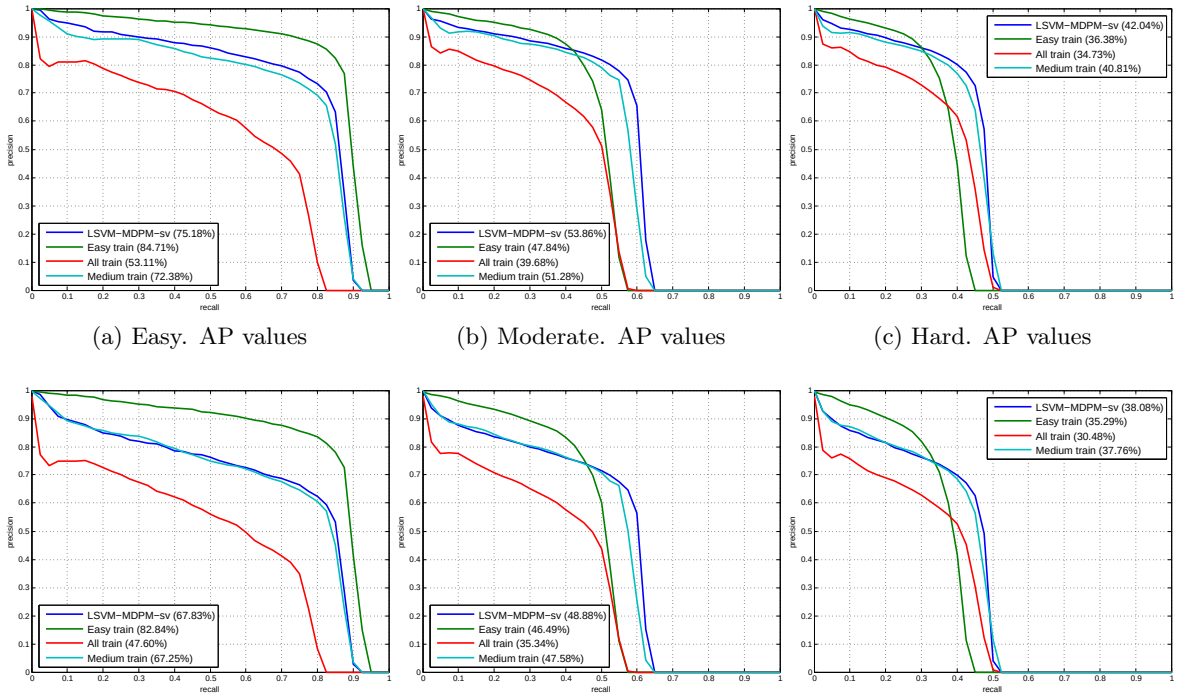
In Section 4.3, the problem of selecting the training samples was introduced. Hence, as recently published in [Yebes et al., 2014], we depict in Fig. 5.1 the results from experiments on increasing the complexity of the training samples. More specifically, we cross-validated three training modalities and compared them against the baseline. They are enumerated below:

1. The pre-trained *'LSVM-MDPM-sv'* available in [KITTI, 2012].

2. *'Easy'*: Cars with height > 40 pixels, fully visible and truncation $< 15\%$.

3. *'Medium'*: Cars included in 'easy' plus those ones with $25 <$ height < 40 pixels and/or partly occluded.

4. *'All'*: All available training samples for the class 'car'.



(a) Easy. AP values (b) Moderate. AP values (c) Hard. AP values

(d) Easy. AOS values (e) Moderate. AOS values (f) Hard. AOS values

Figure 5.1: Class 'car'. Four different training modalities are compared on each plot: *'LSVM-MDPM-sv'* [KITTI, 2012], *'Easy'*, *'All'* and *'Medium'*. These graphs show the importance of selecting a clean dataset, but general enough to represent naturalistic urban scenes. *'All'* yields the worst results (red line), while *'Easy'* (green line) outperforms only on the easy samples and downgrades for the remaining difficulty levels.

Results analysis. As it can be seen in the first column of Fig. 5.1, training on *'Easy'* yields outstanding improvements. Nevertheless, in the subsequent graphs (b, c, e, f), its performance clearly degrades for more complex samples showing higher precision at low recalls but plummeting precision at medium recall. This is caused by a higher number of both FN and FP, the latter one accentuated when increasing recall. On the other hand, training on *'All'* obtains the poorest curves, although showing less FN for heights within $25 - 40$ pixels and/or partially occluded cars. This low performance is due to the lack of cleanliness in training data: too small cars, severe occlusions and truncations, which are an important handicap for parameter learning. Hence, increasing the amount of data does not always produce better results, unless the object model and training methodology could learn complex part-based topologies and adapt to high intraclass variability.

Then, *'LSVM-MDPM-sv'* and *'Medium'* showed the best stability at all evaluation categories. Our *'Medium'* training curves are very close to the baseline *'LSVM-MDPM-sv'*, which is an available pre-trained model but we do not have access to the training setup details employed by Geiger et. al.. In fact, in this experiment we employed a very similar training subset to the one declared in [Geiger et al., 2011b], but without the additional modifications summarized in LSVM-MDPM-sv entry of the website [KITTI, 2012]. These modifications, which do not increase model complexity, seem to provide a small increment in performance. However, a correct level of supervision can provide subtle differences while training DPM, as it will be shown in Section 5.3.2.

It must be noted that observing Fig. 5.1, an increasing gap between AP and AOS appears when increasing the complexity of the training subset. This gap is around 1.5% for *'Easy'* (green lines) and 4-7% for the remaining plots depending also on the evaluation category. This loss of precision in orientation estimation can be motivated by the less informative features extracted from distant (small samples) and partially occluded cars. These errors use to belong to miss-classifications in neighboring viewpoints, which could be mitigated by reducing the number of orientation bins, although this also influences AOS by definition (see Eq. 5.3).

For reference, Fig. 5.2 displays some examples of FP predictions, which typically include cars viewed from the back, multiple cars parked on the street, cars occluded by other cars in parkings or traffic jams, parts of cars, loose fitting around the car and a few background samples.

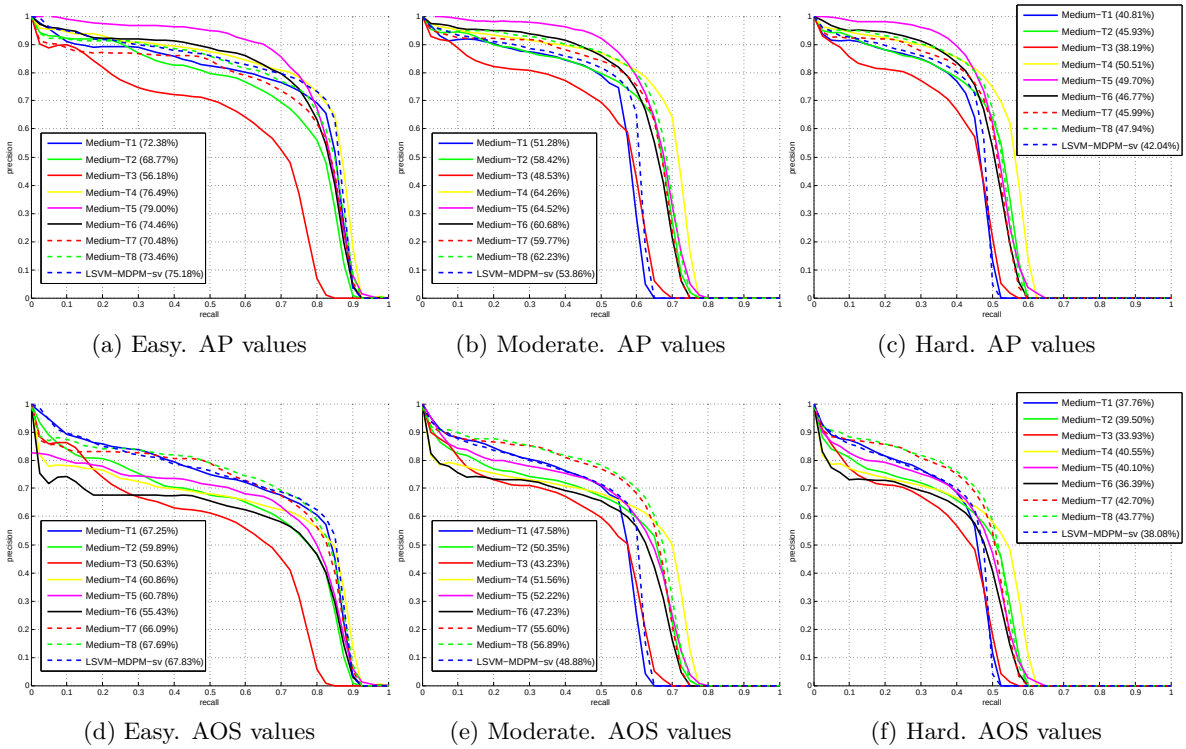
(a) Too tight bounding box (b) Part of a car (c) Multiple cars (d) Side view mirror (e) Too wide bounding box

(f) Loose fitting (g) Multiple cars (h) Background (i) Background

Figure 5.2: Examples of false positives for class car.

5.3.2. Supervised learning based only on color features

In this section, we report results for the class 'car' (Fig. 5.3) on incremental DPM modifications with the aim of tuning the parameter learning, which could lead to increased AP and AOS figures, while also getting a better knowledge of DPM strengths and weaknesses. The tests configurations are based upon the explained aspects in Section 4.3. Besides, *'Medium'* cars are employed as positive labeled samples, in accordance with the results from previous section. It must be noted that each experiment (5 trainings) can take 100-170 hours on an i7 CPU machine, depending on the experiment configuration described next.



(a) Easy. AP values (b) Moderate. AP values (c) Hard. AP values

(d) Easy. AOS values (e) Moderate. AOS values (f) Hard. AOS values

Figure 5.3: Class 'car'. Supervised learning of DPM models based only on color features. For training, *'Medium'* samples are employed. Every column corresponds to one of the evaluated difficulty levels and every plotted curve to the 8 different experiments carried out.

- **Medium-T1:** 16 model components, bilateral asymmetry, L-SVM regularization constant C=0.001 and default root filter area limited to $3,000 - 5,000$ pixels.

- **Medium-T2:** Analyzing previous results, several small cars are missed, then, we propose to allow smaller root filters of $[1000, 5000]$ pixels. This impacts during latent search on the image scale pyramid, producing a detection improvement for the difficult samples (Fig. 5.3.b and 5.3.c). Also, there is a better orientation estimation at higher recalls (Fig. 5.3.e and 5.3.f). However, the shortcoming related to some smaller model components, with lower level of detail, is that AP and AOS decrease for easy samples (Fig. 5.3.a).

- **Medium-T3:** Considering the comments above, we propose to also enlarge the upper limit to 6,000 pixels to favor detection of 'easy' samples (usually closer objects). Besides, we impose a loose fit for latent parts training in order to give more flexibility to the model, moving their overlap requirement from 70% to 60%. As a result of this relaxation, the learned parameters are not representative enough causing a loss of precision for all cases (continuous red plots in Fig. 5.3).

- **Medium-T4:** Consequently, we opt to fix a tighter constraint, i.e. 80% overlap during latent parts search. This yields a medium gain for easy samples, but an important boost for the difficult ones. However, the orientation estimation shows a slight gain in precision (actually below previous curves at low and medium recalls) and AOS falls a 7% for easy samples (yellow plots in Fig. 5.3), which may be related to overfitting.

- **Medium-T5:** In this test, the cropping of hard negatives from strictly positive images is enforced during data mining. In particular, the selected negatives must not overlap more than 20% with a ground-truth sample (like in LSVM-MDPM-sv [KITTI, 2012]). The first bootstrapping step of DPM remains harvesting random negatives from strictly negative images. In spite of the increase in training time, we gain precision (magenta lines in Fig. 5.3) thanks to the rise in the number of background samples. However, AP values are similar to previous experiment due to an earlier drop of precision at upper recalls. Although AOS replicates this behavior, the AP-AOS gap is still too wide. Viewpoint discrimination is benefited for difficult samples given the modifications carried out so far.

- **Medium-T6.** This case adds the mirroring of positive samples and the fixation of the car viewpoints during the merging stage (see Section 4.1.2) of the model components, i.e. the viewpoint is not latent but taken from the ground-truth labels. Additionally, 'DontCare' regions are ignored during hard negatives mining. Despite these modifications, we observe a lower performance for all cases (black plots in Fig. 5.3).

- **Medium-T7** Interestingly, using *Medium-T6* and reducing back the overlap for latent parts to 70%, a superior precision in orientation estimation is achieved (red dashed plots in Fig. 5.3), but lower AP values. Thus, it is confirmed that the overlap requirement during latent search greatly influences the performance in orientation estimation.

- **Medium-T8** Finally, selecting the trade-off constraint of 75% overlap, we obtain a moderate AP and AOS increase at all levels, which is better than the baseline LSVM-MDPM-sv [KITTI, 2012] for evaluation levels 'moderate' and 'hard'. As can be seen, this configuration yields 'p-r' curves for detection that are below the highest *Medium-T5*. However, it behaves better in orientation estimation. Therefore, this final configuration shows a good compromise in precision between both tasks.

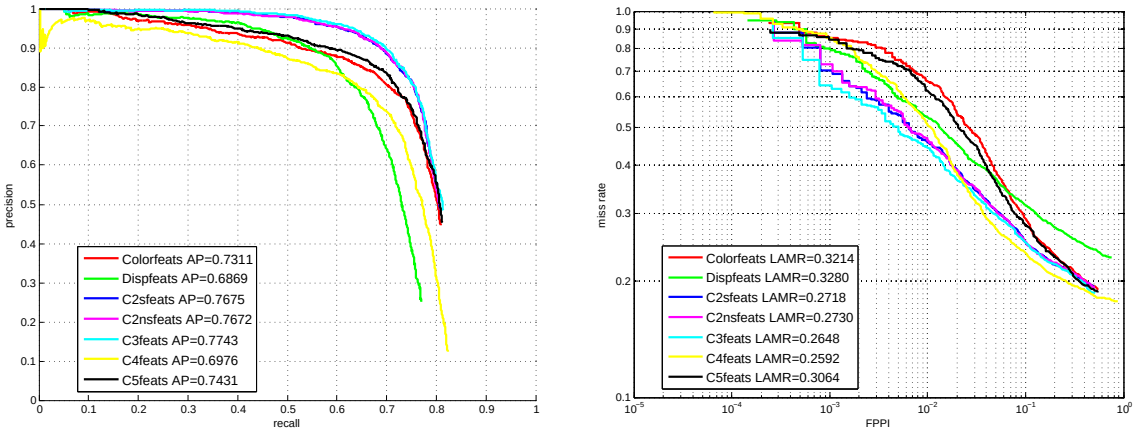
5.3.3. Supervised learning based on 3D-aware features

This section presents several experiments to evaluate the performance of the 3D-aware contributed features. Firstly, we assess the selection of the most appropriate features employing training samples of the class 'car' for a single orientation. Thus, the mixture model in DPM framework is initialized with one component. Although three different views were tested, Fig. 5.4 shows the results for $3\pi/4$, which also represents the main conclusions obtained from tests with the other viewpoints. As initial performance feedback, these comparative plots have been calculated with the PASCAL evaluation protocol [Everingham et al., 2010]. Besides, the "miss rate vs fppi" curves [Dollár et al., 2012] are also drawn for a more complete validation.

For reference, the model learned from *Colorfeats* is depicted in Fig. 5.5 that illustrates the orientation of the vehicle in the scene. Nonetheless, seven features are compared in Fig. 5.4, which are directly related to the ones defined in Section 4.2 as it is detailed on the figure caption. From the results, it can be derived that HOG descriptors on the disparity map (*Dispfeats*) achieve high detection ratios. It must be noted that disparity is single-channel, sparse and has some measurements errors. Therefore, it can be confirmed the viability of the use of disparity and

its gradients to enhance the DPM framework. Nevertheless, *Dispfeats* does not outperform the reference model based on color images, as shown in green in Fig. 5.4.

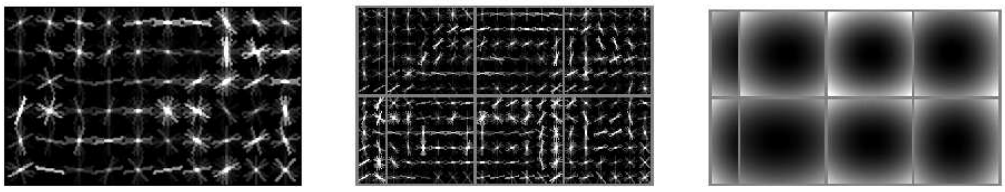
In relation to the 3D-aware features *C2, C3, C4 and C5* proposed in Section 4.2, *C3* outperforms the others at the cost of a big increase in descriptor length (92 dimensions). However, it produces a small AP increment (1%) over *C2* (64 dimensions), which instead produces an improvement of 3.5% with respect to color. Moreover, there are no appreciable differences in detection performance when scaling the disparity to account for the ratio between the object and filter heights in pixels (check Section 4.2.1 for details). On the other hand, *C4* and *C5* do not yield significant gains. Consequently, we opted to intensify the research on *C2* and its variants *C8B1* and *C8B2*, and also evaluate new features as *C6* and *C7*.



(a) Precision-recall curves

(b) Miss rate vs FPPI curves

Figure 5.4: Comparative evaluation of object prediction with 3D-aware features. Class 'car' on a single viewpoint (e.g. the trained model in Fig. 5.5). This is a first evaluation of the 3D-aware features based on PASCAL protocol. *Colorfeats* and *Dispfeats* are HOG descriptors on color and disparity images respectively. *C2sfeats* and *C2nsfeats* correspond to the proposed feature *C2* with and without disparity scaling. The remaining descriptors are exactly as described in Section 4.2. Although *C3feats* yields the best AP and LAMR values, *C2feats* is very close in detection performance but it has the advantage of a lower descriptor dimensionality.



(a) Root filter

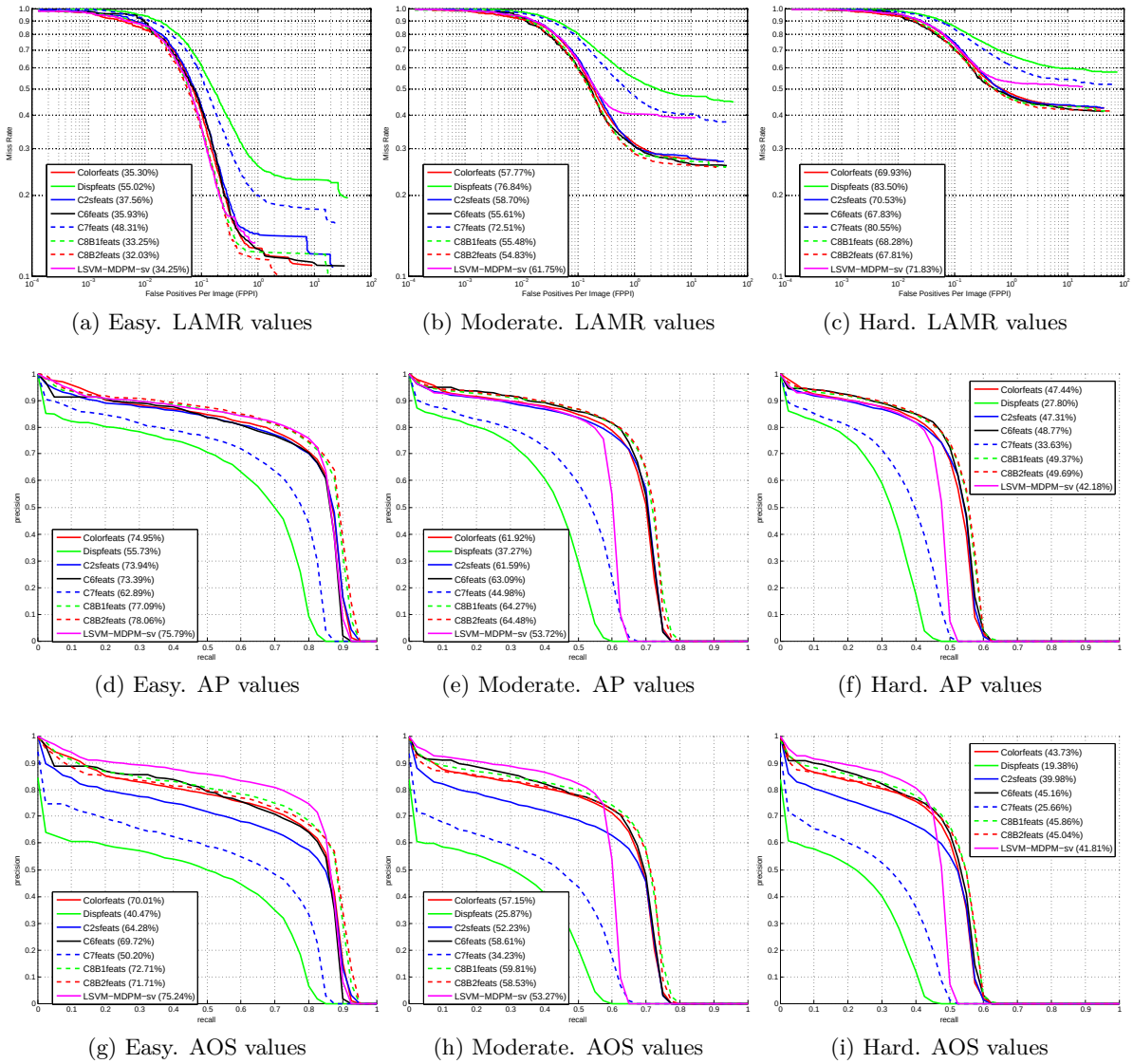
(b) Part filters

(c) Deformation weights

Figure 5.5: Sample of a learned model for the class 'car' and viewpoint $3\pi/4$. This model has been trained with only one component and employing HOG features from color images.

After the first derivations from the previous experiments, Fig. 5.6 shows a more extensive analysis on the 3D-aware features, but in this case employing the KITTI evaluation protocol as exposed in Section 5.2. Fig. 5.6 summarizes the most relevant results, although several related experiments are attached in Appendix B. The setup for the supervised learning of the models employed in Fig. 5.6 is *Medium-T8* due to the reasons in Section 5.3.2.

The figure is divided in 9 subgraphs, where each row corresponds to "miss rate vs FPPI" curves and "precision vs recall" plots for detection and orientation estimation, respectively. The columns are linked to the three difficulty levels evaluated in the KITTI challenge. Besides, each of the 9 subgraphs includes a comparison of 7 features (Section 4.2) against the pre-trained baseline LSVM-MDPM-sv [KITTI, 2012]. Nevertheless, it must be clarified that *C2sfeats* refers to *C2* descriptors computed from scaled disparity. Despite the previous conclusions on scaling disparity, the experiments in Appendix B show that adding this scaling is beneficial for object orientation estimation.



(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure 5.6: Comparative evaluation of the 3D-aware features performance for the class 'car'.

From the tests above it is demonstrated that *C8B1* and *C8B2* (green and red dashed lines) yield the best object detection and orientation estimation ratios. Indeed, they present the highest AP and AOS and the lowest LAMR. Compared to *C2s* (blue continuous curve), they produce a moderate boost in performance that is more clear for the viewpoint prediction. Besides, they have a shorter dimensionality than *C2s* (check Section 4.2) and lead to speedups, which is

more notable during training, although it also benefits the prediction stage. In addition, they outperform the baseline for all difficulty levels, but the gain is more prominent for 'moderate' and 'hard' samples. Interestingly, *C8B2* (red dashed line) peaks in the AP for all evaluated levels, but *C8B1* (green dashed line) is superior for AOS, which can be explained by the nature of its contrast-sensitive features ($0 - 2\pi$ gradient orientations). Indeed, *C8B1* is preferred because it yields 1% increase over *C8B2* at all difficulties.

The remaining tested features also produce interesting results. For example, *C6* descriptor (black plots), which is composed of HOG color and 4 statistics of disparity (no gradients), presents AP, AOS and LAMR figures close to *C8B1* and *C8B2* for the difficult samples. However, it has a slight inferior performance for 'easy' subset, which could be explained by the shorter length of the descriptors when more information is available (easy samples correspond to fully visible and closer objects). In relation to *C7* (blue dashed curves), which computes the minimum between HOG color and HOG disparity descriptors, it does not show any contribution as demonstrated by its low precision values in all cases. This seems to be highly influenced by the *Dispfeats* with the lowest precision (green continuous lines) in all plots. Then, the use of disparity alone does not provide enough visual information about the objects.

In summary, the biggest gains are obtained when using the 3D-aware features *C8B1* and *C8B2*, which clearly outperform the baseline MDPM-LSVM-sv. Mainly, there are two reasons: the appropriate supervised learning of DPM with the characteristics in Section 4.3 and the richer models learned with the employment of 2.5D data.

As a result of the experiments and conclusions above, in the next sections the emphasis is on *C8B1* features in order to broaden the tests in three extents:

1. Show results for the "adaptive parts" approach, which increases model flexibility and helps to adapt the learned filters to the intraclass variability of each object category.

2. Expand the experiments for the classes 'pedestrian' and 'cyclist' to measure the performance of the location and orientation prediction tasks.

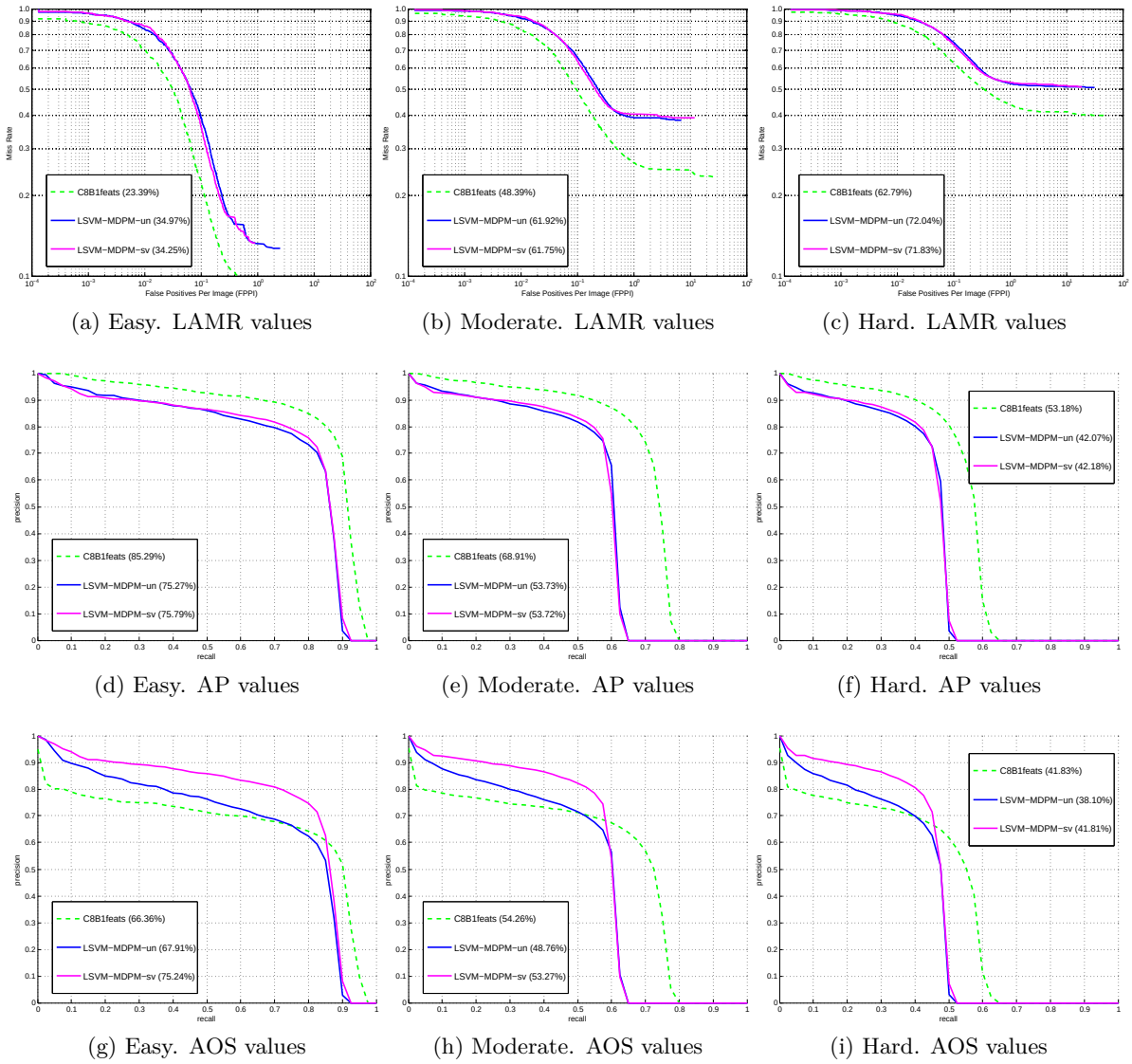
3. Display the public results submitted to [KITTI, 2012], which calculates the AP and AOS figures on the test set and ranks the method among other works of the state of the art.

5.3.3.1. Experiments when employing adaptive parts

Fig. 5.7-5.9 depict the 'p-r curves' when adding the "adaptive parts" approach (check Section 4.3) for cars, pedestrians and cyclists. They have been obtained with 5-fold cross-validation on the training data while also employing the best performing DPM setup configuration ('Medium-T8') and the features *C8B1*. Besides, they are compared against the pre-trained baseline models *LSVM-MDPM-un* and *LSVM-MDPM-sv*, which correspond to the unsupervised and supervised training versions described in [Geiger et al., 2012].

In brief, it is observed an important precision increase for the class 'car' compared to the baselines, which is replicated by the class 'cyclist' at a lower scale. The main factors contributing to this success are the appropriate supervised training, the *C8B1* features and the adaptive parts.

However, pedestrian prediction shows reduced ratios that are below the baseline models. This is caused in part due to poor disparity measurements for distant pedestrians and false detections of cyclists, which are usually standing at traffic lights or walking and they also have a very similar appearance in frontal view at low scales. On the other hand, there is a reduction of precision in orientation estimation for all the classes due to incorrect discretization in the model components. Our intuition is that these features and the learned filters are more invariant to intraclass differences, causing less discrimination power in objects viewpoint.

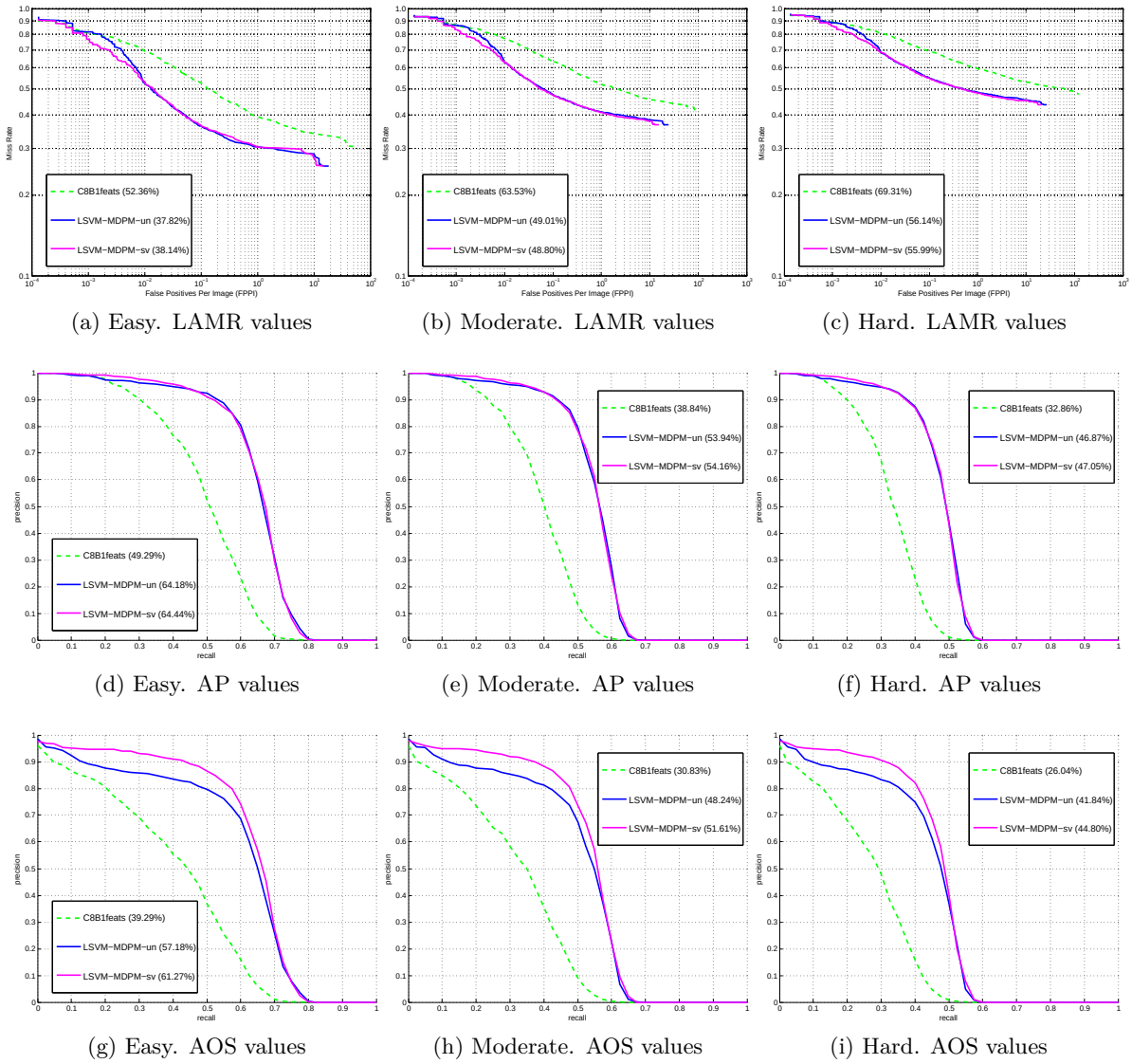


(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure 5.7: Evaluation of C8B1 features and adaptive parts for the class 'car'. As can be observed in the plots of first and middle rows, our approach outperforms the baselines for object detection, both in lower number of FP and FN. However, the orientation estimation shows lower prediction power, which is a drawback for 'easy' samples.

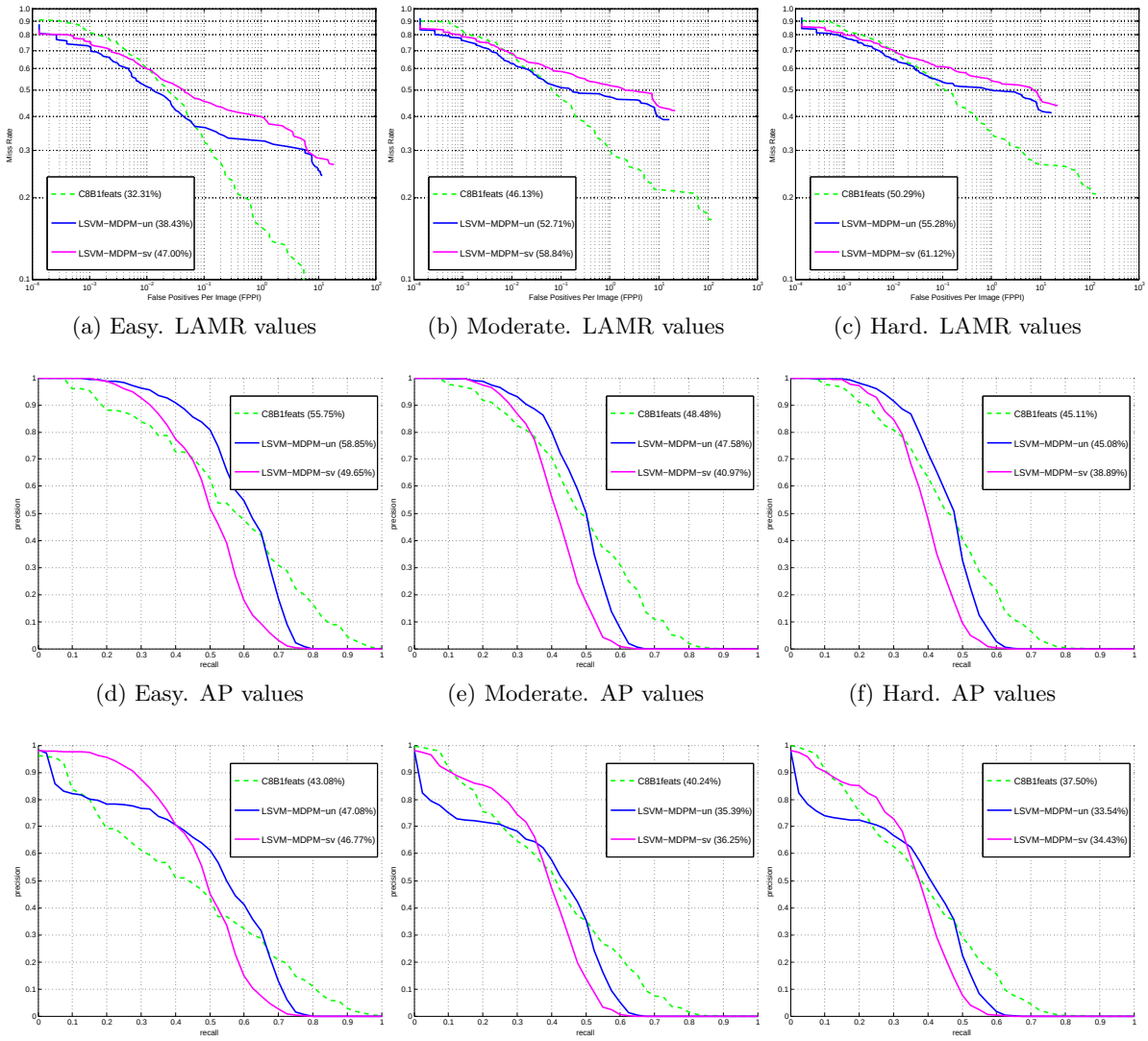


(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure 5.8: Evaluation of C8B1 features and adaptive parts for the class 'pedestrian'. It can be observed that we achieve poor pedestrian predictions, yielding LAMR, AP and AOS figures worst than the baseline. This can be explained by the peculiarities of 'pedestrian' class in urban scenes, so that their modeling should be better approached with specific pedestrian detection methods as already proposed in the literature [Dollár et al., 2012].



(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure 5.9: Evaluation of C8B1 features and adaptive parts for the class 'cyclist'. Despite the increase in the number of FP observed in a), b) and c), this is compensated by an important reduction in the number of FN. Then, the green curves are close to the baselines and LAMR, AP and AOS figures yield a prediction improvement compared to LSVM-MDPM-sv.

5.3.4. Results on unseen data: the KITTI testset

Employing the setup configuration and features described in previous section, which yielded good results, we learned a final model for each object category from the whole training dataset. Then, we run the prediction engine on the testing images and submitted the results to KITTI website [KITTI, 2012] in order to evaluate and rank our contributions. At the date of submission (28th April 2014), the state of the published results from different state-of-the-art approaches is shown in Tables 5.7-5.12. The first three tables present the object detection performance in AP(%) and the next three, the joint object detection and orientation estimation precision in AOS(%). Besides, the column 'DS' refers to the data setting (*c* -> color images; *st* -> stereo color images and *la* -> laser data). The column 'Rt' corresponds to the estimated runtime when predicting objects on a single test image.

Due to visualization issues, we enumerate the references for each entry below. It must be remarked that all the approaches are based upon modifications of the successful Discriminative Part-based Models [Felzenszwalb et al., 2010b], excluding the *mBoW* method for laser data. For more details on the particular approaches and additional updates, please visit [KITTI, 2012].

- *SVM-Res*: SVM rescoring[3].
- *SubCat*: Learning object SubCategories[3].
- *OC-DPM*: Occlusion Patterns Deformable Part Model [Pepik et al., 2013].
- *MDPM-un-BB*: DPM with bounding box prediction [Felzenszwalb et al., 2010b].
- ***DPM-C8B1*: Our supervised DPM learning with C8B1 features**.
- *LSVM-MDPM-sv*: Discriminatively Trained Deformable Part Models with Supervised Training [Geiger et al., 2011b].
- *LSVM-MDPM-us*: Discriminatively Trained Deformable Part Models with Unsupervised Training [Felzenszwalb et al., 2010b].
- *mBoW*: Mixture of Bag-of-Words [Behley et al., 2013].
- *DA-DPM*: Domain Adaptive DPM[4].

From the tables, we rank first in cyclists detection and fourth on cars bounding boxes prediction. However, the results for pedestrian are very poor, as we already seen in validation experiments. The corresponding precision-recall curves on the test images are not attached to this document because they do not provide any additional information. In fact, the can be directly accessed on [KITTI, 2012], but they are represented individually for our method. They cannot be plotted against the other approaches (as we did for training and validation in previous sections). Hence, the information contained in the tables is more helpful for results publication and analysis.

[3] Anonymous submission

[4] Under submission: J. Xu, S. Ramos, D. Vázquez and A. López, "Domain Adaptive Deformable Part-based Model".

Table 5.7: Object detection evaluation. CAR

R	Method	DS	Mod.	Easy	Hard	Rt	Environment
1	SVM-Res	c	67.49	78.11	54.28	10 s	4cores @2.5Ghz (Matlab)
2	SubCat	c	60.37	77.90	49.61	0.4 s	4cores @2.5Ghz (Matlab,C/C++)
3	OC-DPM	c	65.95	74.94	53.86	10 s	8cores @2.5Ghz (Matlab)
4	**DPM-C8B1**	st	**60.99**	**74.33**	**47.16**	28 s	4cores @2.5Ghz (Matlab,C/C++)
5	MDPM-un-BB	c	62.16	71.19	48.43	60 s	4 core @ 2.5 Ghz (Matlab)
6	LSVM-MDPM-sv	c	56.48	68.02	44.18	10 s	4cores @3.0Ghz (C/C++)
7	LSVM-MDPM-un	c	55.42	66.53	41.04	10 s	4cores @3.0Ghz (C/C++)
8	mBoW	la	23.76	36.02	18.44	10 s	1core @2.5Ghz (C/C++)

Table 5.8: Object detection evaluation. PEDESTRIAN

R	Method	DS	Mod.	Easy	Hard	Rt	Environment
1	DA-DPM	c	45.51	56.36	41.08	21 s	1core @3.5Ghz (Matlab, C/C++)
2	LSVM-MDPM-sv	c	39.36	47.74	35.95	10 s	4cores @3.0Ghz (C/C++)
3	LSVM-MDPM-un	c	38.35	45.50	34.78	10 s	4cores @3.0Ghz (C/C++)
4	mBoW	la	31.37	44.28	30.62	10 s	1core @2.5Ghz (C/C++)
5	**DPM-C8B1 (ours)**	st	**29.03**	**38.96**	**25.61**	13 s	4cores @2.5Ghz (Matlab, C/C++)

Table 5.9: Object detection evaluation. CYCLIST

R	Method	DS	Mod.	Easy	Hard	Rt	Environment
1	**DPM-C8B1**	st	**29.04**	**43.49**	**26.20**	7 s	4cores @2.5Ghz (Matlab, C/C++)
2	LSVM-MDPM-un	c	29.88	38.84	27.31	10 s	4cores @3.0Ghz (C/C++)
3	LSVM-MDPM-sv	c	27.50	35.04	26.21	10 s	4cores @3.0Ghz (C/C++)
4	mBoW	la	21.62	28.00	20.93	10 s	1core @2.5Ghz (C/C++)

Table 5.10: Joint object detection and orientation estimation. CAR

R	Method	DS	Mod.	Easy	Hard	Rt	Environment
1	OC-DPM	c	64.42	73.50	52.40	10 s	8cores @2.5Ghz (Matlab)
2	LSVM-MDPM-sv	c	55.77	67.27	43.59	10 s	4cores @3.0Ghz (C/C++)
3	**DPM-C8B1**	st	**50.32**	**59.51**	**39.22**	28 s	4cores @2.5Ghz (Matlab, C/C++)
4	SVM-Res	c	30.38	35.02	24.87	10 s	4cores @2.5Ghz (Matlab)

Table 5.11: Joint object detection and orientation estimation. PEDESTRIAN

R	Method	DS	Mod.	Easy	Hard	Rt	Environment
1	LSVM-MDPM-sv	c	35.49	43.58	32.42	10 s	4cores @3.0Ghz (C/C++)
2	**DPM-C8B1**	st	**23.37**	**31.08**	**20.72**	13 s	4cores @2.5Ghz (Matlab, C/C++)

Table 5.12: Joint object detection and orientation estimation. CYCLIST

R	Method	DS	Mod.	Easy	Hard	Rt	Environment
1	LSVM-MDPM-sv	c	22.07	27.54	21.45	10 s	4cores @3.0Ghz (C/C++)
2	**DPM-C8B1**	st	**19.25**	**27.25**	**17.95**	7 s	4cores @2.5Ghz (Matlab,C/C++)

The precision results above are in accordance with our expectations after 5-fold cross-validation. Therefore, the conclusions and analysis commented in subsection 5.3.3.1 are also applicable for the test set, i.e. we outperform the baseline LSVM-MDPM for cars and cyclists

detection, but not for the orientation estimation. Nevertheless, higher performance for the 'moderate' and 'hard' samples was expected according to our validation experiments. In fact, we achieved lower LAMR and higher AP at those difficulty levels compared to the baseline for the classes 'car' (Fig. 5.7) and 'cyclist' (Fig. 5.9). Although one could think on overfitting as one of the reasons, the real cause comes from a singular limitation of KITTI evaluation protocol: samples with a height < 30 pixels are ignored. This constraint is not implemented on the public evaluation code available in [KITTI, 2012], but it is enforced on KITTI server when submitting the predictions on the test subset. Therefore, we would have expected better performances for the classes 'car' and 'cyclist' when detecting the non-easy samples.

In relation to runtime estimations, we specify the average values for each category on the joint object detection and orientation estimation. However, depending on the number of components of the learned mixture models, the execution time during prediction is different. Besides, the KITTI website only provides one input for publishing the runtime. Consequently, all the state-of-the-art methods ranked on it have a single estimated time for all the evaluations. Hence, we do not reckon on direct comparisons in this aspect because it is not clear whether the published runtimes are the average, highest or lowest delays. Therefore, we can conclude that we are among the state of the art and can get speedups when moving the whole prediction process to optimized C++ code [Dubout and Fleuret, 2013, Kokkinos, 2012a].

The figures on the next pages show some of the predictions on test images, which have been inferred by our **DPM-C8B1** and painted with KITTI development kit. However,they must not be confused with the ground truth, because we remark again that it is not available for the test images. Figures 5.10 to 5.15 display a set of frames with correct object predictions surrounded by a green box and some FP and FN examples. Particularly, FPs have been manually marked with a red bounding box, whilst FNs can be identified as the cars, pedestrians and cyclists not detected in the scenes. It must be noted that trucks or vans detected as cars are neighboring classes and not considered as FP. A similar consideration should have been done for pedestrians and cyclists, because there are many cyclists stopped at traffic lights or walking on the streets that are detected as pedestrians. However, this distinction is not made by KITTI evaluation protocol.

Observing the displayed predictions, many challenging object instances from the three classes are correctly detected. However, several FPs and FNs occur in these naturalistic urban scenes. The most typical cases are cyclists and pedestrians confused between each other, which can be interpreted as a normal detector behavior given the high similarities for some poses. Besides, there are many false positives in trees and other vertical structures of the city for these classes, which matches with the vertical gradients learned by the models. In relation to cars, there exist several miss-classifications due to lorries, trucks and background areas with very similar visual appearance. In addition, typical FN cases for cars are occluded vehicles in road sides where they are parked and near to each other. Furthermore, the lower performance for distant objects is usually related to cyclists and pedestrians, where disparity cannot provide accurate cues and visual appearance features do not achieve a sufficient image patch description. Another source of false positives are the repeated detections of an object at bigger scales and not tight enough predictions, i.e. loose fitting of the 2D bounding box around the object.

Figure 5.10: Examples of predicted labels in KITTI testing frames with TPs, FPs and FNs.

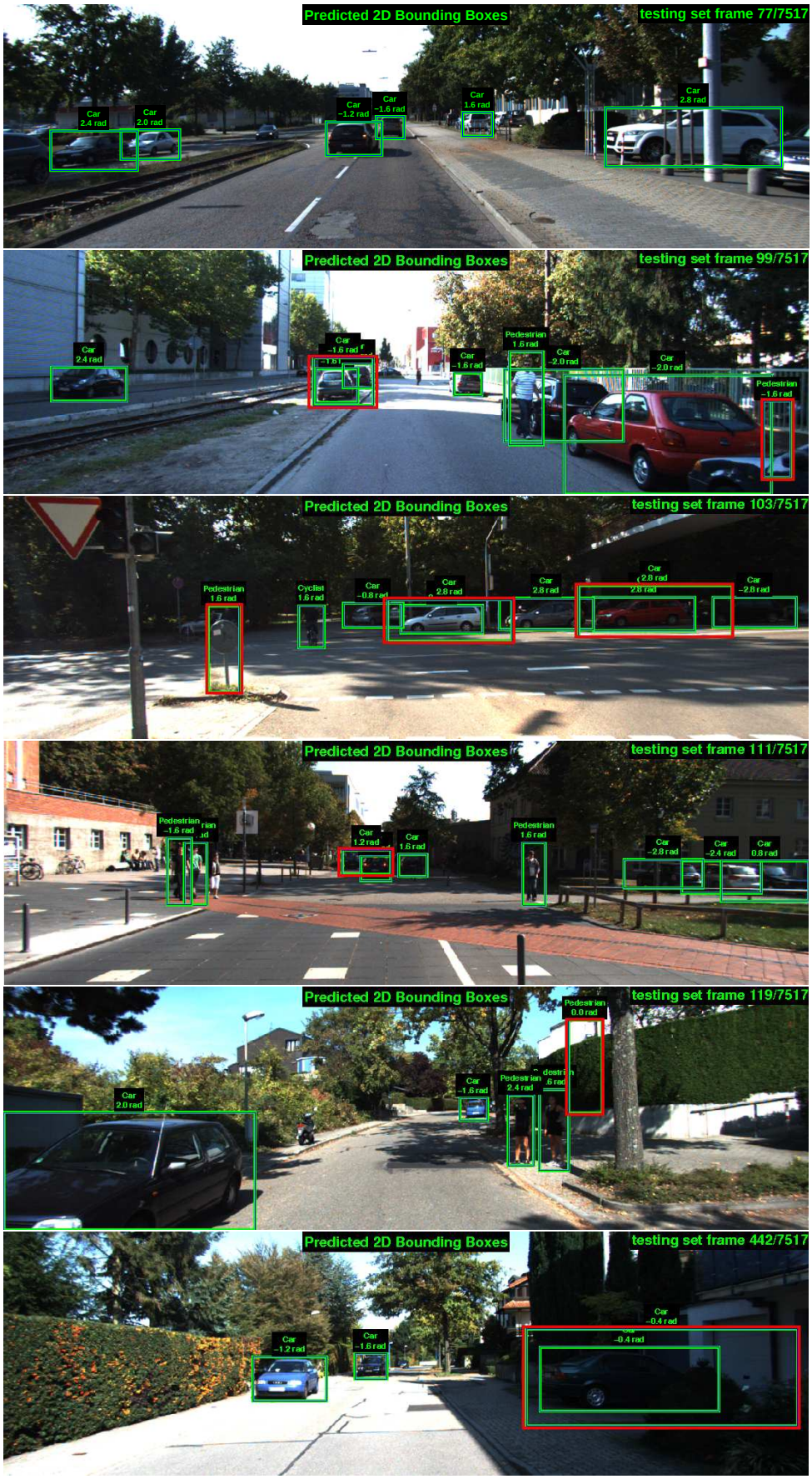


Figure 5.11: Examples of predicted labels (cont.).

Figure 5.12: Examples of predicted labels (cont.).

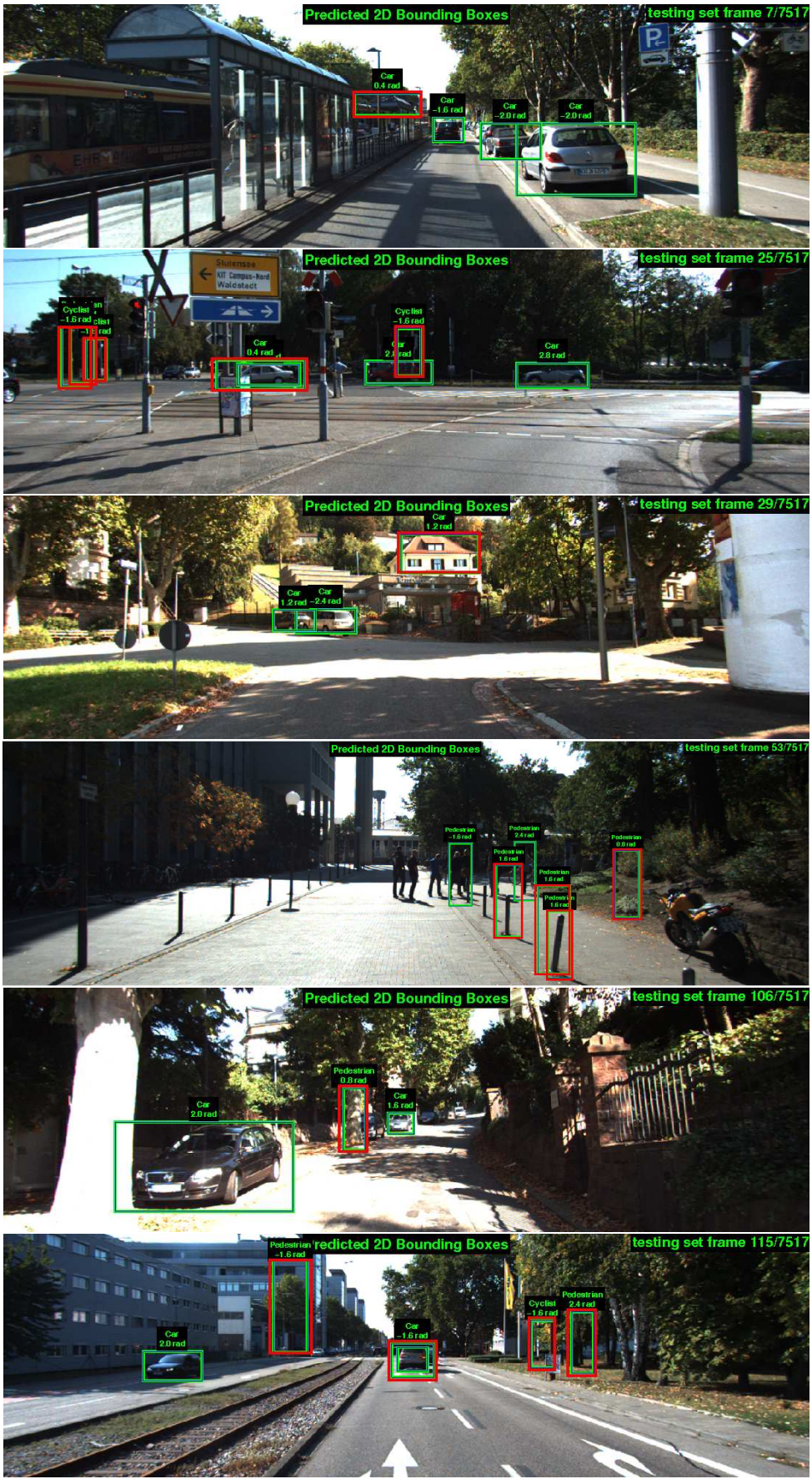


Figure 5.13: Examples of predicted labels (cont.).

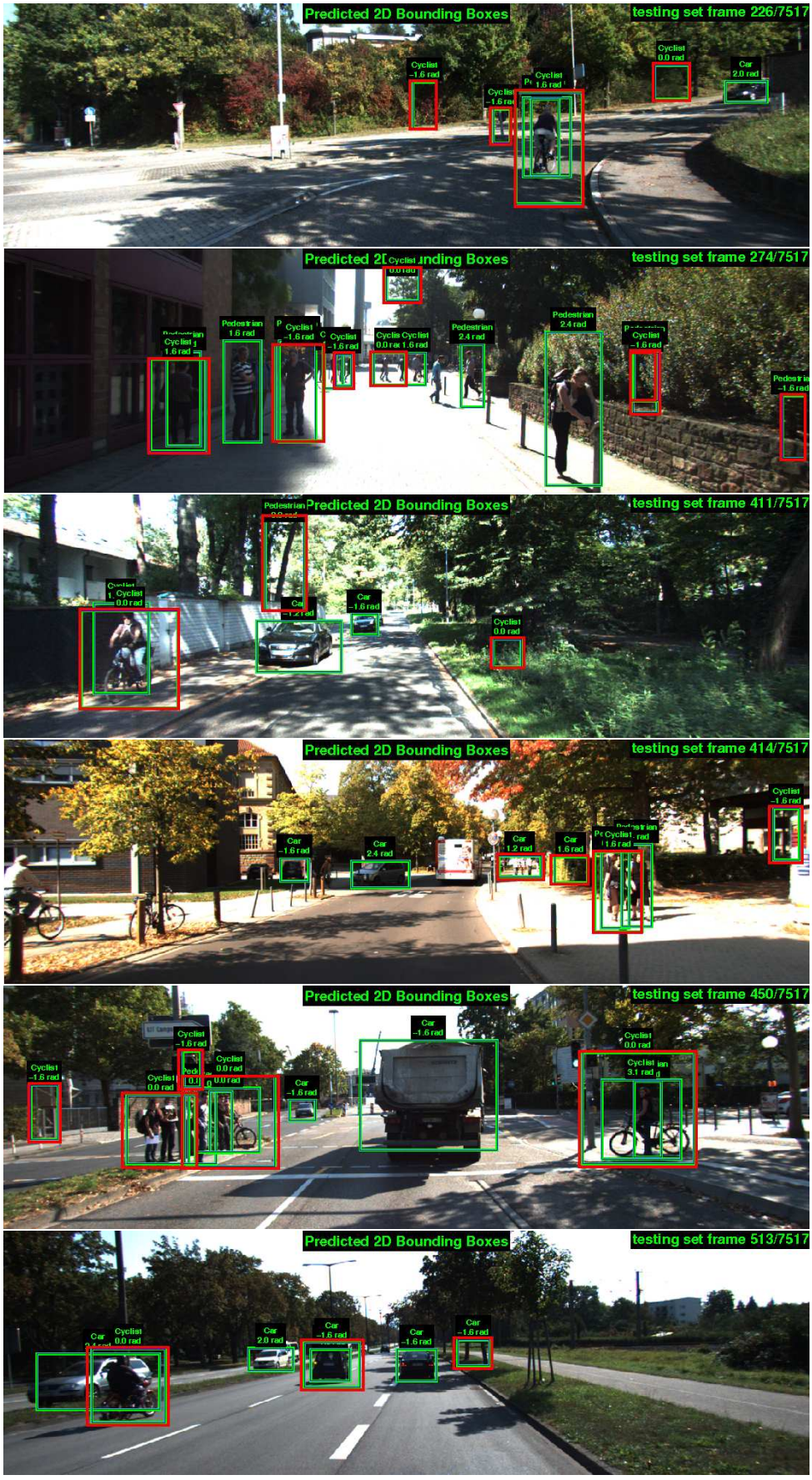


Figure 5.14: Examples of predicted labels (cont.).

Figure 5.15: Examples of predicted labels (cont.).

5.3.5. Results of the additional approaches

The results contained in the next two subsections regard the additional approaches on top of DPM that this Thesis has proposed in Chapter 4, i.e. feature whitening and stereo consistency.

5.3.5.1. Feature whitening

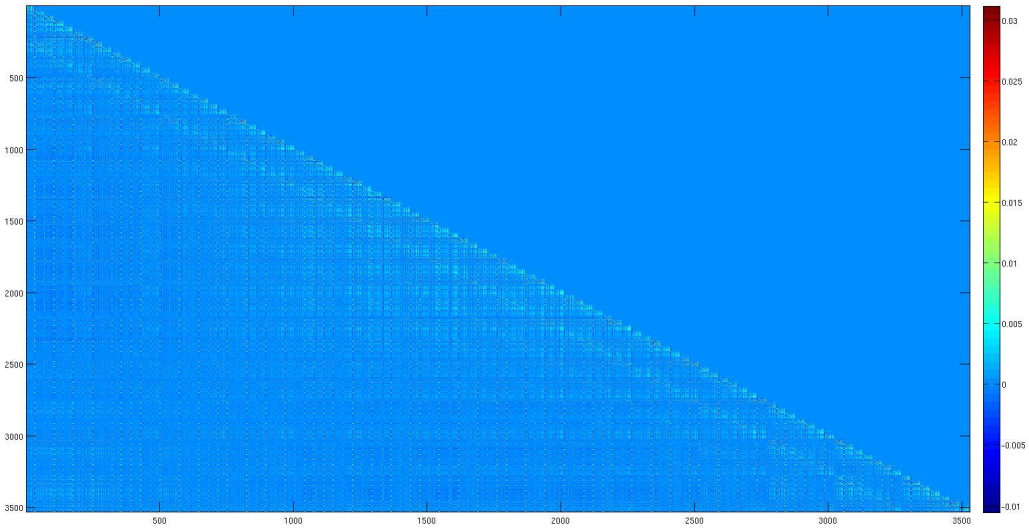
The results and conclusions reported here have been obtained after a set of experiments on feature whitening with the methodology proposed in Section 4.4.1. It must be noted that different implementations have been tested in order to integrate this approach into the DPM training pipeline, which is not straightforward given the size and complexity of the original source code. In particular, the covariance computation and the eigenvalue decomposition have been tested with PCA, online covariance computation and Cholesky decomposition. However, the most promising results have been obtained for the approach described in Section 4.4.1.

All the experiments reported below have been carried out on the class 'car', because it is the KITTI category with the highest number of samples. Thus, better estimations of the covariance matrix can be obtained and a higher number of whitened features can be fed to the learning process. The next figures visualize the covariance matrices for the 'background' and 'car' classes. More specifically, as explained in Section 4.4.1, the 'background' data samples are collected from the training images without ground-truth labels, randomly picking 12 subwindows of fixed size (128×72 pixels $= 7 \times 14$ HOG cells) at 8 different resolutions of the training images (7,480), obtaining a set of 718,080 features that are fed to Algorithm 4. The fixed size corresponds to the biggest root filter for the class 'car', which depends on the maximum configured area in pixels and the dataset pooling during DPM initialization. On the other hand, the covariance of the 'car' class is computed from the ground-truth bounding boxes and their mirrored versions, all of them resized to 128x72 pixels, obtaining 57,484 features in total. Fig. 5.16 shows the resultant covariance matrices for these classes and Fig. 5.17 provides a zoom on the left upper corner of the matrices. Only the lower triangular part is estimated by our online/incremental algorithm because the covariance is symmetric. Then, the figures visualize this part. These examples have been built from our proposed 3D-aware features *C6feats* (check Section 4.2), which have a length of 36 elements.

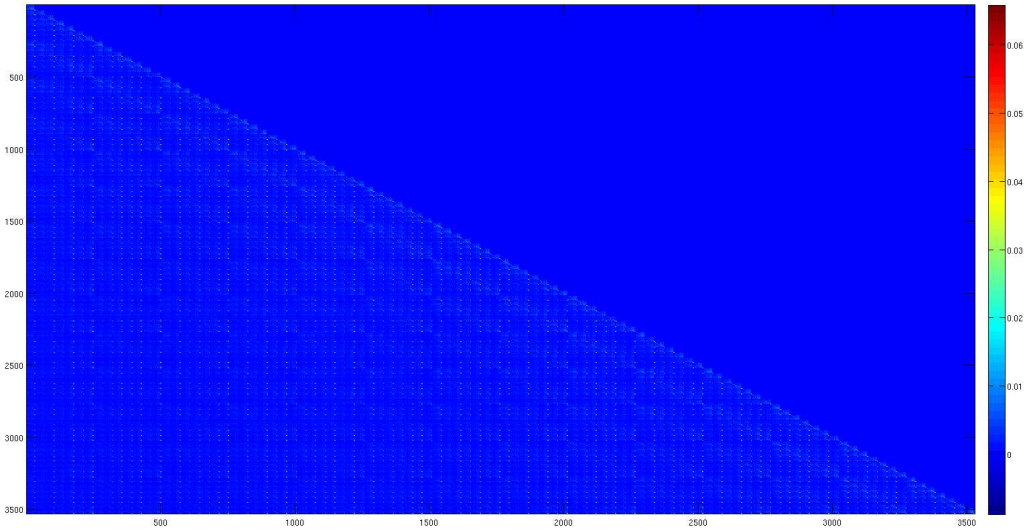
As it can be seen in the figures, the covariance matrices for cars and background have very similar patterns, but different scaling, which may be directly related to the amount of samples that were employed for each class (more background image patches were collected). Besides, the repetitive $d \times d$ "squares" ($d = 36$ for *C6feats*) in Fig. 5.17 come from the fact that the covariance matrix is computed as a spatial correlation between the image cells[5]. Furthermore, high correlations appear for the features in the positions 31, 32, 33 and 34 in different image cells[6]. They correspond to the disparity statistics (max, min, mean, median) added to HOG color that this Thesis proposes for *C6feats*.

[5]This correlation is based on [Hariharan et al., 2012] as we described in Section 4.4.1

[6]Each HOG cell in the image is displayed as $d \times d$ square in the covariance

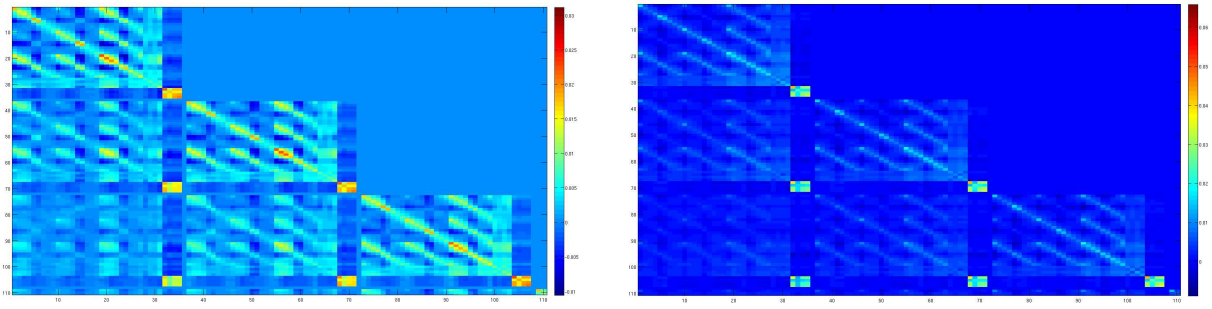


(a) Car. Matrix values from -0.01 to 0.03



(b) Background. Matrix values from -0.01 to 0.06

Figure 5.16: Feature whitening. Covariance matrices of our *C6feats* computed from the KITTI dataset. Their size is $M \times M$, being $M = 7 \cdot 14 \cdot 36 = 3,528$.

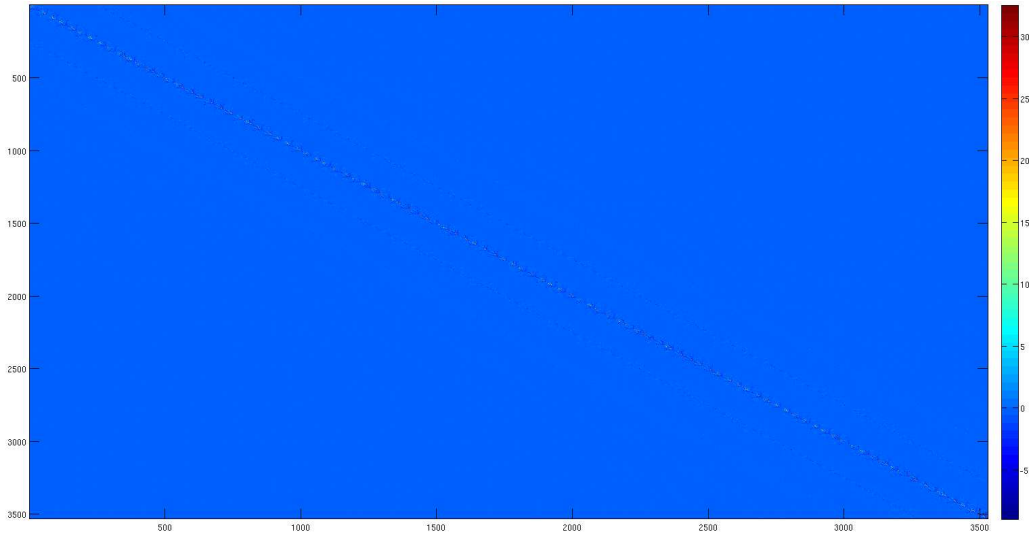


(a) Car (b) Background

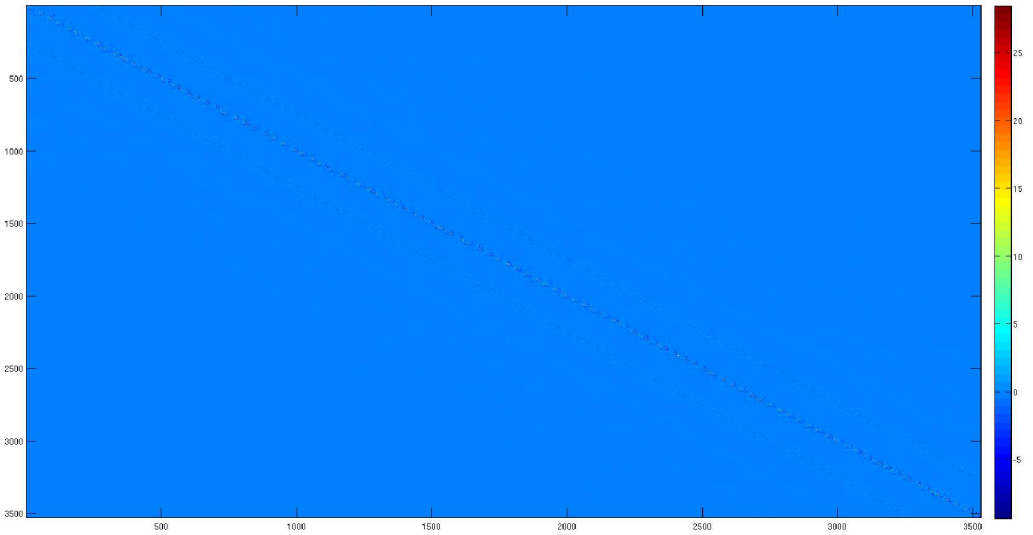
Figure 5.17: Zoom in the left upper corner of the covariances. Showing the first 110 elements (3 HOG image cells). Each cell is a square region of size 36×36 in the covariance.

Another interesting property that can be discovered from the covariance submatrices in Fig. 5.17 is a set of short and lighter color stripes on every 36×36 square on the diagonal and surrounding positions. They are related to the correlations between the contrast sensitive and contrast insensitive subfeatures of the HOG color descriptor (check the diagram in Fig. 4.12). Indeed, the 9 contrast insensitive elements are collapsed from the 18 contrast sensitive values.

In addition, figures 5.18 and 5.19 show the corresponding whitening matrices obtained with the ZCA algorithm as described in Section 4.4.1. Again, these matrices have a very similar aspect for both 'background' and 'car' classes, but also a very similar scaling. This confirms the findings in [Hariharan et al., 2012], which stated that the whitening of the HOG feature space for different object classes can be generalized by randomly collecting a large set of background samples. Then, it is not needed to compute a particular covariance matrix for each object class, given the KITTI dataset.

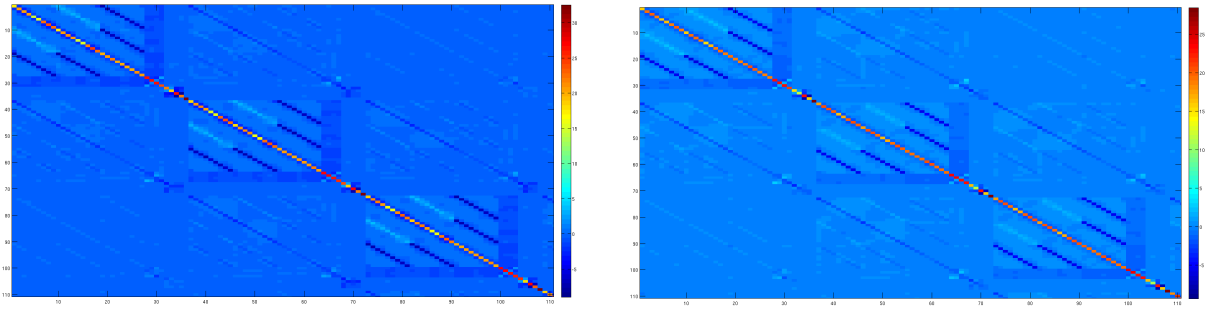


(a) Car. Matrix values from -5 to 30



(b) Background. Matrix values from -5 to 25

Figure 5.18: Whitening matrices obtained from the covariances in Fig. 5.16.



(a) Car (b) Background

Figure 5.19: Zoom in the left upper corner (110 elements) of whitening matrices.

Observing Fig. 5.18, there are correlations between closer cells, which are denoted by the weights on the diagonal and surrounding elements of the matrices. Moreover, zooming in (Fig. 5.19), the dark blue stripes down-weight the correlation between contrast sensitive and insensitive features, as already pointed out before. Besides, in relation to the disparity features, the highest weights are for the positions 31 and 34 and their harmonics, which correspond to the max and median disparity values on a HOG cell and seem to be more important than the descriptor positions 32 and 33 (min and mean), which have a lower positive weights in the whitening matrix.

After the analysis of the covariance and whitening matrices for the specific case of *C6feats*, Table 5.13 provides the car detection performance in average precision when whitening the features in the DPM training pipeline. In addition to *C6feats*, we report results on *C8B1feats*, which have been identified as the best 3D-aware feature in our experiments. Besides, it must be noted that the computation of the covariance matrix is carried out in a separate batch process before learning the models, particularly, obtained from the spatial correlation matrix Γ. Then, during the first training stage for root filter initialization, every covariance is built from Eq. 4.9 and its corresponding whitening matrix is calculated using Eq. 4.10. As a result, every model component have its associated whitening matrix. Next, in the remaining stages of the DPM learning process, every feature is whitened with Eq. 4.7.

Table 5.13: Car detection average precision in % with and without feature whitening

	C6feats			**C8B1feats**		
	Easy	Mod.	Hard	Easy	Mod.	Hard
No whitening	73.39	63.09	48.77	77.09	64.27	49.37
Whitening	70.27	59.50	45.75	73.48	60.62	45.05

In spite of the correlations and properties that were derived from the matrices displayed in previous figures, the application of whitening during learning have not improved the detection ratios. Besides, in general, the learning process has been also slower due to the addition of the whitening transformation, but also due to a longer number of iterations in the iterative parameter learning, which is a non-expected consequence of the feature transformation. Therefore, the feature whitening have not accomplished the starting hypothesis and motivation of this approach. For extending and further reviewing it, we propose to directly compute the covariance matrices

for every model component presenting a different root filter size, instead of using the pre-computed correlation matrix. This implies that the correlations will be searched in the full length of the 3D-aware descriptors ($h_c \times w_c \times d$), instead of the partial correlations between the individual descriptors of size d. Nevertheless, it is not clear whether this new research hypothesis would provide increased precision ratios. It is supposed to give faster convergence, though, during the stochastic gradient descent inside the learning module of DPM framework.

5.3.5.2. Stereo consistency check

In Chapter 4, we introduced an approach to increase the detection performance by a reduction of false positives (FP), based on the idea of deploying a married-matching between the bounding boxes predicted in left and right images. Table 5.14 display the results for the class 'car' in terms of the averaged AP and AOS figures in the three difficulty levels for one 5-fold cross-validation round. This table refers to the use of the left and right color and disparity images and, our contributed 3D-aware *C8B1feats*.

Table 5.14: Results for stereo consistency check employing *C8B1feats* for the class 'car'

	AP (%)			AOS (%)		
C8B1feats	easy	mod.	hard	easy	mod.	hard
single	85.29	68.91	53.18	66.36	54.26	41.83
stereo	85.72	68.96	53.21	68.05	55.26	42.53

As it can be observed in the table, the improvements are not relevant enough, showing slight increases which are more interesting for the orientation estimation (AOS). The reason behind this low increment is explained by high scoring FP, which are detected on both stereo views. Hence, they are also matched with our stereo consistency check and cannot be filtered out. Some examples are displayed in Fig. 5.20. In fact, among 30,000 to 100,000 candidate bounding boxes are detected by DPM per image, before the NMS. This means that our approach explores this large search space to effectively identify the matches and this is done by enforcing the epipolar geometry and height constraints in Section 4.4.2. Besides, this is implemented in C++ for efficiency. As a result, around 15 to 50 matching detections are found, as depicted in Fig. 5.20. Among them, the highest scoring TP are matched, but also some FP because they are detected on left and right images.

One might think that down-weighting or discarding not-matching candidates can remove wrong detections. However, from our experiments, we have checked that detections with lower scores have a higher uncertainty and may correspond to TP or FP. They are typically related to difficult cases: small, occluded, truncated or low contrasted objects, that are detected or not on left and/or right images. Then, we empirically observed very low AP when applying this down-weighting. Similar results were found if we change the average in Eq. 4.12 by a sum, because in this case, the matched false positives are upweighted, influencing the posterior NMS filtering.

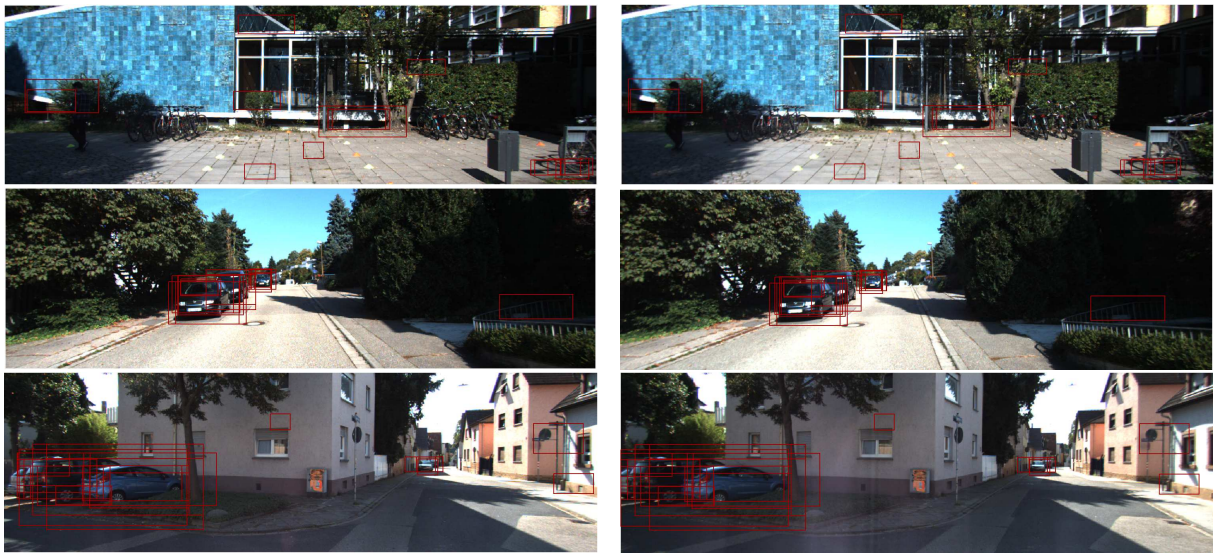


Figure 5.20: Examples of matched candidates on the left and right disparity images when employing our stereo consistency check. The bounding boxes are detections of cars when using the features *C8B1*. As it can be seen, there are true and false positives found in the married-matching. Consequently, there are slight increases in precision because several high-scoring false positives are detected on both views of the stereo and cannot be filtered out with our approach.

5.4. Discussion

In this section we introduce a discussion on four main topics that regards large datasets like KITTI and the high-dimensional models and machine learning techniques like in DPM framework. They are: **bias-variance tradeoff**, **model complexity**, **features** and **learning algorithms**. These aspects are interrelated, then we will comment on them in the next paragraphs.

Prediction models, such as the ones presented in this Thesis for scene layout inference or the object detection and orientation estimation, have a clear target: reducing the overall error of the system when applying a trained model on new unseen data. Considering the size of the KITTI dataset and the wide number of experiments presented in this Thesis, we will focus our analysis on object detection, because we already reported different reasons for the scene layout prediction performance in Chapter 3. Fig. 5.21 illustrates how the error is related to *bias*, *variance* and *model complexity*.

Ideally, we would like to find the most appropriate model complexity (in terms of object parts, features and filters dimensionality, mixture components, deformation parameters, etc.) that minimizes both bias and variance. The first one is related to underfitting, while the second one to overfitting. Hence, high bias is a symptom of a model with a low ability to approximate the data, whilst high variance is for a model that memorizes the training samples including the noise and other non-discriminative correlations. Estimating the main source of the error in our system, bias o variance, is not easy given the complexity of the DPM framework and the large intraclass variability of the KITTI objects. However, as already presented in the Thesis, we took special care of the error measurement to favor the assessment.

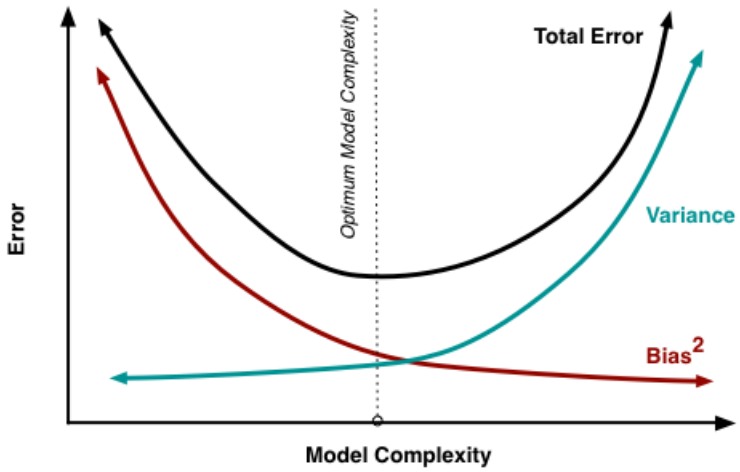


Figure 5.21: Representation of the bias-variance tradeoff that contributes to the total error and also related to the model complexity. This picture have been obtained from [Fortmann-Roe, 2012].

Firstly, the evaluation protocol has been clearly defined and 5-fold cross-validation has been carried out during training. Comparing the results from *LSVM-MDPM-sv* and our *DPM-C8B1* for the class 'car' in Table 5.7 and Fig. 5.7.d,e,f, it can be observed that the relative gap between precision values in test and in validation is higher for our *DPM-C8B1* approach. This could be an indication of possible high variance and a certain level of overfitting [Ng, 1997], in spite of our important improvement over the baseline LSVM-MDPM-sv (between 4-6% in AP). Increasing the number of training samples and/or reducing the features length could ideally help. In this regard, we already mirrored positive training samples (check Section 4.3) and also tested the shorter *C8B1* features. Another typical approach to increase the data samples is jittering the geometry of the training bounding boxes around the ground-truth locations [Keller et al., 2011]. However, this is implicitly included, up to some extent, in DPM framework: in training stage III (Algorithm 2), the ground-truth bounding boxes are treated as latent and surrounding image patches are also searched and scored to favor the learning process.

Differently, the error for pedestrians is high in both validation and test, which is related to high bias. Therefore, we point out as the possible reasons, 1) the adaptive parts, which may not correctly model this category and may be causing a model complexity reduction, and 2) a poor representativeness of the features for this class that may lead to low discriminative models. In this regard, another work already mentioned the benefits of dense stereo maps for pedestrian detection [Keller et al., 2011]. Additionally, reducing the number of mixture models from 8 to 4 could also help to increase pedestrian detection with the cost of reducing the discriminative power in orientation angles.

On the other hand, the class 'cyclist' yielded improved prediction for the easy samples with similar precision estimates in validation and test, such that there is no clear symptom, bias or variance. However, we have checked that the labeled cyclists in the KITTI dataset is limited in number and they are usually more challenging to predict than the object instances for the dominant 'car' category.

Apart from the bias-variance tradeoff, some other aspects influence the finally obtained prediction error. Indeed, we have demonstrated how the cleanliness of the data and the exploration of different setups during supervised learning can modify the behavior of the underlying training method. This have direct consequences on the learned parameters and prediction performance. Moreover, the algorithms and learning strategy must account for the data outliers with appropriate regularization and data samples treatment, i.e. the semiconvex approach of the Latent SVM and the hard negatives data mining proposed in DPM [Felzenszwalb et al., 2010b] and reviewed in this Thesis. Therefore, Machine Learning techniques play a fundamental role for object recognition tasks, where the established image processing methods can help to crop filtered measurements from images, but inferring semantic entities from real 3D scenes requires appropriate modeling of high-dimensional data.

With regard to the dilemma of "better features" or "better models", recently, some works in top vision conferences have already stated that developing better features could help, but the designed models and learning algorithms are key factors to leverage current successful descriptors, such as HOG [Benenson et al., 2013, Zhu et al., 2012]. Similarly, our experiments confirm these statements when analyzing the gains of our contributions. We have observed that increasing features dimensionality when adding disparity produced some precision increments depending on the object category. However, many decisions regarding the setup configuration during training (cleaning training samples, filters size initialization, overlap requirement, adaptive parts, regularization constant, etc.) also had an important effect in the obtained performance. Thus, low-level details matter when one tries to outperform current state-of-the-art approaches that yield good detection ratios in complex and naturalistic urban scenes.

Finally, more data helps until it saturates the model complexity or exceeds the capacities of the learning algorithms to obtain representative and discriminative models. In the end, a learning algorithm should beat our human intuition on what features are more relevant for each prediction tasks given a training dataset. Then, the magic of ML is to up-weight the parameters associated to the relevant features and down-weight the correlated or noisy ones. Therefore, assuming a desired output the training should conduct to the best modeling/fitting to the data. In this sense, future trends also point to the deep learning [Sermanet et al., 2013] and mid-level patch discovery [Maji and Shakhnarovich, 2013].

Chapter 6

Conclusions and Future Works

This Thesis has contributed to the state of the art in the area of 3D urban scene understanding [Geiger et al., 2014] accomplishing two tasks that are under active research in the academic and industrial sectors: I) scene layout inference with structured prediction methods, particularized for the problem of road intersections and, II) object detection and orientation estimation with a robust and well-known part-based object detector (DPM), for the categories 'car', 'pedestrian' and 'cyclist'.

The methodology of this Thesis relies on strong Machine Learning techniques for data mining high-dimensional visual features and it has been applied to real-world problems, i.e. the ones currently posed by autonomous robotics platforms, such as driverless vehicles [Google, 2011, MRG Oxford, 2012, Broggi et al., 2013]. Indeed, complex discriminative models have been defined in terms of Probabilistic Graphical Models and have been applied on stereo images obtained from a naturalistic urban dataset [KITTI, 2012].

6.1. Conclusions

In the next paragraphs we derive the specific conclusions and contributions from our supervised learning and inference of semantic information from road scene images. Firstly, we review below the conclusions when predicting the road intersections layout.

- Different parameterizations in terms of discrete random variables have been studied and proposed, which fully characterize the layouts for 1 straight road and 4 intersecting roads in Bird's eye Perspective images. These images represent 2D occupancy grid maps reconstructed from stereo sequences and have been obtained from [Geiger et al., 2011a].
- The proposed method is the first discriminative approach in the state of the art, devised as an alternative to the generative model in [Geiger et al., 2011a]. The problem has been formulated as a structured prediction, in which the potentials and loss functions have been designed in accordance to the proposed layout parameterizations. Besides, we have employed less cues from the sequences, i.e. only occupancy maps without flow vectors, which made the task more complicated for predicting the road boundaries.

- The supervised learning of the models has been carried out based on Conditional Random Fields, whilst inference has been tested with two different algorithms: an approximate inference approach based on distributed convex Belief Propagation [Schwing et al., 2011] and another exact inference approach based upon BB [Schwing and Urtasun, 2012].

- Given the small number of BeP images (113), we have carried out leave-one-out cross-validation experiments. For straight roads, the parameters have been successfully learned and the layout also predicted from real data, when employing both inference methods. However, several convergence problems have been observed for the 4 intersecting roads on synthetic images, due to the smoothness and shape of the proposed energy function. Particularly, when employing the dcBP engine. On the contrary, our contribution in the 4-roads case is the BB approach, which has yielded good predictions on synthetic images, and the study of how prediction degrades for an increasing level of noise on the synthetic samples.

- Finally, it must be remarked that the sparsity and noise of the real BeP images pose a very challenging prediction task, because they are even not visually recognizable by humans. Thus, the learning and inference algorithms were not able to capture the 4-roads layout with any of our proposed parameterizations. Therefore, additional data or more accurate occupancy maps are needed to estimate the layout of these intersections.

Next, the conclusions for the object detection and orientation estimation are summarized:

- The successful method known as Discriminative Part-based Models (DPM) [Felzenszwalb et al., 2010b] has been revisited because it is the baseline framework for the KITTI challenge [Geiger et al., 2012] on joint object detection and orientation estimation in urban environments, particularized for the categories 'car', 'pedestrian' and 'cyclist'.

- DPM has been adapted to account for 2.5D data (color and disparity). In fact, our research work is the first proposal using stereo data that have been published in the KITTI website [KITTI, 2012].

- New cues have been extracted from the stereo images, i.e. the contributed 3D-aware features that carry information from appearance and depth of objects parts in the scene. Moreover, several experiments have been conducted to select the most discriminative feature, which is *C8B1* and closely followed by *C8B2*.

- Besides, the DPM training pipeline has been modified and tuned with a 5-fold cross-validation to prevent overfitting. Additionally, a large set of comparative experiments have illustrated how to achieve better object models based on data mining concepts and internal parameter configurations.

- A "whitening" algorithm has been proposed for the equalization/normalization of the 3D-aware features with the aim of favoring the stochastic gradient descent optimization process when adjusting the model. However, the models trained with this strategy have not

yielded increased recall nor precision. This is mainly due to two reasons: 1) the different size of the filters for each object orientation that requires to extract submatrices from a precomputed spatial correlation matrix, which do not fully account for the correlations in the feature space and, 2) the lack of a bigger (order of a million) number of samples for the estimation of the covariance matrix particularized for every model component.

- A stereo-consistency check has been introduced in order to reduce the number of false positives of the object detector by applying epipolar geometry constraints. As a result, the experiments have shown that precision do not increase, suggesting that most of the detected bounding boxes (TP and FP) are predicted in both images and they are matched during the check.

- Several subtleties related to the KITTI evaluation protocol have been explained in detail and contrasted with experiments. Indeed, the evaluation metrics and algorithm have an important influence on the reported performance when comparing state-of-the-art methods. This Thesis has followed the guidelines from [KITTI, 2012] and also supported by a recent survey on pedestrian detection [Dollár et al., 2012].

- In addition to the large set of cross-validated experiments for the three object classes, we have published our results on the public ranking in KITTI website, as shown on the tables in the fifth chapter. Our *DPM-C8B1* approach ranked above the baseline method (MDPM-LSVM-sv), gaining a (3%-6%) increase in precision for cars detection. Besides, it also ranked first for cyclists. Then, this Thesis has contributed to push forward the current state of the art for the visual recognition of these object classes in naturalistic scenes.

- An unusual low detection performance has been obtained for pedestrians, which is related to a poor discriminative model that is learned when adding the adaptive parts. Besides, we have the guess that this issue is closely related to pedestrians aspect ratio and the addition of the disparity data, which may carry misleading information of the objects.

- Although the addition of 3D cues to DPM model could potentially help a better discrimination between the object parts of different viewpoints, our proposals (DPM-C8B1 and DPM-C8B2) yielded lower precision than the baseline MDPM-LSVM-sv. Therefore, we can conclude that object detection was improved at the cost of losing discriminative power in orientation.

Finally, having better features and measurements are great advantages for any Machine Learning or Computer Vision problem. However, the complexity of the proposed models can leverage the approaches when facing naturalistic scenes with hard challenges (illumination, occlusion, clutter, dynamic changes, ...). For instance, explicit occlusion modeling [Pepik et al., 2013] and object reasoning from monocular images [Fidler et al., 2012]. On other hand, the algorithms for learning and inference are also crucial in order to capture the variances and correlations in the data and to up-weight the discriminative features. Nevertheless, when employing contrasted methodologies for learning and inference, the "better features" vs "better models" dilemma is the recurrent topic for any previous and future research challenge. Indeed, we have observed that

increasing the dimensionality (and complexity at the same time) of the DPM object detector with our 3D-aware features, was beneficial to produce increased average precision, but taking special care of the number and the selection of the training samples. Differently, we have shown that robust machine learning methods, such as CRF and dcBP, can fail when employing well designed models (for scene layout prediction) but poor measured features.

6.2. Future Works

This section identifies some of the future works that can be derived from the experiments, contributions and conclusions of this Thesis. First of all, the focus has not been the real-time and most of the code is in Matlab[1]. However, for reducing processing times, several approaches could be addressed in the future, i.e. translating all the code to C++, parallelizing and optimizing the computation of the features and employing dedicated hardware as GPUs. Some of these extensions have been recently proposed as DPM applications: translation to C++ [Dubout and Fleuret, 2013] and GPU-based server with telecommunication facilities [Hirabayashi et al., 2013]. Besides, more complex approaches could be explored for algorithmic optimizations of the feature pyramid computation inside DPM [Dollár et al., 2014].

In relation to the features in our scene layout methodology, we have proposed a smart feature computation based on Integral Geometry, but further optimization can be studied, together with a more efficient implementation of the BB algorithm.

Furthermore, it must be mentioned that tracking algorithms can supplement the performance of the objects and layout predictors in applications with video sequences. Autonomous robotics platforms are characterized by their navigation in a dynamic environment and tracking methods in this context entail different issues that are closer to intelligent control systems than pure vision approaches, thus, tracking has been out of the scope of this research work, but we consider it as a natural follow-path. In fact, the temporal object trajectories known as 'tracklets' have been recently proposed and labeled in KITTI sequences [Geiger et al., 2014].

In addition, the probabilistic graphical models proposed for inferring the intersections layout can be extended and further studied for additional topologies, i.e. left-turn, right-turn, 3-roads intersections, etc. This would also require a larger dataset with more accurate disparity estimations and stereo reconstruction methods, which can be based on the recent scene flow approaches [Yamaguchi et al., 2013], in order to provide less sparse and less noisy occupancy grids.

On the other hand, for autonomous vehicles, pedestrian detection is a key challenge where we must revise our approach due to the low detection ratios observed in our experiments. Nevertheless, as can be also observed in [KITTI, 2012], there are many state-of-the-art methods that only focus on one object class in urban scenes, e.g. cars detection in KITTI. Similarly, pedestrian detection is a very specific topic, widely treated and under active research in the last years [Sotelo et al., 2006, Keller et al., 2011, Dollár et al., 2012]. It must be remarked that this

[1]Some parts of the code that are CPU-intensive are in C++. E.g. features computation at low level, stochastic gradient descent and branch and bound

Thesis has provided results for all the classes, but intensifying on cars and obtaining increased precision for two the categories cars and cyclists.

In relation to objects viewpoint prediction, although more accurate orientation estimation can be of interest, adding tracking capabilities and modeling of trajectories can substitute the need of the viewpoint labels. In fact, some of the approaches with best detection ratios for the class 'car' in [KITTI, 2012], report very poor results in orientation estimation, showing precision values below ours. Hence, is seems that the strategy of state-of-the-art works is seeking improved detection ratios at the cost of low discrimination in viewing angles.

For future works, we also propose to perform a joint object and scene layout prediction, such that both tasks can benefit to each other in order to obtain better detections and precision-recall curves. Specially, the objects and their tracklets can provide additional information for inferring the geometry of the roads and even the lanes. Similarly, the detection of the roads can bound the image search for objects and focus only on most promising areas. In close relation, the contextual priming is also of interest for the future of robotics perception, which means that more representative or discriminative features can be automatically learned when considering information from a semantic context and surrounding features or pixels. In our case, this can be investigated for the description of objects in the urban scenes given the naturalistic and high-resolution images of KITTI. In this regard, recent results on deep learning indicate that the generic descriptors extracted from the *Convolutional Neural Networks (CNN)* are very powerful for recognition tasks [Razavian et al., 2014]. Besides, it has been also demonstrated that the fixed size of HOG cells limits the training process, but flexible HOG descriptors of different sizes and shapes can be learned, obtaining outstanding results over previous object recognition methods [Benenson et al., 2013].

Finally, as already mentioned along this Thesis, autonomous vehicles and 3D urban scene understanding are hot topics with very recent and interesting approaches [Broggi et al., 2013, Geiger et al., 2014]. Moreover, the tendency towards onboard stereo systems [Vislab.it, 2014], which replaces the expensive Lidar technology, and the already mentioned stereo reconstruction methods based on scene flow, both will provide better disparity maps for 3D scene reasoning and autonomous navigation.

Bibliography

[Adobe, 2014] Adobe, 2014, 'Photoshop'. [web page] http://www.photoshop.com/.

[Alcantarilla et al., 2011] P. Alcantarilla, L. Bergasa, P. Jiménez, I. Parra, D. F. Llorca, M. A. Sotelo and S. S. Mayoral, 2011, 'Automatic LightBeam Controller for driver assistance'. *Machine Vision and Applications*, 22(5):819–835.

[Alcantarilla et al., 2012] P. F. Alcantarilla, J. Yebes, J. Almazán and L. M. Bergasa, 2012, 'On Combining Visual SLAM and Dense Scene Flow to Increase the Robustness of Localization and Mapping in Dynamic Envinronments'. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 1290–1297, St. Paul, Minnesota, USA.

[Asus, 2014] Asus, Last Viewed: March 2014, 'Xtion PRO'. [web page] http://www.asus.com/Multimedia/Xtion_PRO/.

[Autoliv, 2014] Autoliv, Last viewed: January 2014, 'Active Safety'. [web page] www.autoliv.com/ProductsAndInnovations/ActiveSafetySystems/Pages/RadarSystems.aspx.

[Badino et al., 2008] H. Badino, U. Franke and R. Mester, 2008, 'Free Space Computation Using Stochastic Occupancy Grids and Dynamic Programming'. In *Workshop on Dynamical Vision, ICCV*, 1–12, Rio de Janeiro, Brazil.

[Bao et al., 2012] S. Bao, Y. Xiang and S. Savarese, 2012, 'Object Co-detection'. In *Eur. Conf. on Computer Vision (ECCV)*, 86–101, Florence, Italy.

[Bay et al., 2008] H. Bay, A. Ess, T. Tuytelaars and L. V. Gool, 2008, 'Speeded-Up Robust Features (SURF)'. *Computer Vision and Image Understanding*, 110(3):346–359.

[Behley et al., 2013] J. Behley, V. Steinhage and A. B. Cremers, 2013, 'Laser-based segment classification using a mixture of bag-of-words'. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 4195–4200, Tokyo, Japan.

[Benenson et al., 2013] R. Benenson, M. Mathias, T. Tuytelaars and L. J. V. Gool, 2013, 'Seeking the strongest rigid detector'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 3666–3673, Portland, Oregon, USA.

[Bergasa et al., 2014] L. M. Bergasa, D. Almería, J. Almazán, J. J. Yebes and R. Arroyo, 2014, 'DriveSafe: an App for Alerting Inattentive Drivers and Scoring Driving Behaviors'. In *IEEE Intelligent Vehicles Symposium (IV)*, 1–6, Detroit, USA.

[Bishop, 2006] C. M. Bishop, 2006, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[Bleyer et al., 2012] M. Bleyer, C. Rhemann and C. Rother, 2012, 'Extracting 3D Scene-Consistent Object Proposals and Depth from Stereo Images'. In *Eur. Conf. on Computer Vision (ECCV)*, 467–481, Florence, Italy.

[BMW, 2014] BMW, Last viewed: January 2014, 'Intelligent vision'. [web page] www.bmw.com/com/en/insights/technology/connecteddrive/2013/driver_assistance/intelligent_vision.html.

[Braillon et al., 2008] C. Braillon, C. Pradalier, K. Usher, J. Crowley and C. Laugier, 2008, 'Occupancy Grids from Stereo and Optical Flow Data'. In *Experimental Robotics*, vol. 39 of *Springer Tracts in Advanced Robotics*, 367–376, Springer Verlag.

[Broggi et al., 2013] A. Broggi, M. Buzzoni, S. Debattisti, P. Grisleri, M. C. Laghi, P. Medici and P. Versari, 2013, 'Extensive Tests of Autonomous Driving Technologies'. *IEEE Trans. on Intelligent Transportation Systems*, 14(3):1403–1415.

[Buehler et al., 2009] M. Buehler, K. Iagnemma and S. Singh, 2009, *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*, vol. 56 of *Springer Tracts in Advanced Robotics*. Springer Verlag, George Air Force Base, Victorville, CA, USA.

[Dalal and Triggs, 2005] N. Dalal and B. Triggs, 2005, 'Histograms of Oriented Gradients for Human Detection'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 886–893.

[Danescu and Nedevschi, 2009] R. Danescu and S. Nedevschi, 2009, 'Probabilistic Lane Tracking in Difficult Road Scenarios Using Stereovision'. *IEEE Trans. on Intelligent Transportation Systems*, 10(2):272–282.

[DARPA RC, 2013] DARPA RC, 2013, 'The DARPA Robotics Challenge'. [web page] http://www.theroboticschallenge.org/.

[Daza et al., 2014] I. G. Daza, L. M. Bergasa, S. Bronte, J. J. Yebes, J. Almazán and R. Arroyo, 2014, 'Fusion of Optimized Indicators from Advanced Driver Assistance Systems (ADAS) for Driver Drowsiness Detection'. *Sensors*, 14(1):1106–1131.

[Dollár et al., 2014] P. Dollár, R. Appel, S. Belongie and P. Perona, 2014, 'Fast Feature Pyramids for Object Detection'. *IEEE Trans. Pattern Anal. Machine Intell.*, 99(PrePrints):1.

[Dollár et al., 2012] P. Dollár, C. Wojek, B. Schiele and P. Perona, 2012, 'Pedestrian Detection: An Evaluation of the State of the Art'. *IEEE Trans. Pattern Anal. Machine Intell.*, 34(4):743–761.

[Domke, 2010] J. Domke, 2010, 'Implicit Differentiation by Perturbation'. In *Advances in Neural Information Processing Systems*, 523–531.

[Dubout and Fleuret, 2013] C. Dubout and F. Fleuret, 2013, 'Accelerated Training of Linear Object Detectors'. In *IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 572–577.

[Duda et al., 2001] R. O. Duda, P. E. Hart and D. G. Stork, 2001, *Pattern Classification*. Wiley, New York, 2 edn.

[Erbs et al., 2013] F. Erbs, B. Schwarz and U. Franke, 2013, 'From stixels to objects - A conditional random field based approach'. In *IEEE Intelligent Vehicles Symposium (IV)*, 586–591, Gold Coast, Australia.

[Ess et al., 2008] A. Ess, B. Leibe, K. Schindler, and L. van Gool, 2008, 'A Mobile Vision System for Robust Multi-Person Tracking'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1–8, Anchorage, Alaska.

[Ess et al., 2009] A. Ess, T. Mueller, H. Grabner and L. J. V. Gool, 2009, 'Segmentation-Based Urban Traffic Scene Understanding'. In *British Machine Vision Conf. (BMVC)*, London, UK.

[European Commission, 2014] European Commission, 2014, 'Mobility and Transport: Road Safety'. [web page] http://ec.europa.eu/transport/road_safety/specialist/statistics/index_en.htm.

[Everingham et al., 2010] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn and A. Zisserman, 2010, 'The Pascal Visual Object Classes (VOC) Challenge'. *Intl. J. of Computer Vision*, 88(2):303–338.

[Fei-Fei et al., 2004] L. Fei-Fei, R. Fergus and P. Perona, 2004, 'Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). Workshop on Generative Model Based Vision*, vol. 12, 1–9, Los Alamitos, CA, USA.

[Felzenszwalb et al., 2010a] P. F. Felzenszwalb, R. B. Girshick and D. McAllester, 2010a, 'Discriminatively Trained Deformable Part Models, Release 4'. [web page] http://people.cs.uchicago.edu/~pff/latent-release4/.

[Felzenszwalb et al., 2010b] P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan, 2010b, 'Object Detection with Discriminatively Trained Part-Based Models'. *IEEE Trans. Pattern Anal. Machine Intell.*, 32(9):1627–1645.

[Fidler et al., 2012] S. Fidler, S. Dickinson and R. Urtasun, 2012, '3D Object Detection and Viewpoint Estimation with a Deformable 3D Cuboid Model'. In *Advances in Neural Information Processing Systems*, vol. 25, 620–628, Lake Tahoe, NE, USA.

[Fischler and Elschlager, 1973] M. A. Fischler and R. A. Elschlager, 1973, 'The Representation and Matching of Pictorial Structures'. *IEEE Transactions on Computers*, C-22(1):67–92.

[Fortmann-Roe, 2012] S. Fortmann-Roe, 2012, 'Understanding the Bias-Variance tradeoff'. [web page] http://scott.fortmann-roe.com/docs/BiasVariance.html/.

[Geiger, 2011] A. Geiger, 2011, 'Intersections dataset'. [web page] http://www.cvlibs.net/projects/intersection/.

[Geiger et al., 2011a] A. Geiger, M. Lauer and R. Urtasun, 2011a, 'A Generative Model for 3D Urban Scene Understanding from Movable Platforms'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1945–1952, Colorado Springs, USA.

[Geiger et al., 2014] A. Geiger, M. Lauer, C. Wojek, C. Stiller and R. Urtasun, 2014, '3D Traffic Scene Understanding from Movable Platforms'. *IEEE Trans. Pattern Anal. Machine Intell.*, 1–14.

[Geiger et al., 2012] A. Geiger, P. Lenz and R. Urtasun, 2012, 'Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 3354–3361, Providence, RI, USA.

[Geiger et al., 2010] A. Geiger, M. Roser and R. Urtasun, 2010, 'Efficient Large-Scale Stereo Matching'. In *Asian Conf. on Computer Vision (ACCV)*, 25–38, Queenstown, New Zealand.

[Geiger et al., 2011b] A. Geiger, C. Wojek and R. Urtasun, 2011b, 'Joint 3D Estimation of Objects and Scene Layout'. In *Advances in Neural Information Processing Systems*, vol. 24, 1467–1475, Granada, Spain.

[Gengenbach et al., 1995] V. Gengenbach, H. H. Nagel, F. Heimes, G. Struck and H. Kollnig, 1995, 'Model-based recognition of intersections and lane structures'. In *IEEE Intelligent Vehicles Symposium (IV)*, 512–517.

[Gharbi et al., 2012] M. Gharbi, T. Malisiewicz, S. Paris and F. Durand, 2012, 'A Gaussian Approximation of Feature Space for Fast Image Similarity'. Tech. Rep. TR-2012-032, MIT CSAIL, URL http://people.csail.mit.edu/tomasz/papers/gharbi_techreport_2012.pdf.

[González et al., 2013] A. González, L. M. Bergasa and J. J. Yebes, 2013, 'Text Detection and Recognition on Traffic Panels From Street-Level Imagery Using Visual Appearance'. *IEEE Trans. on Intelligent Transportation Systems*, PP(99):1–11.

[Google, 2011] Google, October 2011, 'Self-driving car'. [web page] http://spectrum.ieee.org/automaton/robotics/artificial-intelligence/how-google-self-driving-car-works.

[Gupta et al., 2010] A. Gupta, A. A. Efros and M. Hebert, 2010, 'Blocks World Revisited: Image Understanding Using Qualitative Geometry and Mechanics'. In *Eur. Conf. on Computer Vision (ECCV)*, 482–496, Heraklion, Crete, Greece.

[Gupta et al., 2011] A. Gupta, S. Satkin, A. A. Efros and M. Hebert, 2011, 'From 3D Scene Geometry to Human Workspace'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1961–1968, Washington, DC, USA.

[Hariharan et al., 2012] B. Hariharan, J. Malik and D. Ramanan, 2012, 'Discriminative Decorrelation for Clustering and Classification'. In *Eur. Conf. on Computer Vision (ECCV)*, 459–472, Florence, Italy.

[Hazan and Urtasun, 2010] T. Hazan and R. Urtasun, 2010, 'A Primal-Dual Message-Passing Algorithm for Approximated Large Scale Structured Prediction'. In *Advances in Neural Information Processing Systems*, vol. 23, 838–846, Vancouver, Canada.

[Hedau et al., 2010] V. Hedau, D. Hoiem and D. Forsyth, 2010, 'Thinking inside the box: using appearance models and context based on room geometry'. In *Eur. Conf. on Computer Vision (ECCV)*, 224–237, Heraklion, Crete, Greece.

[Hejrati and Ramanan, 2012] M. Hejrati and D. Ramanan, 2012, 'Analyzing 3D Objects in Cluttered Images'. In *Advances in Neural Information Processing Systems*, 602–610, Lake Tahoe, NE, USA.

[Helmer and Lowe, 2010] S. Helmer and D. G. Lowe, 2010, 'Using stereo for object recognition'. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 3121–3127, Anchorage, Alaska, USA.

[Hirabayashi et al., 2013] M. Hirabayashi, S. Kato, M. Edahiro, K. Takeda, T. Kawano and S. Mita, 2013, 'GPU implementations of object detection using HOG features and deformable models'. In *IEEE 1st International Conference on Cyber-Physical Systems, Networks, and Applications (CPSNA)*, 106–111, Taipei, Taiwan.

[Hirschmuller, 2008] H. Hirschmuller, 2008, 'Stereo Processing by Semiglobal Matching and Mutual Information'. *IEEE Trans. Pattern Anal. Machine Intell.*, 30(2):328–341.

[ICCV Workshop, 2013] ICCV Workshop, December 2013, 'Reconstruction Meets Recognition Challenge'. http://ttic.uchicago.edu/~rurtasun/rmrc/index.php.

[INRIA, 2012] INRIA, 2012, 'Visual Recognition and Machine Learning Summer School'. [web page] http://www.di.ens.fr/willow/events/cvml2012/.

[Jiang and Xiao, 2013] H. Jiang and J. Xiao, 2013, 'A Linear Approach to Matching Cuboids in RGBD Images'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2171–2178, Portland, OR, USA.

[Jiménez et al., 2012] D. Jiménez, D. Pizarro, M. Mazo and S. E. Palazuelos, 2012, 'Modelling and correction of multipath interference in time of flight cameras'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 893–900, Providence, RI, USA.

[Kammel et al., 2008] S. Kammel, J. Ziegler, B. Pitzer, M. Werling, T. Gindele, D. Jagzent, J. Schröder, M. Thuy, M. Goebl, F. von Hundelshausen, O. Pink, C. Frese and C. Stiller, 2008, 'Team AnnieWAY's autonomous system for the 2007 DARPA Urban Challenge'. *Journal of Field Robotics*, 25(9):615–639.

[Keller et al., 2011] C. G. Keller, M. Enzweiler, M. Rohrbach, D. F. Llorca, C. Schnorr and D. M. Gavrila, 2011, 'The Benefits of Dense Stereo for Pedestrian Detection'. *IEEE Trans. on Intelligent Transportation Systems*, 12(4):1096–1106.

[KITTI, 2012] KITTI, 2012, 'Object Detection and Orientation Estimation Benchmark'. [web page] http://www.cvlibs.net/datasets/kitti/eval_object.php.

[Kläser et al., 2008] A. Kläser, M. Marszalek and C. Schmid, 2008, 'A Spatio-Temporal Descriptor Based on 3D-Gradients'. In *British Machine Vision Conf. (BMVC)*, 1–10, Leeds, UK.

[Knuth, 1998] D. E. Knuth, 1998, *The Art of Computer Programming, Volume 3: (2Nd Ed.) Sorting and Searching*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA.

[Kokkinos, 2011] I. Kokkinos, 2011, 'Rapid Deformable Object Detection using Dual-Tree Branch-and-Bound'. In *Advances in Neural Information Processing Systems*, vol. 24, 2681–2689, Granada, Spain.

[Kokkinos, 2012a] I. Kokkinos, 2012a, 'Bounding Part Scores for Rapid Detection with Deformable Part Models'. In *2nd Parts and Attributes Workshop in Eur. Conf. on Computer Vision (ECCV)*, 41–50, Florence, Italy.

[Kokkinos, 2012b] I. Kokkinos, 2012b, 'Rapid Deformable Object Detection using Bounding-based Techniques'. Tech. Rep. RR-7940, INRIA, URL http://hal.inria.fr/hal-00696120.

[Koller and Friedman, 2009] D. Koller and N. Friedman, 2009, *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press.

[Krizhevsky, 2009] A. Krizhevsky, 2009, 'Learning multiple layers of features from tiny images'. Master's thesis, Department of Computer Science, University of Toronto.

[Lampert et al., 2009] C. H. Lampert, M. B. Blaschko and T. Hofmann, 2009, 'Efficient Subwindow Search: A Branch and Bound Framework for Object Localization'. *IEEE Trans. Pattern Anal. Machine Intell.*, 31:2129–2142.

[Lazebnik et al., 2006] S. Lazebnik, C. Schmid and J. Ponce, 2006, 'Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2169–2178, New York, USA.

[Lissel et al., 1994] E. Lissel, H. Rohling and W. Plagge, 1994, 'Radar sensor for car applications'. In *IEEE 44th Vehicular Technology Conference*, 438–442 vol.1.

[Lowe, 2004] D. Lowe, 2004, 'Distinctive Image Features from Scale-Invariant Keypoints'. *Intl. J. of Computer Vision*, 60(2):91–110.

[López-Sastre et al., 2011] R. J. López-Sastre, T. Tuytelaars and S. Savarese, 2011, 'Deformable part models revisited: A performance evaluation for object category pose estimation'. In *Intl. Conf. on Computer Vision (ICCV) Workshops*, 1052–1059, Barcelona, Spain.

[Maji and Shakhnarovich, 2013] S. Maji and G. Shakhnarovich, 2013, 'Part Discovery from Partial Correspondence'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 931–938, Portland, OR, USA.

[Makris et al., 2013] A. Makris, M. Perrollaz and C. Laugier, 2013, 'Probabilistic Integration of Intensity and Depth Information for Part-Based Vehicle Detection'. *IEEE Trans. on Intelligent Transportation Systems*, 14(4):1896–1906.

[Malisiewicz et al., 2011] T. Malisiewicz, A. Gupta and A. A. Efros, 2011, 'Ensemble of exemplar-SVMs for object detection and beyond'. In *Intl. Conf. on Computer Vision (ICCV)*, 89–96, Barcelona, Spain.

[Mattoccia, 2013] S. Mattoccia, 2013, 'Stereo Vision: Algorithms and Applications'. URL http://vision.deis.unibo.it/~smatt/Seminars/StereoVision.pdf.

[Matzka et al., 2012] S. Matzka, A. M. Wallace and Y. R. Petillot, 2012, 'Efficient resource allocation for attentive automotive vision systems'. *IEEE Trans. on Intelligent Transportation Systems*, 13(2):859–872.

[McCall and Trivedi, 2006] J. C. McCall and M. M. Trivedi, 2006, 'Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation'. *IEEE Trans. on Intelligent Transportation Systems*, 7(1):20–37.

[Microsoft, 2014] Microsoft, Last Viewed: March 2014, 'Kinect for Xbox 360'. [web page] http://www.xbox.com/en-US/KINECT.

[Milford and Wyeth, 2012] M. Milford and G. Wyeth, 2012, 'SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights'. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 1643–1649, St. Paul, MN, USA.

[Molinos et al., 2013] E. J. Molinos, Ángel Llamazares, N. Hernández, R. Arroyo, A. Cela, J. J. Yebes, M. Ocaña and L. M. Bergasa, 2013, 'Perception and Navigation in Unknown Environments: The DARPA Robotics Challenge'. In *ROBOT2013: First Iberian Robotics Conference*, vol. 253 of *Advances in Intelligent Systems and Computing*, 321–329.

[Montemerlo et al., 2008] M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. Hoffmann, B. Huhnke, D. Johnston, S. Klumpp, D. Langer, A. Levandowski, J. Levinson, J. Marcil, D. Orenstein, J. Paefgen, I. Penny, A. Petrovskaya, M. Pflueger, G. Stanek, D. Stavens, A. Vogt and S. Thrun, 2008, 'Junior: The Stanford Entry in the Urban Challenge'. *Journal of Field Robotics*, 25(9):569–597.

[MRG Oxford, 2012] MRG Oxford, 2012, 'RobotCar UK'. http://mrg.robots.ox.ac.uk/robotcar/index.html.

[Munaro and Menegatti, 2014] M. Munaro and E. Menegatti, 2014, 'Fast RGB-D people tracking for service robots'. *Autonomous Robots*, 1–16.

[Ng, 1997] A. Y. Ng, 1997, 'Preventing "Overfitting" of Cross-Validation Data'. In *Intl. Conf. on Machine Learning (ICML)*, 245–253, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[Nguyen et al., 2012] T.-N. Nguyen, B. Michaelis, A. Al-Hamadi, M. Tornow and M. Meinecke, 2012, 'Stereo-Camera-Based Urban Environment Perception Using Occupancy Grid and Object Tracking'. *IEEE Trans. on Intelligent Transportation Systems*, 13(1):154–165.

[Nowozin and Lampert, 2011] S. Nowozin and C. H. Lampert, 2011, 'Structured Learning and Prediction in Computer Vision'. *Foundations and Trends in Computer Graphics and Vision*, 6(3-4):185–365.

[Oliva and Torralba, 2001] A. Oliva and A. Torralba, 2001, 'Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope'. *Intl. J. of Computer Vision*, 42(3):145–175.

[Ozuysal et al., 2009] M. Ozuysal, V. Lepetit and P.Fua, 2009, 'Pose Estimation for Category Specific Multiview Object Localization'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Miami, FL, USA.

[Park et al., 2010] D. Park, D. Ramanan and C. Fowlkes, 2010, 'Multiresolution Models for Object Detection'. In *Eur. Conf. on Computer Vision (ECCV)*, 241–254, Heraklion, Crete, Greece.

[PASCAL VOC, 2012] PASCAL VOC, 2012, 'The Pattern Analysis, Statistical modeling and Computational Learning Visual Object Classes Homepage'. [web page] http://pascallin.ecs.soton.ac.uk/challenges/VOC/.

[Pepik et al., 2012a] B. Pepik, P. Gehler, M. Stark and B. Schiele, 2012a, '3D2PM - 3D Deformable Part Models'. In *Eur. Conf. on Computer Vision (ECCV)*, 356–370, Florence, Italy.

[Pepik et al., 2012b] B. Pepik, M. Stark, P. Gehler and B. Schiele, 2012b, 'Teaching 3D Geometry to Deformable Part Models'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 3362–3369, Washington, DC, USA.

[Pepik et al., 2013] B. Pepik, M. Stark, P. Gehler and B. Schiele, 2013, 'Occlusion Patterns for Object Class Detection'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 3286–3293, Portland, OR, USA.

[Rasmussen, 2003] C. Rasmussen, 2003, 'Road shape classification for detecting and negotiating intersections'. In *IEEE Intelligent Vehicles Symposium (IV)*, 422–427.

[Razavian et al., 2014] A. S. Razavian, H. Azizpour, J. Sullivan and S. Carlsson, 2014, 'CNN Features off-the-shelf: an Astounding Baseline for Recognition'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Colombus,USA.

[Ren and Ramanan, 2013] X. Ren and D. Ramanan, 2013, 'Histograms of Sparse Codes for Object Detection'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 3246–3253, Portland, OR, USA.

[Rodríguez et al., 2012] A. Rodríguez, J. J. Yebes, P. F. Alcantarilla, L. M. Bergasa, J. Almazán and A. Cela, 2012, 'Assisting the Visually Impaired: Obstacle Detection and Warning System by Acoustic Feedback'. *Sensors*, 12(12):17476–17496.

[Rohrbach et al., 2009] M. Rohrbach, M. Enzweiler and D. M. Gavrila, 2009, 'High-level fusion of depth and intensity for pedestrian classification'. In *In Proc. of DAGM*, 101–110, Jena, Germany.

[Roig et al., 2011] G. Roig, X. Boix, H. B. Shitrit and P. Fua, 2011, 'Conditional Random Fields for multi-camera object detection'. In *Intl. Conf. on Computer Vision (ICCV)*, 563–570, Barcelona, Spain.

[Scharstein and Szeliski, 2001] D. Scharstein and R. Szeliski, 2001, 'A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms'. *Intl. J. of Computer Vision*, 47(1/2/3):7–42.

[Schwing et al., 2011] A. Schwing, T. Hazan, M. Pollefeys and R. Urtasun, 2011, 'Distributed Message Passing for Large Scale Graphical Models'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1833–1840, Colorado Springs, USA.

[Schwing et al., 2012] A. G. Schwing, T. Hazan, M. Pollefeys and R. Urtasun, 2012, 'Efficient Structured Prediction for 3D Indoor Scene Understanding'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2815–2822, Providence, RI, USA.

[Schwing and Urtasun, 2012] A. G. Schwing and R. Urtasun, 2012, 'Efficient exact inference for 3D indoor scene understanding'. In *Eur. Conf. on Computer Vision (ECCV)*, 299–313, Florence, Italy.

[Sermanet et al., 2013] P. Sermanet, K. Kavukcuoglu, S. Chintala and Y. LeCun, 2013, 'Pedestrian Detection with Unsupervised Multi-stage Feature Learning'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 3626–3633, Portland, OR, USA.

[Singh and Kosecká, 2012] G. Singh and J. Kosecká, 2012, 'Acquiring semantics induced topology in urban environments'. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 3509–3514, St. Paul, MN, USA.

[Singh et al., 2012] S. Singh, A. Gupta and A. A. Efros, 2012, 'Unsupervised Discovery of Mid-Level Discriminative Patches'. In *Eur. Conf. on Computer Vision (ECCV)*, 73–86, Florence, Italy.

[Sivaraman and Trivedi, 2013] S. Sivaraman and M. M. Trivedi, 2013, 'A review of recent developments in vision-based vehicle detection'. In *IEEE Intelligent Vehicles Symposium (IV)*, 310–315, Gold Coast, Australia.

[Sotelo et al., 2006] M. A. Sotelo, I. Parra, D. Fernandez and E. Naranjo, 2006, 'Pedestrian Detection Using SVM and Multi-Feature Combination'. In *IEEE Intelligent Transportations Systems Conference (ITSC)*, 103–108, Toronto, CA.

[Sukkarieh et al., 1999] S. Sukkarieh, E. M. Nebot and H. F. Durrant-Whyte, 1999, 'A high integrity IMU/GPS navigation loop for autonomous land vehicle applications'. *IEEE Trans. Robot. Automat.*, 15(3):572–578.

[Thrun, 2003] S. Thrun, 2003, 'Learning Occupancy Grid Maps with Forward Sensor Models'. *Autonomous Robots*, 15(2):111–127.

[Torralba et al., 2003] A. Torralba, K. Murphy, W. Freeman and M. Rubin, 2003, 'Context-based vision system for place and object recognition'. In *Intl. Conf. on Computer Vision (ICCV)*, 273–280 vol.1, Madison, WI, USA.

[Toyota, 2014] Toyota, Last viewed: January 2014, 'Intelligent Parking System'. [web page] www.toyota-global.com/innovation/safety_technology/safety_technology/parking/.

[Tretyak et al., 2012] E. Tretyak, O. Barinova, P. Kohli and V. Lempitsky, 2012, 'Geometric Image Parsing in Man-Made Environments'. *Intl. J. of Computer Vision*, 97(3):305–321.

[UNECE, 2010] UNECE, 2010, 'Number of pedestrians killed in road traffic accidents'. [web page] http://w3.unece.org/pxweb/quickstatistics/readtable.asp?qs_id=59.

[Urmson et al., 2007] C. Urmson, J. Anhalt, J. A. D. Bagnell, C. R. Baker, R. E. Bittner, J. M. Dolan, D. Duggins, D. Ferguson, T. Galatali, H. Geyer, M. Gittleman, S. Harbaugh, M. Hebert, T. Howard, A. Kelly, D. Kohanbash, M. Likhachev, N. Miller, K. Peterson, R. Rajkumar, P. Rybski, B. Salesky, S. Scherer, Y.-W. Seo, R. Simmons, S. Singh, J. M. Snider, A. T. Stentz, W. R. L. Whittaker and J. Ziglar, 2007, 'Tartan Racing: A Multi-Modal Approach to the DARPA Urban Challenge'. Tech. Rep. CMU-RI-TR-, Robotics Institute, Pittsburgh, PA.

[Vedaldi et al., 2009] A. Vedaldi, V. Gulshan, M. Varma and A. Zisserman, 2009, 'Multiple kernels for object detection'. In *Intl. Conf. on Computer Vision (ICCV)*, 606–613.

[Velodyne, 2014] Velodyne, Last Viewed: March 2014, 'HDL 64 lidar'. [web page] http://velodynelidar.com/lidar/hdlproducts/hdl64e.aspx.

[Viola and Jones, 2001a] P. Viola and M. Jones, 2001a, 'Fast and robust classification using asymmetric AdaBoost and a detector cascade'. In *Advances in Neural Information Processing Systems*, 1311–1318, MIT Press.

[Viola and Jones, 2001b] P. Viola and M. Jones, 2001b, 'Rapid object detection using a boosted cascade of simple features'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, I–511–I–518 vol.1, Kauai, HI, USA.

[Vislab.it, 2014] Vislab.it, 2014, '3DV Stereo System'. [web page] http://vislab.it/products/3dv-stereo-system/.

[Walk et al., 2010] S. Walk, K. Schindler and B. Schiele, 2010, 'Disparity statistics for pedestrian detection: Combining appearance, motion and stereo'. In *Eur. Conf. on Computer Vision (ECCV)*, 182–195, Heraklion, Crete, Greece.

[Wang et al., 2010] H. Wang, S. Gould and D. Koller, 2010, 'Discriminative learning with latent variables for cluttered indoor scene understanding'. In *Eur. Conf. on Computer Vision (ECCV)*, 497–510, Heraklion, Crete, Greece.

[Wang et al., 2013] T. Wang, X. He and N. Barnes, 2013, 'Learning Structured Hough Voting for Joint Object Detection and Occlusion Reasoning'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1790–1797, Los Alamitos, CA, USA.

[Wojek and Schiele, 2008] C. Wojek and B. Schiele, 2008, 'A Dynamic Conditional Random Field Model for Joint Labeling of Object and Scene Classes'. In *Eur. Conf. on Computer Vision (ECCV)*, 733–747, Marseille, France.

[Wojek et al., 2013] C. Wojek, S. Walk, S. Roth, K. Schindler and B. Schiele, 2013, 'Monocular Visual Scene Understanding: Understanding Multi-Object Traffic Scenes'. *IEEE Trans. Pattern Anal. Machine Intell.*, 35(4):882–897.

[Yamaguchi et al., 2013] K. Yamaguchi, D. McAllester and R. Urtasun, 2013, 'Robust Monocular Epipolar Flow Estimation'. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1862–1869, Portland, Oregon, USA.

[Yebes et al., 2011] J. Yebes, P. F. Alcantarilla and L. M. Bergasa, 2011, 'Occupant Monitoring System for Traffic Control Based on Visual Categorization'. In *IEEE Intelligent Vehicles Symposium (IV)*, 212–217, Baden-Baden,Germany.

[Yebes et al., 2014] J. Yebes, L. M. Bergasa, R. Arroyo and A. Lázaro, 2014, 'Supervised learning and evaluation of KITTI's cars detector with DPM'. In *IEEE Intelligent Vehicles Symposium (IV)*, 1–6, Detroit, USA.

[Zhu et al., 2012] X. Zhu, C. Vondrick, D. Ramanan and C. C. Fowlkes, 2012, 'Do we need more training data or better models for object detection?' In *British Machine Vision Conf. (BMVC)*, 1–11, British Machine Vision Association, Surrey, UK.

Appendix A

Additional approaches for inferring 4-roads layout.

Chapter 3 presented the theory and concepts for learning and inferring the layout of intersections from Bird's eye Perspective images. More specifically, this Thesis tackles the problem as a structured prediction, in which the conditional probability $p(y|x, \boldsymbol{\theta})$ is modeled by defining a factorization that decomposes the joint reasoning of all discrete random variables y_i into low order potentials.

The approach presented in Section 3.3, to infer the layout of 4-road intersections, has been effectively demonstrated on synthetic images (Section 3.4) after a long research on the most convenient and also feasible model. Indeed, inferring the scene layout from the difficult BeP images (check Fig. 3.7) [Geiger et al., 2011a] is a challenging task. Therefore, we have explored different configurations that were initially tested with the message-passing inference algorithm in [Schwing et al., 2011], but yielded convergence problems on synthetic images as summarized in Section 3.4. Three approaches for learning and inferring the geometry of 4-roads intersections are included in this appendix in order to introduce a more detailed description of the research carried out.

A.1. 4-roads layout: Approach 1.

From the very beginning, our inspiration was on the room layout decomposition by [Schwing et al., 2012] to achieve a smart factorization of the conditional probability to be modeled using CRFs. As already introduced in Section 3.3, the BeP image is divided in 5 regions of interest. However, the first parameterization approach is shown in Fig. A.1.

Every region of interest is described by one intersecting point $I_r = (u_r, v_r)$ and two angles β_i and β_j. The points are referenced to the lower left corner of the image. The angles are measured with respect to a local coordinate system aligned with the image borders and centered at the street lines intersection point (in green color). As a result, there are 12 discrete random variables: $[\beta_1, \beta_2, \beta_3, \beta_4, u_A, v_A, u_B, v_B, u_C, v_C, u_D, v_D]$.

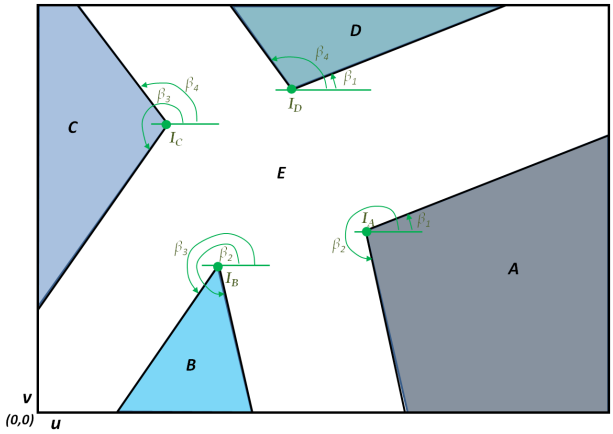


Figure A.1: First parameterization of 4-roads layout.

Considering this parameterization, the proposed factorization is detailed below:

$$log\psi(\mathbf{y},\mathbf{x}) = \boldsymbol{\theta}_E^T \cdot \phi_E(\mathbf{x},\mathbf{u}_E,\mathbf{v}_E,\boldsymbol{\beta}_E) + \sum_{r\in\Upsilon} \boldsymbol{\theta} r^T \cdot \phi_r(\mathbf{x},u_r,v_r,\beta_i,\beta_j) + \sum_{c\in\Psi} \theta_c \cdot \phi_c(y_{c1},y_{c2}) \quad \text{(A.1)}$$

- The vectors $\mathbf{u}_E, \mathbf{v}_E$ and $\boldsymbol{\beta_E}$ represent all the random variables of each type in our model
- $r \in \Upsilon = \{A, B, C, D\}$
- $i \neq j$ such that $i = \{1, ...4\}$ and $j = \{1, ...4\}$
- $c \in \Psi =$ Set of constraints
- y_{c1} and y_{c2} are the variables for the pairwise potentials that encode the constraints. They will be defined later in this section

Furthermore, considering the Integral Geometry approach introduced in subsection 3.3.2, every image feature potential in Eq. A.1 can be decomposed as a linear combination of lower order potentials. In particular, the following decomposition into third order functions:

$$\phi_r(\mathbf{x},u_r,v_r,\beta_i,\beta_j) = H_r(\mathbf{x},u_r,v_r,\beta_i) - H_r(\mathbf{x},u_r,v_r,\beta_j) + G(\beta_i,\beta_j) \quad \text{(A.2)}$$

$$\phi_E(\mathbf{x},\mathbf{u}_E,\mathbf{v}_E,\boldsymbol{\beta}_E) = \phi(\mathbf{x}) - \sum_{r\in\Upsilon} \phi_r(\mathbf{x},u_r,v_r,\beta_i,\beta_j) \quad \text{(A.3)}$$

The function $H_r(...)$ is an accumulator that counts the visual features (2D grids in the BeP) in a hypothesis region of the image determined by the random variables in its argument. Besides, $G(...)$ adjusts the computation of visual features depending on the angles. Finally, the term $\phi(\mathbf{x})$ represents the visual features (occupancy grids) in the whole image. For clarification, we are employing the same features defined in Section 3.3, i.e. the 2D vector $[O, F]$ normalized by the total number of grids of each type.

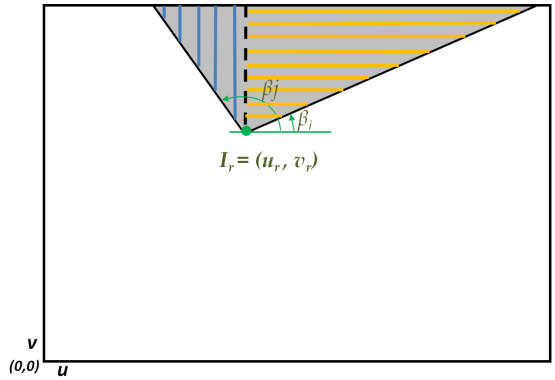
Let us review the mathematical expression for the functions H_r and G defined in Eq A.2 and their physical meaning with some examples.

$$H_r(\mathbf{x},u_r,v_r,\beta_k) = \begin{cases} H_r^{left}(\mathbf{x},u_r,v_r,\beta_k) & \text{if } 0 \leq \beta_k \leq \pi/2 \\ -H_r^{right}(\mathbf{x},u_r,v_r,\beta_k) & \text{if } \pi/2 < \beta_k < 2\pi \end{cases} \quad \text{(A.4)}$$

In the expression above, H_r^{left} counts the visual features in the region delimited by two lines. The first line is given by u_r, v_r and β_k, and the second one is the perpendicular line to the top image border that passes through the point $I_r = (u_r, v_r)$. This function is evaluated towards the *left* direction from the first line. Besides, H_r^{right} counts the visual features in the region delimited by the same two lines, but evaluated towards the *right* direction from the first line. Next, the function G is defined in Eq A.5. The value 1 represents the total number of normalized occupancy grids of one type (for example: *O=occupied* or *F=free/road*). For illustration, we include four graphical representations in Table A.1, which shows the decomposition of every proposed region of interest. As it can be seen, we reduce from fourth to third order potentials.

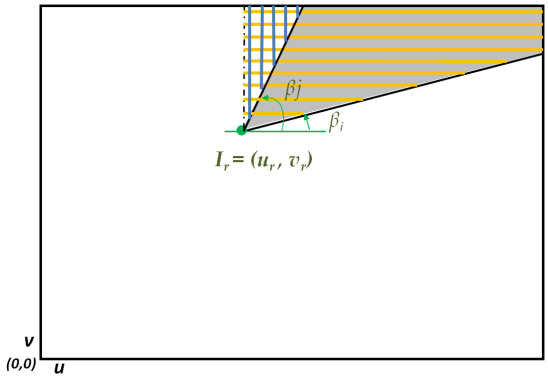
$$G(\beta_i, \beta_j) = \begin{cases} 0 & \text{if } \beta_i \leq \beta_j \\ 1 & \text{if } \beta_i > \beta_j \end{cases} \tag{A.5}$$

Table A.1: Geometric decomposition to count the visual features on every region (grey color). Given the factorization in Eq.A.1, we can divide the potentials into the third order and pairwise functions H_r and G of Eq. A.2.



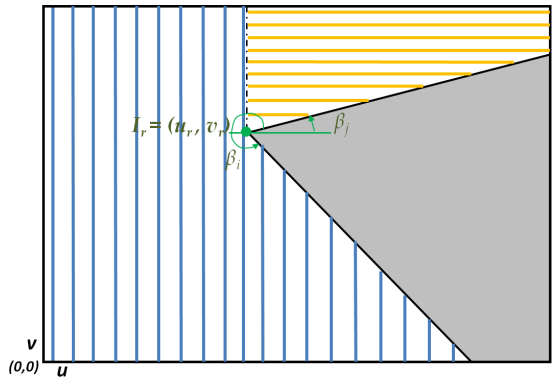
$H_r^{left}(\mathbf{x}, u_r, v_r, \beta_i) - (-H_r^{right}(\mathbf{x}, u_r, v_r, \beta_j))$
$+G(\beta_i, \beta_j)$

where $G(\beta_i, \beta_j) = 0$



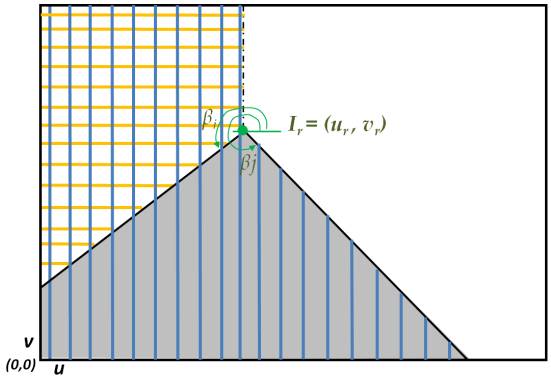
$H_r^{left}(\mathbf{x}, u_r, v_r, \beta_i) - H_r^{left}(\mathbf{x}, u_r, v_r, \beta_j)$
$+G(\beta_i, \beta_j)$

where $G(\beta_i, \beta_j) = 0$



$-H_r^{right}(\mathbf{x}, u_r, v_r, \beta_i) - H_r^{left}(\mathbf{x}, u_r, v_r, \beta_j)$
$+G(\beta_i, \beta_j)$

where $G(\beta_i, \beta_j) = 1$



$-H_r^{right}(\mathbf{x}, u_r, v_r, \beta_i) - (-H_r^{right}(\mathbf{x}, u_r, v_r, \beta_j)$
$+G(\beta_i, \beta_j))$

where $G(\beta_i, \beta_j) = 0$

Next, we define the set of constraints $\phi_c(y_{c1}, y_{c2})$ for every region to impose some restrictions in the relative position of the intersecting points (I_A, I_B, I_C, I_D). In fact, we impose tight constraints on their adjacent position, but allow some freedom between opposite points like I_A and I_C, I_B and I_D. All the proposed constraints are pairwise potentials that depend on u and v image coordinates.

$$\phi_{AB}(u_A, u_B) = \begin{cases} 0 & \text{if } u_A > u_B \\ -\infty & \text{other} \end{cases} \qquad \phi_{DC}(u_D, u_C) = \begin{cases} 0 & \text{if } u_D > u_C \\ -\infty & \text{other} \end{cases}$$

$$\phi_{DA}(v_D, v_A) = \begin{cases} 0 & \text{if } v_D > v_A \\ -\infty & \text{other} \end{cases} \qquad \phi_{CB}(v_C, v_B) = \begin{cases} 0 & \text{if } v_C > v_B \\ -\infty & \text{other} \end{cases}$$

$$\phi_{DB}(v_D, v_B) = \begin{cases} 0 & \text{if } v_D > v_B \\ -\infty & \text{other} \end{cases}$$

The complete graphical model that describes the relationships between the proposed parameterization and corresponding factors is depicted in Fig. A.2.

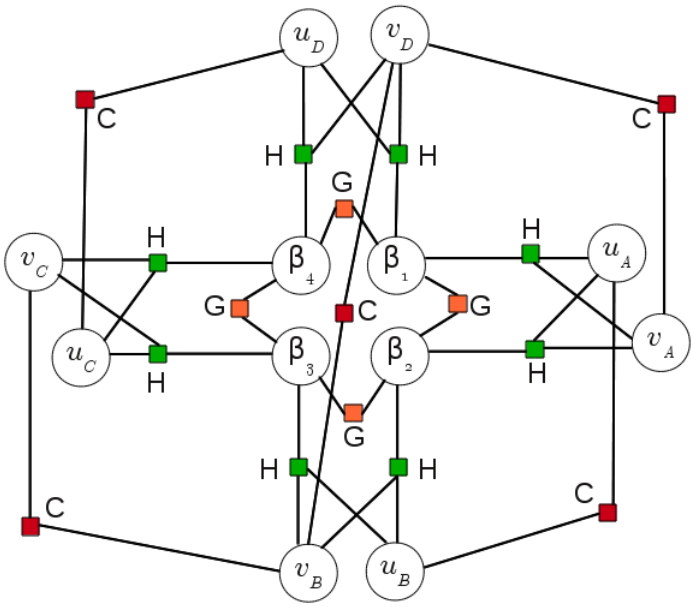


Figure A.2: Factor graph for inferring the layout of 4-roads intersections 4-way. Approach 1.

Until this point we have defined the structured prediction potentials of the approach 1. For learning, we extend the same decomposition to compute the loss function, but removing the constraint factors of the inference task. However, the loss factors account for the pixel-wise error in every region hypothesis previously described by Fig. A.1. Then, our loss function for this approach is exposed in Eq. A.6, where the vector $\mathbf{y}$ denotes the ground-truth labels for every discrete random variable y_i. They are obtained from the training dataset with image samples $\mathbf{x}$. Besides, $\hat{\mathbf{y}}$ refers to the predicted labels during learning.

$$\Delta(\mathbf{y}, \hat{\mathbf{y}}) = \phi_E^{loss}(\mathbf{x}, \mathbf{y}, \mathbf{u}_E, \mathbf{v}_E, \boldsymbol{\beta}_E) + \sum_{r \in \Upsilon} \phi_r^{loss}(\mathbf{x}, \mathbf{y}, u_r, v_r, \beta_i, \beta_j) \tag{A.6}$$

A.1.1. Discussion on experiments with approach 1.

After several experiments on synthetic BeP images (see Fig. 3.15), we observed these issues:

- During learning, the proposed potentials in Eq. A.1, plus the constraints and the defined loss do not seem to capture the intersection layout in synthetic images. Consequently, we observed swapped signs for some of the values of the learned parameter vector $\boldsymbol{\theta}$. It must be noted that ideally, this vector should be $\boldsymbol{\theta} = [+\theta^O_A, -\theta^F_A, +\theta^O_B, -\theta^F_B, +\theta^O_C, -\theta^F_C, +\theta^O_D, -\theta^F_D, -\theta^O_E, +\theta^F_E, +\theta_{AB}, +\theta_{DC}, +\theta_{DA}, +\theta_{CB}, +\theta_{DB}]$, which corresponds to maximizing occupied grids (O) in A,B,C and D regions and maximizing free grids (F) in E model zone.
- During inference, for some cases, we observed convergence problems, i.e. the dcBP algorithm was not able to find the maximum energy after a high number of iterations, and for other cases, we got suboptimal solutions, i.e. the algorithm converges but the predicted labels do not match the expected solution.

Therefore, we reconsidered the definition of the potentials and constraints to deal with some possible ties that we detected as the main source of the problems above. This means that in our first approach, some hypotheses of the search space could have exactly the same energy (Eq A.1) and, could also make the energy function not monotonic, affecting the underlying learning and inference algorithms. The revised approach is presented in the next section.

A.2. 4-roads layout. Approach 2.

According to the previously observed issues, some modifications are introduced for the computation of the model potentials and constraints. In this second approach, we maintain the parameterization in Fig. A.1 and the general factorization of Eq. A.1. However, the decomposition of the potentials is redefined as follows, which presents a more clear notation and it is easier to generalize to all the model regions. This also facilitates the implementation for debugging purposes.

$$\phi_r(\mathbf{x}, u_r, v_r, \beta_i, \beta_j) = H_r(\mathbf{x}, u_r, v_r, \beta_i) + G_r(\mathbf{x}, u_r, v_r, \beta_j) \tag{A.7}$$

$$\phi_E(\mathbf{x}, \mathbf{u}_E, \mathbf{v}_E, \boldsymbol{\beta}_k) = \phi(\mathbf{x}) - \sum_{r \in \Upsilon} \theta_r^T \cdot \phi_r(\mathbf{x}, u_r, v_r, \beta_i, \beta_j) \tag{A.8}$$

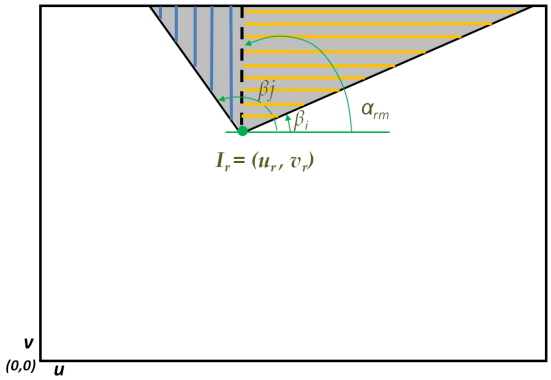
$$H_r(\mathbf{x}, u_r, v_r, \beta_i) = H_r^{left}(\mathbf{x}, u_r, v_r, \beta_i) \tag{A.9}$$

$$G_r(\mathbf{x}, u_r, v_r, \beta_j) = \begin{cases} -H_r^{left}(\mathbf{x}, u_r, v_r, \beta_j) & \text{if } \alpha_{rl} \leq \beta_j < \alpha_{rm} \\ H_r^{right}(\mathbf{x}, u_r, v_r, \beta_j) & \text{if } \alpha_{rm} \leq \beta_j \leq \alpha_{ru} \end{cases} \tag{A.10}$$

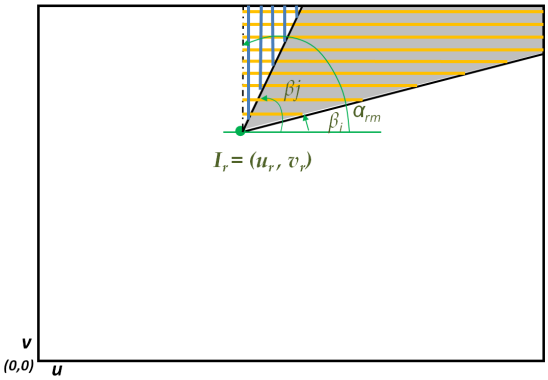
From the equations above, α_{rl}, α_{rm} and α_{ru} are lower, medium and upper limits for the angles β_k in our model. Their values depend on each region r. More specifically, the tuples $b_r = [\alpha_{rl}$, α_{rm} and $\alpha_{ru}]$ for every region are: $b_A = [-90, 0, 90]$, $b_B = [-180, -90, 0]$, $b_C = [-90, 180, 90]$ and $b_D = [0, 90, 180]$. These values have been empirically determined from the training dataset.

On the one hand, H_r^{left} counts the visual features in the region delimited by two lines. The first line is given by u_r, v_r and β_k, and the second one is the perpendicular line to the corresponding image border. This line forms an angle of α_{rm} with respect to an horizontal line that passes through the point $I_r = (u_r, v_r)$. This function is evaluated towards the *left* direction (anticlockwise) from the first line. On the other hand, H_r^{right} counts the visual features in the region delimited by the same two lines defined before, but now the function is evaluated towards the *right* direction (clockwise) from the first line. An illustration of this decomposition is displayed in Table A.2, which exposes two sample cases for region D (check Fig. A.1).

Table A.2: Geometric decomposition to count the visual features on every region (grey color) for the second approach.



$$H_r(\mathbf{x}, u_r, v_r, \beta_i) + G_r(\mathbf{x}, u_r, v_r, \beta_j) =$$
$$= H_r^{left}(\mathbf{x}, u_r, v_r, \beta_i) + H_r^{right}(\mathbf{x}, u_r, v_r, \beta_j)$$



$$H_r(\mathbf{x}, u_r, v_r, \beta_i) + G_r(\mathbf{x}, u_r, v_r, \beta_j) =$$
$$= H_r^{left}(\mathbf{x}, u_r, v_r, \beta_i) - H_r^{left}(\mathbf{x}, u_r, v_r, \beta_j)$$

As it can be seen in the figures, to count the visual features in the grey patch of interest, we divide the problem in two halves, which are the factors H and G. The splitting line segment is defined by α_{rm}, which divides the image patch in two subregions. This angle equals 90 degrees for the displayed example that corresponds to region D. For the remaining regions A, B and C, the same evaluation can be carried out, but previously rotating 90 degrees anticlockwise the images in Table A.2.

Moreover, in addition to the constraints in Section A.1, we enforce tighter physical constraints between the angles to guide the convergence during inference.

$$\phi_1(\beta_1, \beta_4) = \begin{cases} 0 & \text{if } \beta_4 \geq \beta_1 \\ -\infty & \text{other} \end{cases} \qquad \phi_3(\beta_2, \beta_3) = \begin{cases} 0 & \text{if } \beta_3 \geq \beta_2 \text{ and } \beta_3 > 0 \\ 0 & \text{if } \beta_2 \geq \beta_3 \text{ and } \beta_3 < 0 \\ -\infty & \text{other} \end{cases}$$

$$\phi_2(\beta_1, \beta_2) = \begin{cases} 0 & \text{if } \beta_1 \geq \beta_2 \\ -\infty & \text{other} \end{cases} \qquad \phi_4(\beta_3, \beta_4) = \begin{cases} 0 & \text{if } \beta_3 \geq \beta_4 \text{ and } \beta_3 > 0 \\ 0 & \text{if } \beta_4 \geq \beta_3 \text{ and } \beta_3 < 0 \\ -\infty & \text{other} \end{cases}$$

A.2.1. Discussion on experiments with approach 2.

Although this second approach is an smart and elegant way for decomposing the problem, the experiments yielded poor prediction performance on synthetic images, as already observed with the approach 1. After a deep analysis and the evaluation of the energy shape of the "Boltzmann distribution" (go to Eq. 3.1), we found a potential overlap from the sharing of angles. The nature of the problem comes from the assumption that a street has two parallel lines (same β_k angle), which belong to different polygons (regions A, B, C or D). The situation is depicted in Fig. A.3. Given the same point (u_r, v_r) and considering the parameterization in Fig. A.1, the grey region hypothesis could be delimited by two different pairs of angles β_i and β_j. Indeed, depending on whether we take region C or region D as reference, the values of the *betas* are different.

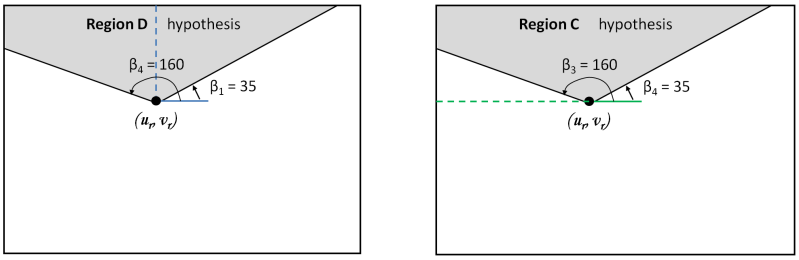


Figure A.3: Two hypotheses of our model with the same energy.

Assuming the rotation of the reference system to compute the features on the different regions, if the ranges of the angles β_k for each region expand to 180 degrees, there is an overlap area between adjacent regions (see Fig. A.4). For example, regions C and D share the upper part (in blue color) and similarly, regions B and C share the left part (in green), etc.

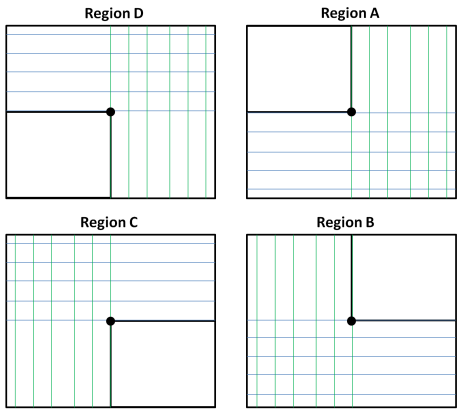


Figure A.4: Domain of the angles β_k for the proposed parameterization in Fig. A.1. Every region is bounded by two angles which are shown in green an blue vertical and horizontal stripes. All the regions share a full domain of angles, e.g. C and D share the upper part in blue color.

This situation can lead to different hypotheses of the search space having the same energy, thus causing ties which are an important drawback for the convergence of the message-passing algorithm [Schwing et al., 2011]. As a consequence, we devised a newer approach in the next section.

A.3. 4-roads layout. Approach 3.

The last section of this appendix presents the third approach that was tested with the approximate inference algorithm [Schwing et al., 2011]. After detecting some issues with the angles encoding in the model, as exposed in previous section, we propose a modification to solve them. Again, our basement is the parameterization in Fig. A.1 and factorization in Eq. A.1.

In order to remove the overlapping between the domains of the road angles β_i, we force tighter geometric constraints into the model. In particular, $\beta_1 = [-45, 45]$, $\beta_2 = [-135, -45]$, $\beta_3 = [135, -135]$ and $\beta_4 = [45, 135]$. Besides, these angles are counted anticlockwise from an horizontal line passing trough the corresponding intersecting point. This is equivalent to restrict the angles domain to the areas with square grids in Fig. A.4 and adding a rotation of 45 degrees for practical purposes. Then, the boundaries of the set of hypotheses for one region are no colinear with the reference axes of the image. Moreover, the values for α_{rl} , α_{rm} and α_{ru} (lower, middle and upper bounds respectively) are updated depending on each region r. Thus, the tupples $b_r = [\alpha_{rl}$, α_{rm} and $\alpha_{ru}]$ for each region are: $b_A = [-45, 0, 45]$, $b_B = [-135, -90, -45]$, $b_C = [135, 180, -135]$ and $b_D = [45, 90, 135]$.

Consequently, the pairwise constraints $\phi_k(\beta_i, \beta_j)$ defined on previous section are not required anymore, as they are intrinsically enforced with the limited domain of the angles. However, this modification also reduces the space of hypotheses. For instance, it would not be possible to model a region D with a triangle shape, centered at 90 degrees segment among 60 and 120 angles (considering any point in the image), which is depicted in Fig. A.5.

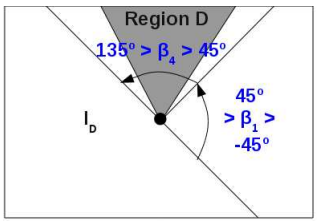


Figure A.5: Example of a region D that would not fit in the model proposed in approach 3.

To further support the convergence during inference, two constraints relating opposite intersection points are added to the ones defined in Section A.2.

$$\phi_{AC}(u_A, u_C) = \begin{cases} 0 & \text{if } u_A > u_C \\ -\infty & \text{other} \end{cases} \qquad \phi_{DB}(v_D, v_B) = \begin{cases} 0 & \text{if } v_D > v_B \\ -\infty & \text{other} \end{cases}$$

A.3.1. Discussion on experiments with approach 3.

The approaches described in this appendix have set many details on the research carried out during months for inferring the layout of road intersections. Apart from the angle particularization in approach 3, we have intensified on the addition of more specific geometrical constraints to guide the inference task. In fact, we tested the incorporation of elasticity constraints on the horizontal and vertical coordinates of the intersecting points (check the parameterization in Fig. A.1), as it is described by the expressions below:

$$\phi_{DB}(u_D, u_B) = \begin{cases} 0 & \text{if } u_D > u_B \\ -\infty & \text{other} \end{cases} \qquad \phi_{AC}(v_A, v_C) = \begin{cases} 0 & \text{if } v_C > v_A \\ -\infty & \text{other} \end{cases}$$

$$\phi_{ADadj}(u_A, u_D) = \begin{cases} 0 & \text{if } |u_A - u_D| < K_1 \\ K(u_A, u_D) & \text{other} \end{cases}$$

$$\phi_{BCadj}(u_B, u_C) = \begin{cases} 0 & \text{if } |u_B - u_C| < K_2 \\ K(u_B, u_C) & \text{other} \end{cases}$$

$$\phi_{ABadj}(v_A, v_B) = \begin{cases} 0 & \text{if } |v_A - v_B| < K_3 \\ K(v_A, v_B) & \text{other} \end{cases}$$

$$\phi_{CDadj}(v_C, v_D) = \begin{cases} 0 & \text{if } |v_C - v_D| < K_4 \\ K(v_C, v_D) & \text{other} \end{cases}$$

where the constant values K_1 - K_4 represent a bound that determines the maximum separation distance between the involved adjacent points. This bound is computed as the empirical covariance of the distance between each pair of random variables given the training dataset. On the other hand, the penalization coefficients on the horizontal $K(u_{r1}, u_{r2})$ and vertical $K(v_{r1}, v_{r2})$ directions, depend on the distance between points and some step factor empirically adjusted. In fact, the last four equations above represent hard constraints imposed by the inequality.

Moreover, several functions shapes for these constraints have been tested in order to prevent abrupt changes in the energy distribution of the search space, because this can lead to convergence problems during learning and inference as we observed experimentally. We employed the shapes: heavyside step function; jump + step; exponential and scaled versions with different factors. After the experiments, some of them lead to slight improvements or better predicted layouts. However, we were not able to find a function that achieves a smooth global energy objective for a successful learning of the parameter vector $\boldsymbol{\theta}$. Similarly, the inference engine was not able to find the optimal solution for the synthetic images given the learned parameter vector. We depict in Fig. A.6 some of the wrongly predicted layouts from the a set of experiments and model configurations tested.

Alternatively, unary potentials for the 12 discrete random variables in the factor graph of Fig. A.2 were also added into the learning and inference framework. These evidence data was computed from the real dataset of BeP images [Geiger et al., 2011a]. Despite a small reduction in the pixel-wise error, the message-passing algorithm was not able to find the optimal solution after inference.

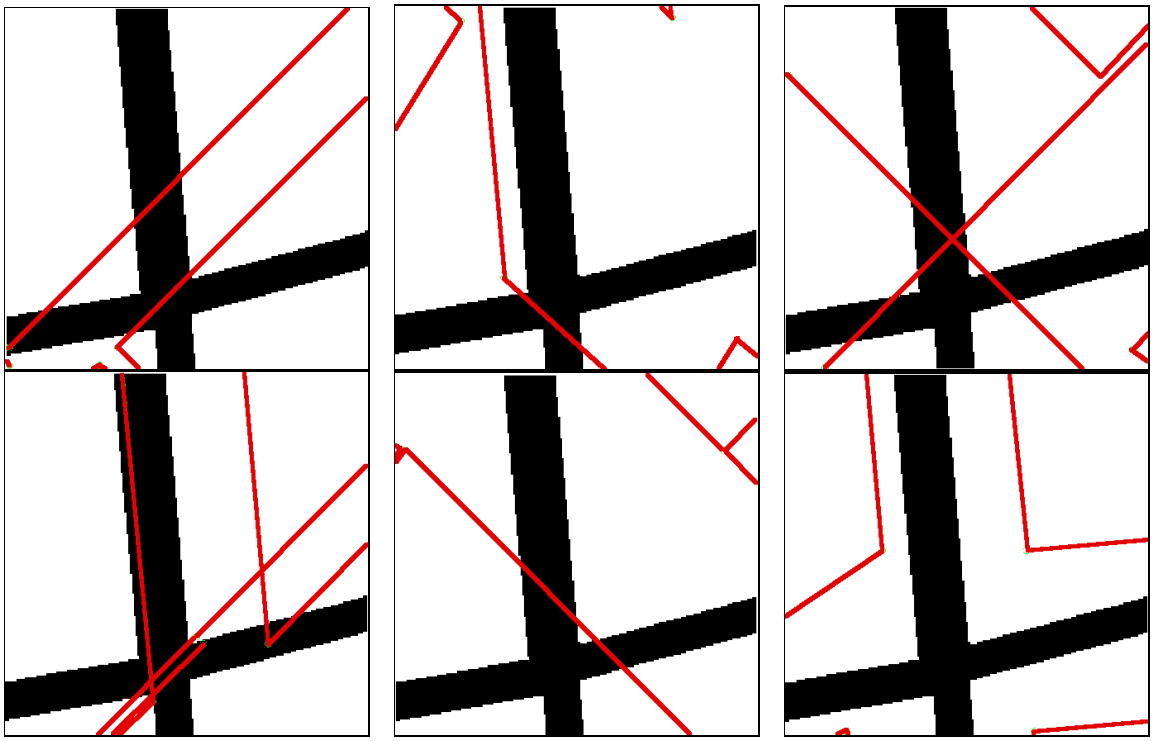


Figure A.6: Examples of wrongly predicted intersections employing Approach 3 and different sets of constraints. The ground truth has been discretized and painted in black pixels representing the road area. The green lines are the predictions, which do not fit the ground-truth model, in spite of being ideal images (black and white).

Appendix B

Additional results from object detection experiments

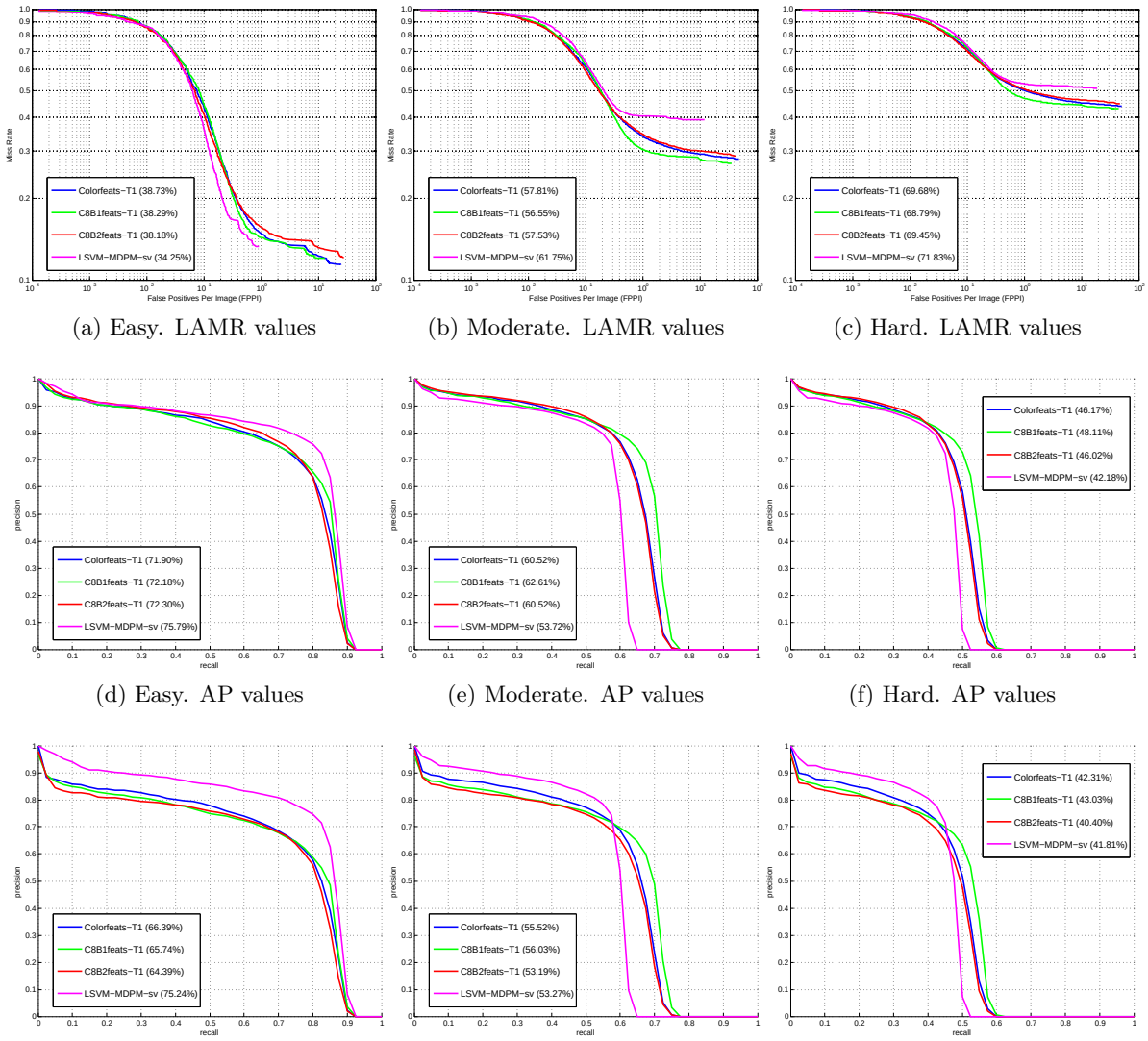
Several comparative plots of the experiments carried out along the research of the Thesis are attached in this appendix. They are supplementary material that provides further details over the main results in Chapter 5. The experimental results are divided in the three object categories of interest: cars, pedestrians and cyclists.

B.1. Experiments for the class 'car'

Figures B.1-B.7 compare 3D-aware and color-based features when employing different DPM parameter tuning against the baseline MDPM-LSVM-sv [KITTI, 2012]. Every figure contains 9 subplots, in which the three rows correspond to LAMR, AP and AOS measurements and the columns are related to each difficulty level evaluated (see Section 5.2).

Test 1 yields the 4 curves that are depicted in Fig. B.1, which correspond to MDPM-LSVM-sv and the use of 3 different features: the original HOG color in [Felzenszwalb et al., 2010a] and the newly contributed C8B1 and C8B2. The supervised learning of the models is done based on the configuration of *Medium-T7* in Section 5.3.2, but adding the upsampling by factor 3 of the small ground-truth samples.

Compared to the baseline MDPM-LSVM-sv, the DPM tunning and features in *Test 1* produce an important improvement in AP for moderate (8%) and hard (6%) levels. Besides, there is also a slight increase when adding disparity (C8B1 and C8B2) vs the use of color features alone. However, this is not accomplished for 'easy' samples (Fig. B.1.d). Then, similar conclusions can be extracted from orientation estimation (Fig. B.1.g,h,i), also motivated by the increase or decrease in AP depending on the evaluation category (=difficulty level). According to these results, the best performing feature is *C8B1*, which also obtains the lowest miss rate. Additionally, observing Fig. B.1.b, the green curve achieves an important 10% reduction in the miss rate compared to the baseline (magenta) at 1 false positive per image. On the other hand, as can be seen in Fig. B.1.a,b,c the longer tails of our proposals indicate further lowering down



(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

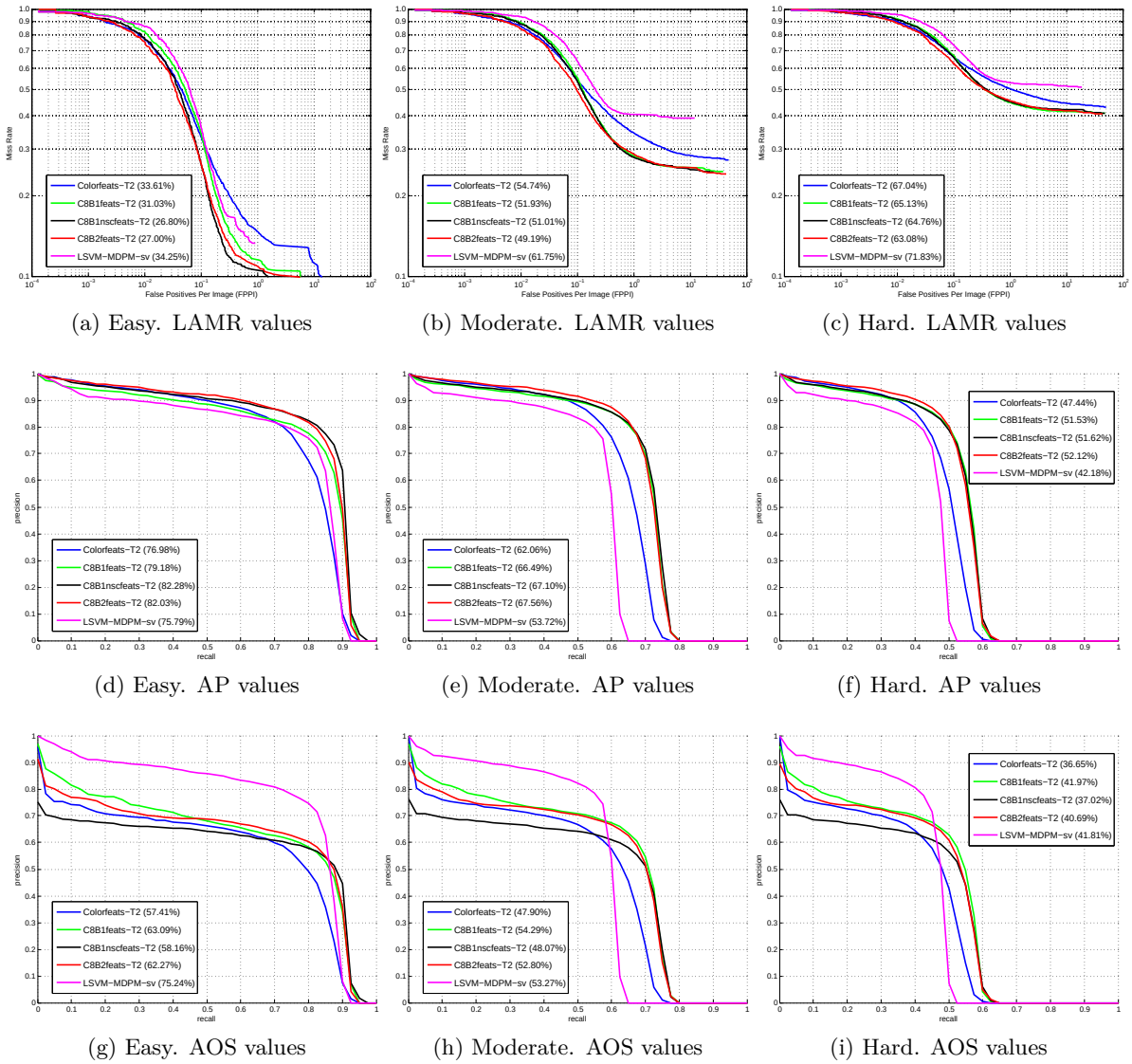
(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure B.1: Evaluation of 3D-aware features for cars, Test 1.

the miss rate at the cost of having more false positives, which is not the best scenario, but we also achieve reduced miss rate at low fppi $[10^{-1}, 5]$ for moderate and hard evaluation levels.

Fig. B.2 displays the results for Test 2, which has the same configuration as Test 1, but changing the latent overlap to 80%. Besides, we add and additional plot *C8B1nscfeats*, which corresponds to the same *C8B1* feature but not scaling the disparity (see Section 4.2.1).

Test 2 yields an important boost in AP (7%, 14% and 10% maximum differences respectively for each difficulty level) compared to the baseline. This behavior is due to the tighter overlap constraint for latent parts and components. However, AOS decreases in all the cases, but keeping a moderate value at higher recalls (curves pushed to the right). Thus, *C8B1* and *C8B2* obtain superior AOS for the difficult samples compared to the baseline MDPM-LSVM-sv. On the contrary, the improvement due to the addition of disparity cues is more notable in the orientation estimation (green, black and red lines in Fig. B.2.g-i), being *C8B1* the best performing one. Hence, we can conclude that not scaling the disparity causes a poorer precision



(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

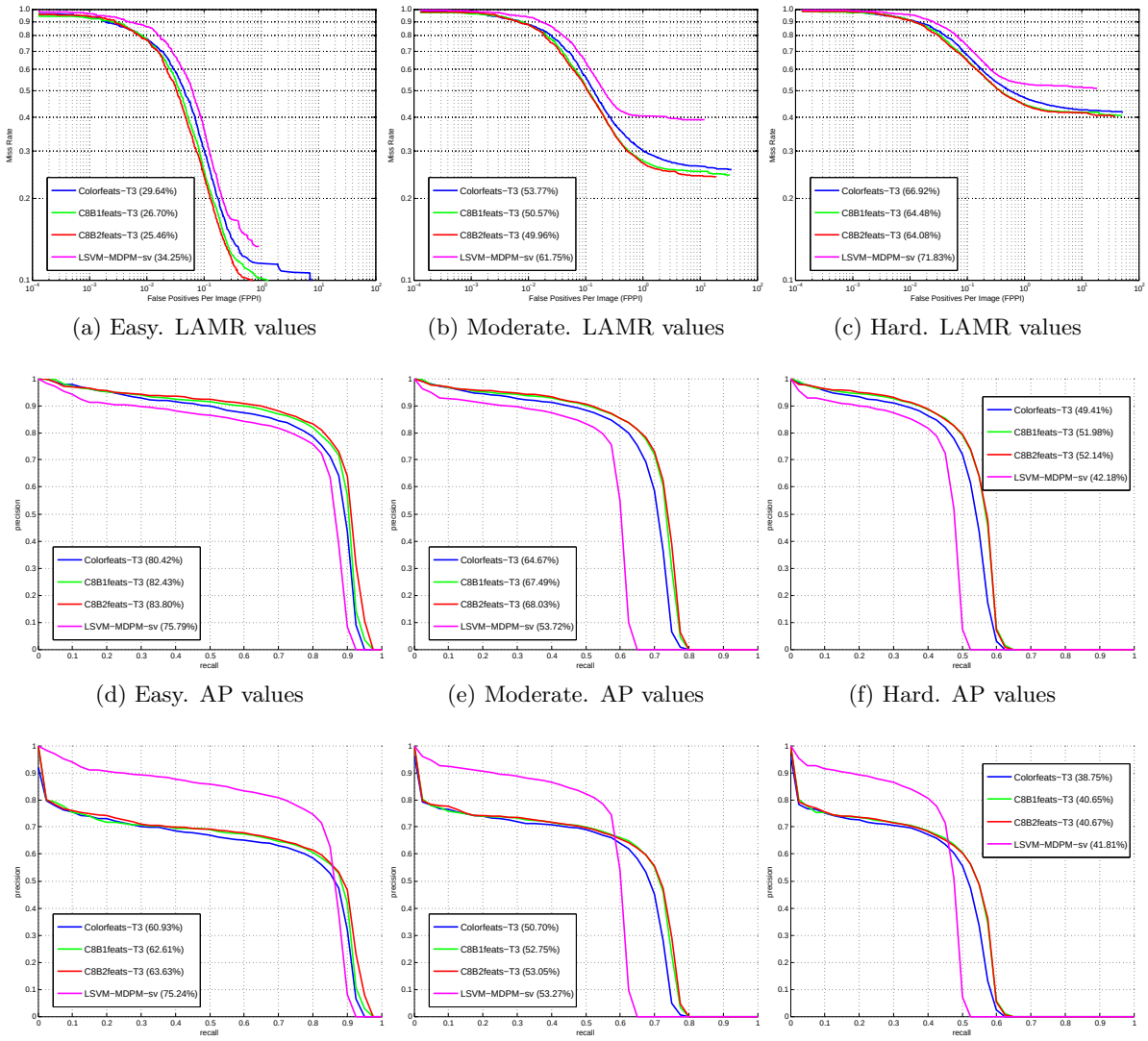
(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure B.2: Evaluation of 3D-aware features for cars, Test 2.

in orientation estimation. Another remarkable achievement is the very low *fppi* rate at 0.1 miss rate, when using our proposed 3D-aware features.

Fig. B.3 displays the results for Test 3, which has a DPM configuration as in Test 2, but changing the regularization of L-SVM such that: C=0.001 during the model initialization and components merging, and C=0.0001 during latent parts training. This is motivated by the larger size of the dataset [Zhu et al., 2012] when learning the latent parts. This approach yields a slight improvement in AP and AOS for the easy samples and replicates similar conclusions that were regarded in the previous test.

Next, Fig. B.4 shows a more extensive analysis on feature combinations for a configuration as in Test 3, but loosing the latent overlap requirement to 75%. The objective is to achieve a trade-off between object detection and orientation estimation accuracies, which means improving orientation estimation at the cost of reducing precision for object detection. This supervised learning setup was already selected in Section 5.3.2 (*Medium-T8*) as a good candidate to train



(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

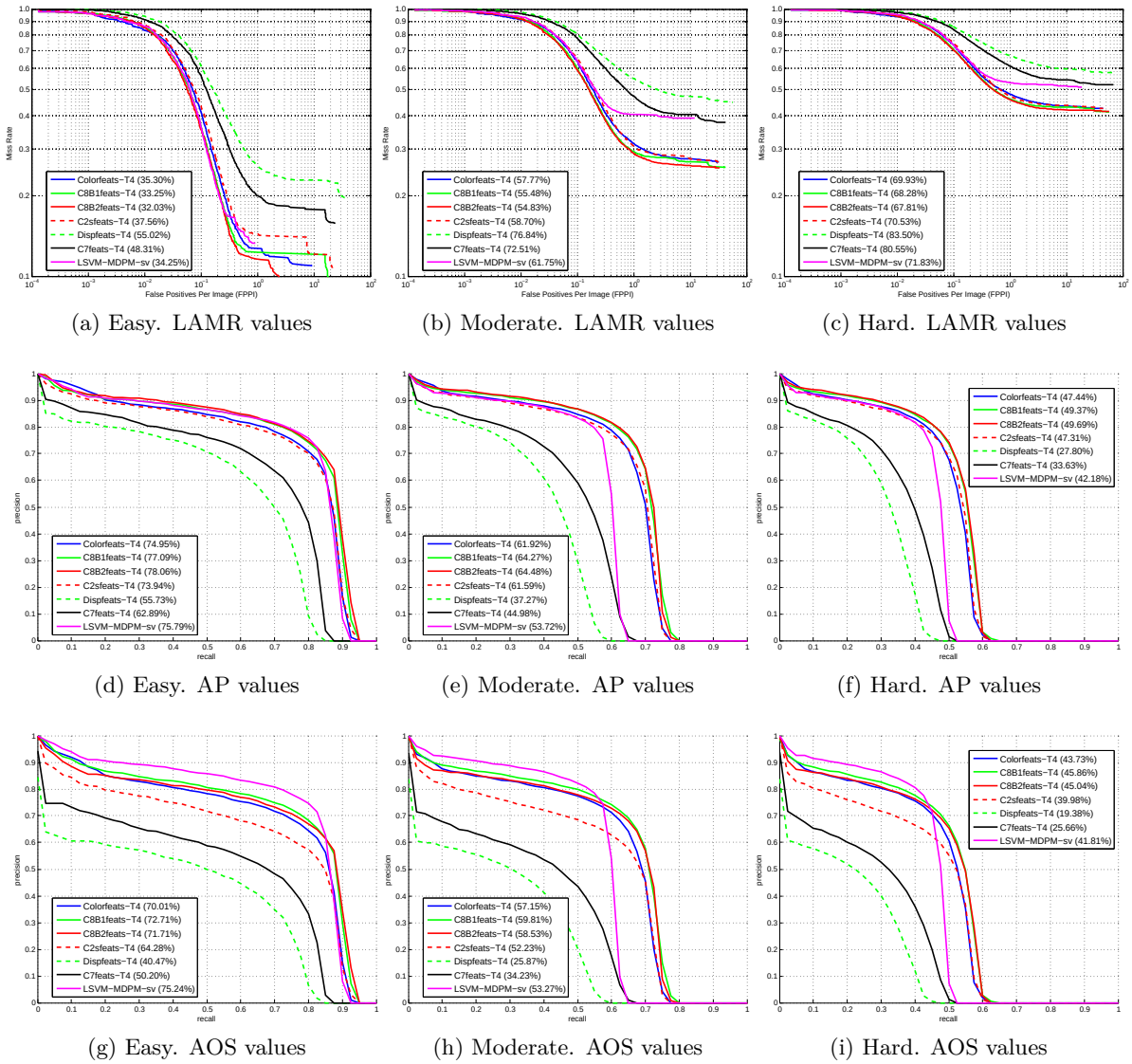
(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure B.3: Evaluation of 3D-aware features for cars, Test 3.

a model of cars that generalizes well to unseen data (testset). Therefore, Fig. B.4 contains more plots from other 3D-aware feature proposals (check Section 4.2 for details). Nevertheless, it must be clarified that "C2s" refers to "C2" features computed from scaled disparity, and "Disp" are histograms of gradients computed on the disparity map without any color data.

This test produces increased values for AP and AOS employing only *C8B1* and *C8B2* features (green and red continuous lines respectively) than using their concatenation *C2s* (red dashed lines), which also leads to a speedup because of their lower dimensionality. They also show a lower number of fppi at low miss rate for easy samples (Fig.ẽffig:test4.a). Moreover, the 'p-r curves' for orientation estimation based on *C2s* are below the ones for *Color* (blue lines). Interestingly, *C8B2* (red continuous line) peaks in the AP for all evaluated levels, but *C8B1* (green continuous line) is superior for AOS, which is reasonably explained due the nature of its contrast-sensitive features ($0 - 2\pi$ gradient orientations). Indeed, *C8B1* yields 1% increase over



(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

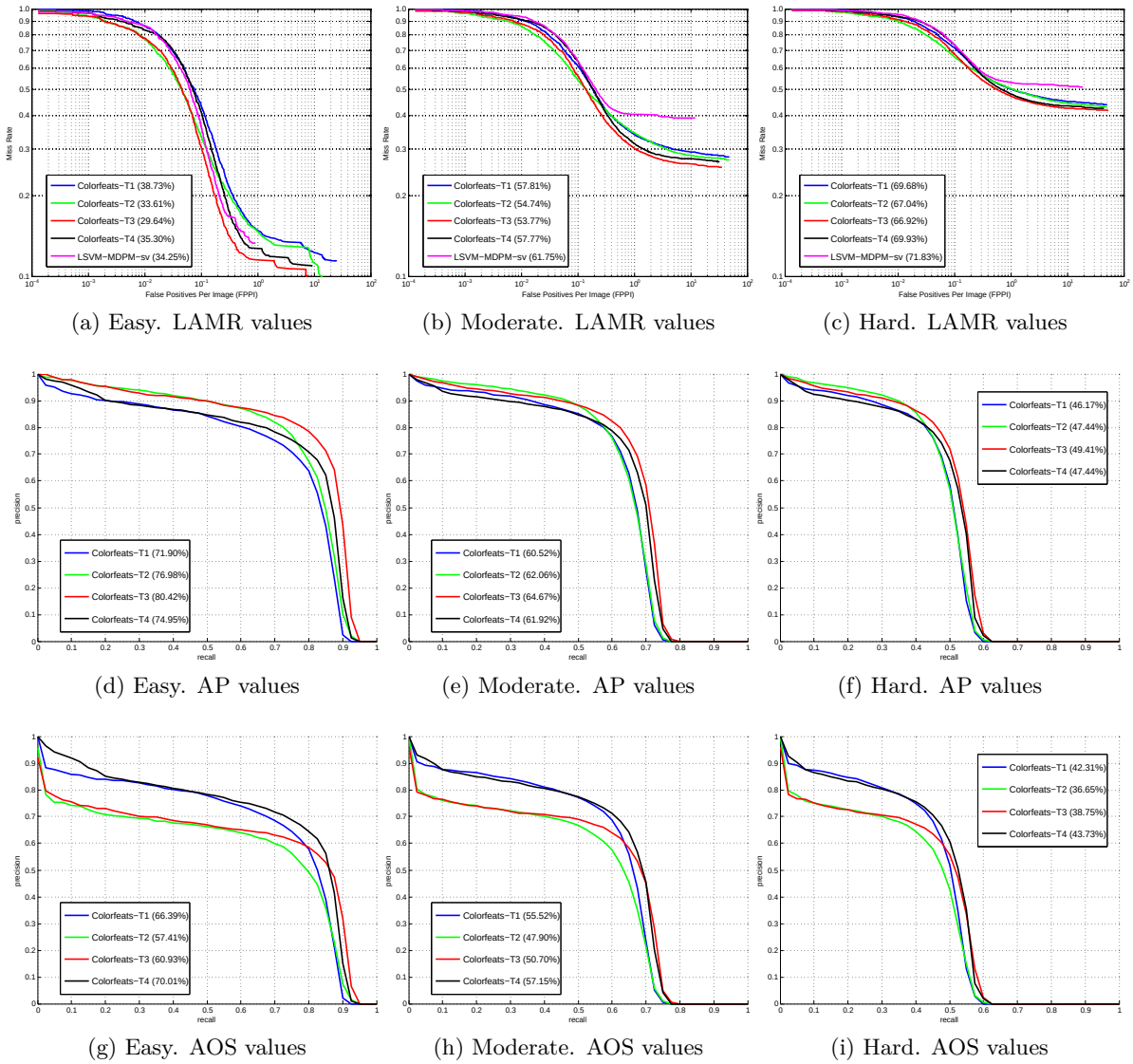
(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure B.4: Evaluation of 3D-aware features for cars, Test 4.

C8B2 at all difficulties, so it is preferred over *C8B2*. In addition, the *Dispfeats* show the lowest precision (green dashed line) in all plots of Fig. B.4, which demonstrates that the use of disparity by itself is not beneficial, but its combination with color cues yields a moderate improvement. Moreover, *C7* does not show any contribution as demonstrated by its low precision values in all cases. In summary, the biggest benefits when using the 3D-aware features *C8B1* and *C8B2* are for the difficult samples, where our proposals clearly outperforms the baseline MDPM-LSVM-sv.

Given the previous precision-recall curves and analysis, we also consider of interest to show the individual evolution of each feature depending on the setup for the supervised learning of the DPM mixture models. Then, Fig. B.5 to B.6 depict the evolution of the features *C8B1, C8B2* and *Color* in the previous tests. From these figures, independently of the feature, it can be seen that a higher latent overlap requirement during training produces important gains in AP (red, green and black lines). However, the 80% of overlap (red and green lines) yields lower precision when estimating the cars viewpoint. On the contrary, setting two different

values for L-SVM C parameter (red above green and black above blue lines) clearly benefits the orientation estimation, but also the object detection, showing both increased precision and recall, resulting in higher AP and AOS figures. Nevertheless, there is one exception, the AOS plots in Fig. B.6 do not show a benefit when changing C parameter at latent overlap of 80%, i.e. the plots are sorted in decreasing order as blue-green-red, instead of blue-red-green in previous figures. This may be related to the length of the *C8B1*, which is longer than *Color* and *C8B2* features.



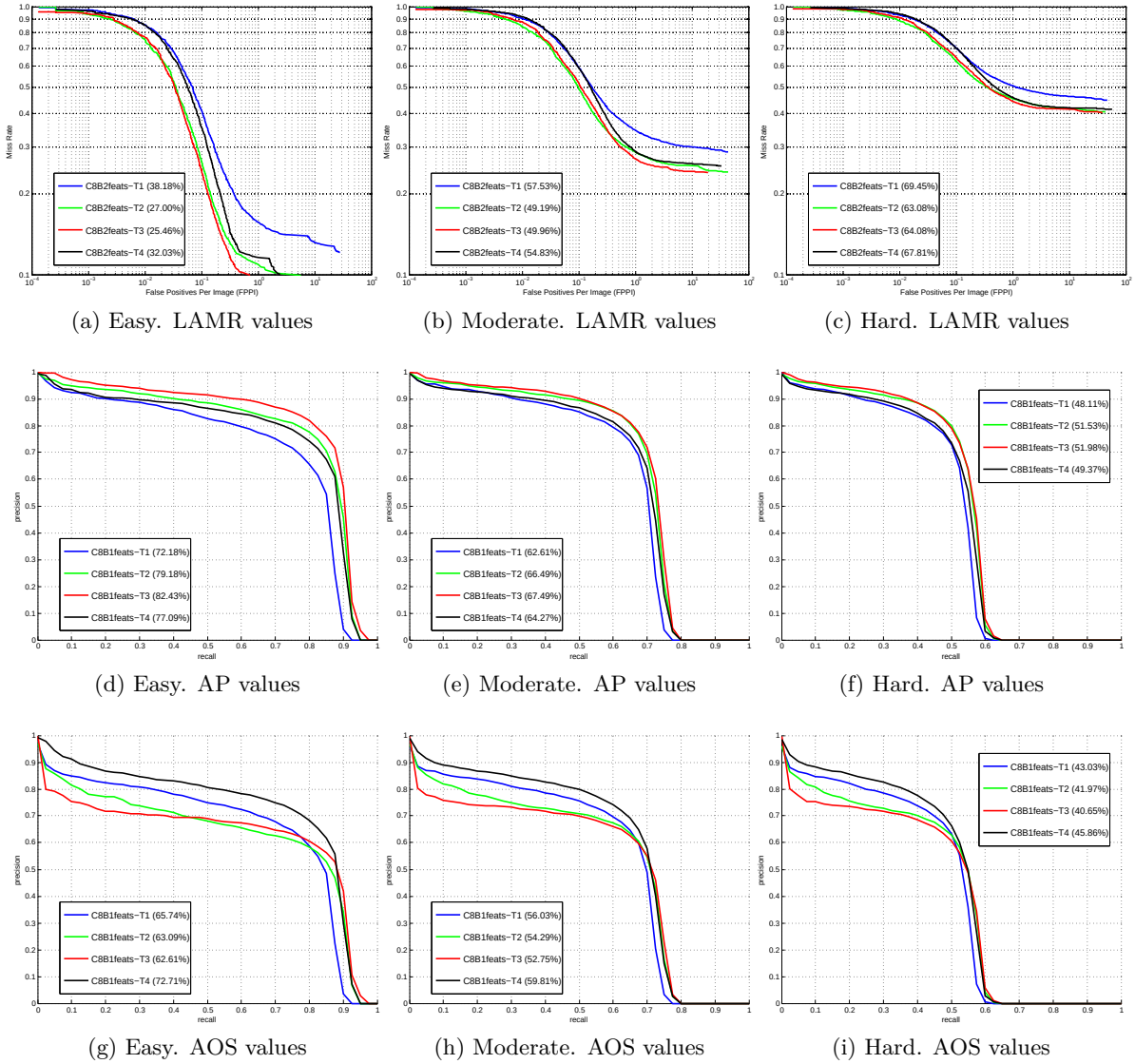
(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure B.5: Performance evolution of HOG color features for cars and different setups of the supervised learning in Tests 1-4.

Finally, the configuration proposed in Test 3 is the best performing one, independently of the tested features, if we consider only the object detection challenge. On the other hand, the configuration in Test 4 yields the best results in orientation estimation, showing also a good trade-off for object detection, in both cases outperforming the baseline MDPM-LSVM-sv.

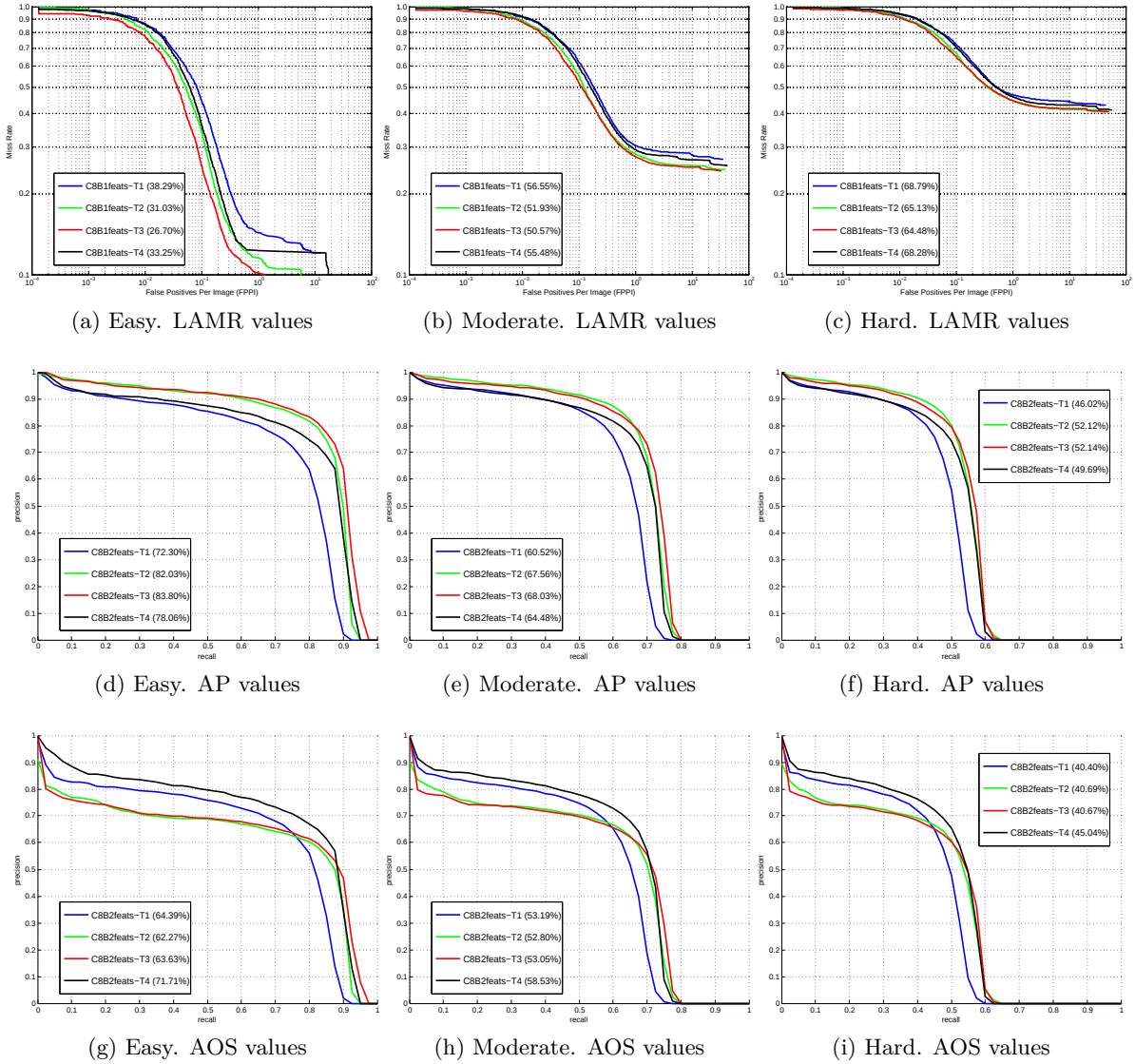


(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure B.6: Performance evolution of C8B1 features for cars and different setups of the supervised learning in Tests 1-4.



(a) Easy. LAMR values (b) Moderate. LAMR values (c) Hard. LAMR values

(d) Easy. AP values (e) Moderate. AP values (f) Hard. AP values

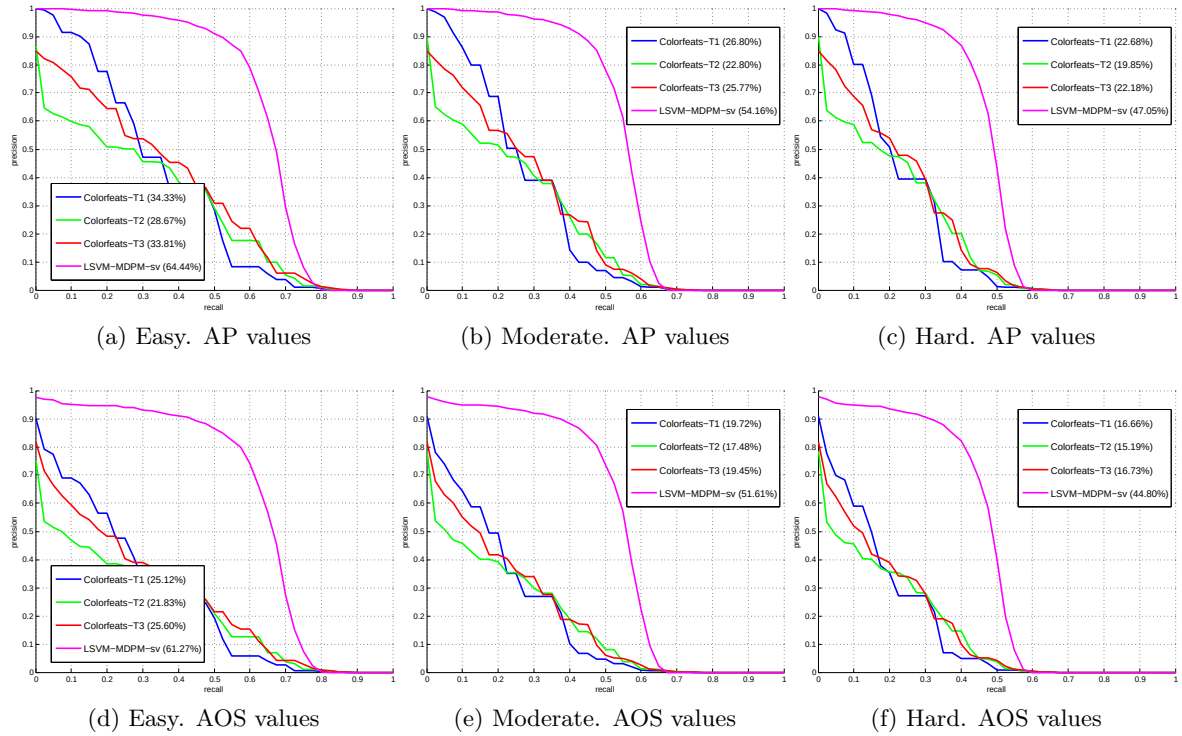
(g) Easy. AOS values (h) Moderate. AOS values (i) Hard. AOS values

Figure B.7: Performance evolution of C8B2 features for cars and different setups of the supervised learning in Tests 1-4.

B.2. Experiments for the class 'pedestrian'

This Section presents experimental results for the class 'pedestrian'. Following the same experimental methodology employed before, we carried out a set of 5-fold cross-validation tests based on the supervised learning setup of the previous Test 4 for the class Car. This is because it yielded a good trade-off between object detection and orientation estimation ratios. On the other hand, due to initialization errors during the training pipeline of DPM framework for pedestrians, we have adapted the initialization of the parts size to a more appropriate aspect ratio. Hence, instead of 6x6 cells parts, we enforce 4x2 parts for each component of the model. Besides, let us remind also that the number of components for this category is 8, as we already mentioned in Section 5.1.

Therefore, three tests based on the selected setup are compared: **Test 1**. 8 parts of size 4x2 and overlap requirement of 80% during latent search; **Test 2**. Same as before with a lower overlap of 75% and **Test 3**. Same as the second test but lowering the number of parts to 6. These tests are further split depending on the feature. In particular, *Color*, *C8B1* and *C8B2* are compared against the pre-trained baseline LSVM-MDPM-sv [KITTI, 2012]. The precision-recall curves are plotted in Fig. B.8-B.9.

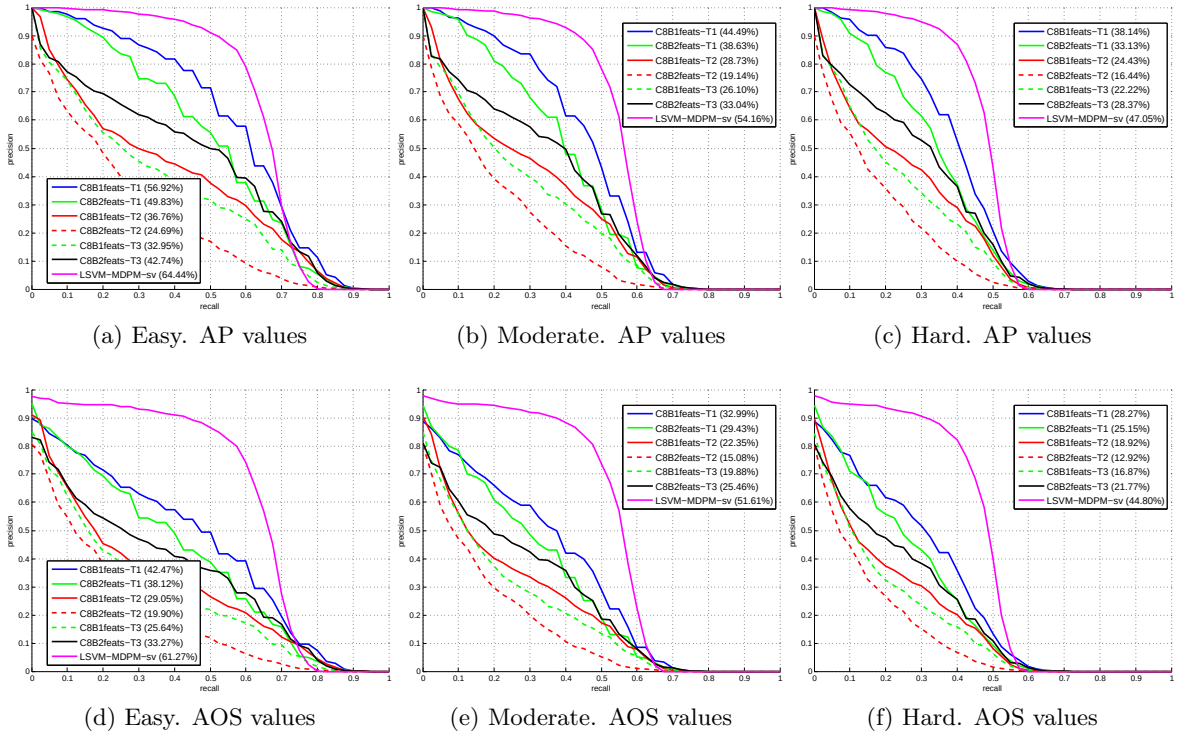


(a) Easy. AP values (b) Moderate. AP values (c) Hard. AP values

(d) Easy. AOS values (e) Moderate. AOS values (f) Hard. AOS values

Figure B.8: Evaluation of color features for pedestrians, Tests 1-2-3.

As can been seen in the plots, the benefits observed for cars are not repeated for pedestrians. Indeed, the results are importantly below the baseline. Our best guess is that this maybe related to wrong initialization of fixed parameters of the model, i.e. root and part filters size and LSVM regularization constant. The need for a good seeds for the model was also marked as important by the seminar work of [Felzenszwalb et al., 2010b]. However, the good news is that 3D-aware



(a) Easy. AP values (b) Moderate. AP values (c) Hard. AP values

(d) Easy. AOS values (e) Moderate. AOS values (f) Hard. AOS values

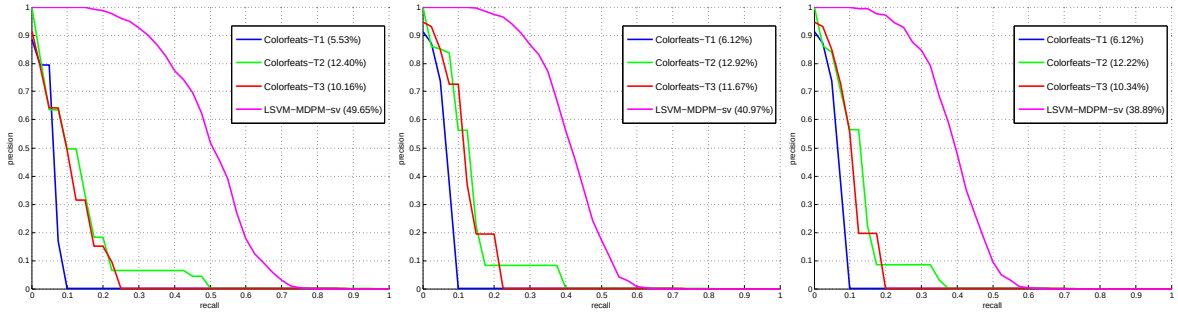
Figure B.9: Evaluation of 3D-aware features for pedestrians, Tests 1-2-3.

features perform better than our training experiments based only on color data. Particularly, both features *C8B1* and *C8B2* in setup configuration of Test 1.

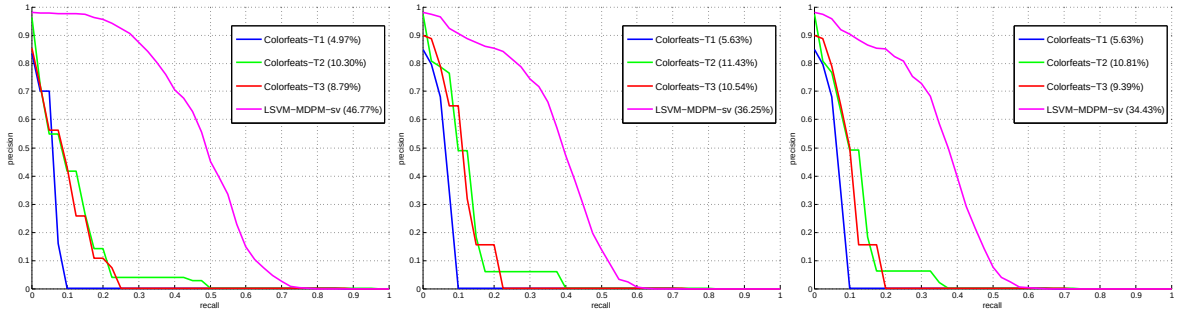
The explanation of "Test 1" success can be related to the tighter fit imposed during latent search (80% overlap), that makes the model to concentrate on learning human body parts while discarding the background as much as possible.

B.3. Experiments for the class 'cyclist'

As already reported for the class 'pedestrian', we attach here the results for the same test setups as above, but employing 'cyclist' image patches. The precision-recall curves are depicted in Fig. B.10-B.11, where very similar conclusions can be regarded about the prediction capacity of the learned models. Nonetheless, there is a difference in the training setup that obtains the highest precision. In this case, it is Test 2. Perhaps, this is motivated by background information around the cyclists, such that employing an overlap of 75% (lower than 80%), samples not too tightly fitted to the ground truth lead to a better model learning. In fact, the cyclists could be differentiated from pedestrians given additional features around them, because they use to be on the road and parked cars maybe also behind them or at their sides.

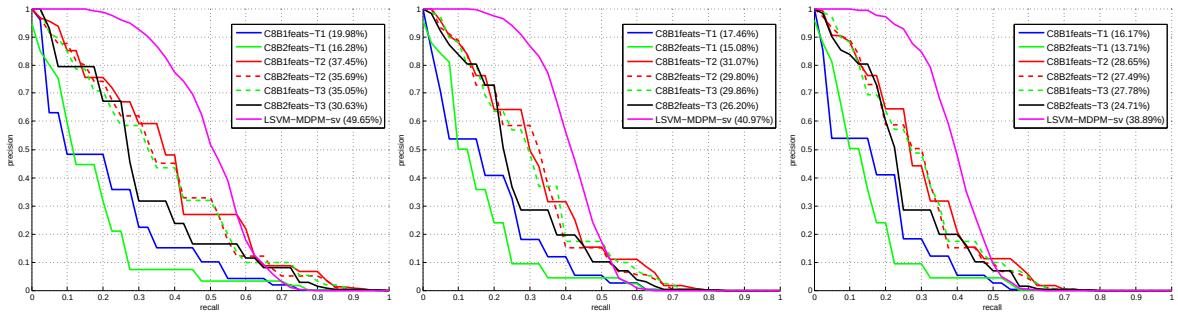


(a) Easy. AP values (b) Moderate. AP values (c) Hard. AP values

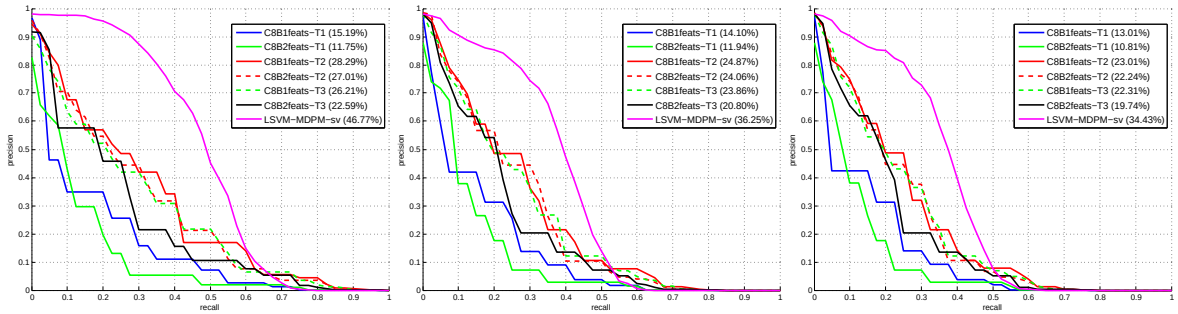


(d) Easy. AOS values (e) Moderate. AOS values (f) Hard. AOS values

Figure B.10: Evaluation of color features for cyclists, Tests 1-2-3.



(a) Easy. AP values (b) Moderate. AP values (c) Hard. AP values



(d) Easy. AOS values (e) Moderate. AOS values (f) Hard. AOS values

Figure B.11: Evaluation of 3D-aware features for cyclists, Tests 1-2-3.