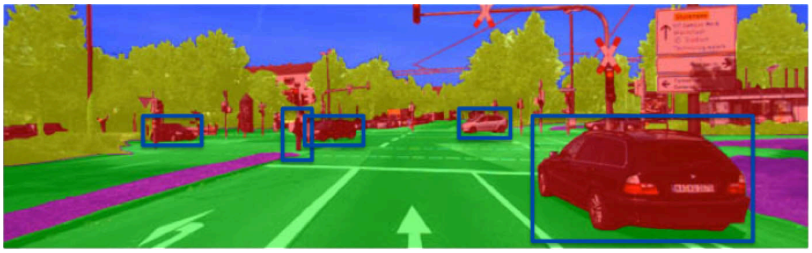
Information fusion for scene understanding

Philippe Xu

Par **Philippe XU**

Information fusion for scene understanding

Thèse présentée
pour l'obtention du grade
de Docteur de l'UTC



Soutenue le 28 novembre 2014
Spécialité : Technologies de l'Information et des Systèmes

D2153

Information fusion for scene understanding

Philippe XU

Université de Technologie de Compiègne

28th November 2014

JURY

Isabelle Bloch	Professor, Télécom ParisTech (France)	(Rapporteur)
Franck Davoine	Researcher, UTC (France)	(Directeur de thèse)
Thierry Denœux	Professor, UTC (France)	(Directeur de thèse)
Yves Grandvalet	Senior researcher, UTC (France)	(Président)
Frédéric Jurie	Professor, Université de Caen (France)	(Examinateur)
Patrick Pérez	Senior researcher, Technicolor (France)	(Rapporteur)
Hongbin Zha	Professor, Peking University (China)	(Examinateur)

Preface

This thesis was done at the Heudiasyc laboratory of the Université de Technologie de Compiègne. It was carried out in the framework of the Labex MS2T (Control of Technological Systems of Systems), which was funded by the French Government, through the program "Investments for the future" managed by the National Agency for Research (ANR-11-IDEX-0004-02). Our work contributed to the second axis of the Labex ("Managing uncertainties").

This thesis was also part of a Sino-French research collaboration through the MPR (Multimodal Perception and Reasoning) project of the LIAMA laboratory (Laboratoire Sino-Européen d'Informatique, d'Automatique et de Mathématiques Appliquées). The first half of this thesis was conducted at the Key Laboratory of Machine Perception of Peking University. The work done in China was financed by the Cai Yuanpei program (26193PE) from the Chinese Ministry of Education, the French Ministry of Foreign and European Affairs and the French Ministry of Higher Education and Research. This thesis was also supported by the ANR-NSFC Sino-French PRETIV project (ANR-11-IS03-0001). For this project, we contributed to the Task 1 ("Multimodal perception") and the Task 2 ("Reasoning and scene understanding").

I would like to gratefully thank my supervisors Franck Davoine and Thierry Denœux for guiding and supporting me throughout this PhD thesis. I am also most grateful to the professors of Peking University, Huijing Zhao, Jinshi Cui and Hongbin Zha, for welcoming me into their research group. My thanks also go to the researchers of Heudiasyc, Vincent Frémont, Véronique Cherfaoui, Philippe Bonnifait, Gérald Dhergomez, Benjamin Quost, Sébastian Destercke, Yves Grandvalet, Antoine Bordes and Nicolas Usunier, with whom I had many insightful discussions about many aspects of research.

I would also like to thank my colleagues of the Ecole Centrale de Pékin, Marc Pauly, Yves Dulac, Denis Monasse, Adeline Minet and Guillaume Merle, with whom I learned to teach.

I wish to thank all the current and past students with whom I shared some wonderful time both in Beijing and Compiègne. Special thanks go to Brice Dudout, Wu Jia, Pascal Zille, Jean-Baptiste Bordes, Gautier Marti and Martin Boissier for making my stay in China an unforgettable experi-

ence. I thank PKU students, Yao Wen, Wang Chao, Yu Yufeng, Lin Yubin, He Mengwen, Xu Donghao, Zhao Yipu, Huang Yingning and UTC students, Zhou Dingfu, Li Hui, Wang Bihao, Yu Chunlei, Shameem Puthiya parambath, Liu Xiao, among many others, for struggling along with me in the world of research.

Finally, my warmest thanks go to my parents, my brother and my better half for their never-ending love and support.

Abstract

Image understanding is a key issue in modern robotics, computer vision and machine learning. In particular, driving scene understanding is very important in the context of advanced driver assistance systems for intelligent vehicles. In order to recognize the large number of objects that may be found on the road, several sensors and decision algorithms are necessary. To make the most of existing state-of-the-art methods, we address the issue of scene understanding from an information fusion point of view.

The combination of many diverse detection modules, which may deal with distinct classes of objects and different data representations, is handled by reasoning in the image space. We consider image understanding at two levels: object detection and semantic segmentation. The theory of belief functions is used to model and combine the outputs of these detection modules. We emphasize the need of a fusion framework flexible enough to easily include new classes, new sensors and new object detection algorithms.

In this thesis, we propose a general method to model the outputs of classical machine learning techniques as belief functions. Next, we apply our framework to the combination of pedestrian detectors using the Caltech Pedestrian Detection Benchmark. The KITTI Vision Benchmark Suite is then used to validate our approach in a semantic segmentation context using multi-modal information.

Keywords Information fusion · Driving scene understanding · Theory of belief functions · Dempster-Shafer theory · Object detection · Semantic segmentation

Résumé

La compréhension d'image est un problème majeur de la robotique moderne, la vision par ordinateur et l'apprentissage automatique. En particulier, dans le cas des systèmes avancés d'aide à la conduite, la compréhension de scènes routières est très importante. Afin de pouvoir reconnaître le grand nombre d'objets pouvant être présents dans la scène, plusieurs capteurs et algorithmes de classification doivent être utilisés. Afin de pouvoir profiter au mieux des méthodes existantes, nous traitons le problème de la compréhension de scènes comme un problème de fusion d'informations.

La combinaison d'une grande variété de modules de détection, qui peuvent traiter des classes d'objets différentes et utiliser des représentations distinctes, est faite au niveau d'une image. Nous considérons la compréhension d'image à deux niveaux : la détection d'objets et la segmentation sémantique. La théorie des fonctions de croyance est utilisée afin de modéliser et combiner les sorties de ces modules de détection. Nous mettons l'accent sur la nécessité d'avoir un cadre de fusion suffisamment flexible afin de pouvoir inclure facilement de nouvelles classes d'objets, de nouveaux capteurs et de nouveaux algorithmes de détection d'objets.

Dans cette thèse, nous proposons une méthode générale permettant de transformer les sorties d'algorithmes d'apprentissage automatique en fonctions de croyance. Nous étudions, ensuite, la combinaison de détecteurs de piétons en utilisant les données du Caltech Pedestrian Detection Benchmark. Enfin, les données du KITTI Vision Benchmark Suite sont utilisées pour valider notre approche dans le cadre d'une fusion multimodale d'informations pour de la segmentation sémantique.

Mots-clés Fusion d'informations · Compréhension de scènes routières · Théorie des fonctions de croyance · Théorie de Dempster-Shafer · Détection d'objets · Segmentation sémantique

摘要

图像理解是现代机器人学、计算机视觉和机器学习领域的关键问题之一。对行驶场景的理解在高级智能驾驶辅助系统中至关重要。为了更好地识别大量的路面上的物体，通常我们需要使用多个传感器和不同的识别算法。在这里我们从信息融合的角度来解决场景理解的问题，以求获得先进的识别算法。

不同种类的物体使用不同的模型来识别，因为不同的物体需要使用不同的数据来表达。我们考虑从图像空间的角度来组合这些不同的识别方法。我们将从物体检测和图像语义分割这两个的层次来考虑图像理解。借助于信任函数理论，我们实现对多种检测模块输出的建模和组合。值得强调的是，我们需要建立一个灵活自由的数据融合框架，以便于随时加入新的物体类别、传感器数据和检测方法。

本文提出了一种通用的将经典的机器学习的输出转化为信任函数的方法。利用提出的方法，我们组合所有发布在Caltech Pedestrian Detection Benchmark行人检测标准上的行人检测器的输出来得到最终的检测结果。同时我们利用KITTI Vision Benchmark Suite的视觉标准数据来验证我们的多传感器数据融合方法在解决场景语义分割上的有效性。

关键词 信息融合 · 路况图像理解 · 信任函数 · Dempster-Shafer理论 · 物体检测 · 意义分割

List of Publications

International journals

Ph. Xu, F. Davoine, J.-B. Bordes, H. Zhao and T. Denœux.
Multimodal Information Fusion for Urban Scene Understanding.
Machine Vision and Applications, 2014. (Accepted for publication)

National journals

Ph. Xu, F. Davoine, J.-B. Bordes and T. Denœux.
Fusion d'informations pour la compréhension de scènes.
Traitement du Signal, 31(1–2):57–80, 2014.

International conferences and workshops

Ph. Xu, F. Davoine and T. Denœux.
Evidential Combination of Pedestrian Detectors.
British Machine Vision Conference (Oral), 2014.

Ph. Xu, F. Davoine and T. Denœux.
Evidential Logistic Regression for Binary SVM Classifier Calibration.
International Conference on Belief Functions, 2014.
Best Student Paper Award

Ph. Xu, F. Davoine, J.-B. Bordes, H. Zhao and T. Denœux.
Information Fusion on Oversegmented Images: An Application for Urban Scene Understanding.
International Conference on Machine Vision Applications (Oral), 2013.

J.-B. Bordes, Ph. Xu, F. Davoine, H. Zhao and T. Denœux.
Information Fusion and Evidential Grammars for Object Class Segmentation.
IROS Workshop on Planning, Perception and Navigation for Intelligent Vehicles, 2013.

J.-B. Bordes, F. Davoine, Ph. Xu and T. Denœux.
Evidential Grammars for Image Interpretation. Application to Multimodal Traffic Scene Understanding.
International Symposium on Integrated Uncertainty in Knowledge Modeling and Decision Making, 2013.

National conferences

Ph. Xu, F. Davoine and T. Denœux.
Transformation de scores SVM en fonctions de croyance.
Congrès national sur la Reconnaissance de Formes et l'Intelligence Artificielle (RFIA'14), 2014.

Ph. Xu, F. Davoine, J.-B. Bordes and T. Denœux.
Fusion d'informations sur des images sursegmentées : une application à la compréhension de scènes routières.
Congrès des jeunes chercheurs en vision par ordinateur (ORASIS'13), 2013.

Datasets and code

KITTI semantic segmentation ground truth

A set of 110 images from the KITTI Vision Benchmark Suite were manually annotated with the software Adobe® Photoshop® CS2. They can be downloaded at https://www.hds.utc.fr/~xuphilip/dokuwiki/en/data.

Combination of pedestrian detectors

The MATLAB® code to combine pedestrian detectors can be downloaded at https://www.hds.utc.fr/~xuphilip/dokuwiki/en/data.

Contents

List of Figures

List of Tables

Acronyms & notations

ALE	Automatic Labeling Environment
BB	Bounding Box
CRF	Conditional Random Field
FPPI	False Positive Per Image
HOG	Histogram of Oriented Gradient
KITTI	Karlsruhe Institue of Technology and Toyota Techonological Institute at Chicago
LiDAR	Light Detection And Ranging
MSE	Mean-Squared Error
NMS	Non-Maximal Suppression
PAV	Pair-Adjacent Violators
RGB	Red, Green and Blue color channels
ROC	Receiver Operating Characteristic
ROI	Region Of Interest
SLAM	Simultaneous localization and mapping
SLIC	Simple Linear Iterative Clustering
VOC	Visual Object Classes (PASCAL challenge)

A^w	simple mass function defined as $m(A) = 1 - w$, $m(\Omega) = w$
Bel	belief function
$BetP$	pignistic probability
$\mathcal{F}$	set of focal elements
m	mass function
${}^{\alpha}m$	discounted mass function
P	probability distribution
Pl	plausibility function
pl	contour function
$\mathbf{x}$	observation vector
$\mathbb{X}$	feature space of observations
Ω	frame of discernment

Introduction

Scene understanding is a very important task for different applications: robotics, human machine interaction, surveillance, etc. For automotive applications, sub-tasks such as road recognition, pedestrian detection or traffic sign understanding, among many others, are already by themselves very challenging. Many algorithms have been developed over the last decades to tackle these individual problems, each of them using possibly different kinds of sensors. For autonomous vehicles, it is necessary to have a deep understanding of the driving scene. The vehicles need to assess the 3D structure of the scene, detect static and moving objects and analyze their dynamic interactions. In this thesis, we do not tackle this task in its entirety but focus on advanced driver assistance systems for intelligent vehicles. The goal is to warn the driver of potential dangers by projecting relevant information on an image that reflects what the driver sees.

Scene understanding, when seen as an image understanding problem, is generally handled at two levels: object detection and semantic segmentation. The computer vision community puts extensive efforts into object detection issues. In particular, pedestrian detection is one of the most studied cases. Dozens of new pedestrian detectors are proposed every year. In the recent years, a general trend is to aggregate several types of visual features and to use powerful machine learning techniques to learn a classifier. Based on the same idea, using multiple sensors to get more information is common practice in robotics. Ranging sensors are typically used to differentiate obstacles from the navigable space. Three dimensional information is often captured with stereo camera and used for object recognition. GPS and proprioceptive sensors are also considered for localization and egomotion estimation.

A deeper understanding of an image can be achieved with semantic segmentation. Instead of only detecting objects, usually represented by bounding boxes, semantic segmentation aims at classifying every pixels of the image of the scene. Such a task is clearly much more difficult than object detection. Yet, with the availability of more and more training data, features and machine learning algorithms, the performances of semantic segmentation algorithms keep increasing year by year.

In this thesis, we do not aim at introducing new detection algorithms but we investigate the combination of existing methods. The diversity, thus

potential complementarity, of existing detection algorithms makes their combination a challenging and interesting issue. To make the most of existing techniques, one has to find a way to properly fuse all relevant sources of information. Existing detection or semantic segmentation algorithms output different types of imperfect information. This information may be uncertain but also partial.

The theory of probabilities is commonly used to model imperfect information. However, it is often not powerful enough to deal with certain kinds of imperfections such as imprecision or ignorance. Therefore, many theories were developed during the last decades to construct more powerful representations. In this thesis, we focus on the theory of belief functions, also known as Dempster-Shafer theory. It is a fairly simple generalization of the theory of probabilities. It can be used to model different kinds of imperfect information and also to combine information from multiple sources. We consider scene understanding from an information fusion point of view by combining multiple classifiers that may reason with different types of objects and based on data from different kinds of sensors.

This report is structured in five chapters. Chapter 1 presents general aspects of information fusion for scene understanding. A short review of existing methods for object detection and semantic segmentation is provided. Several aspects of the combination of pattern classifiers are then discussed. Finally, the classical Bayesian approach for information fusion is presented. Chapter 2 focuses on the theory of belief functions. We describe how information is represented and combined. Decision making and statistical inference are also presented. In Chapter 3, we propose a general calibration framework that can transform the output of any classifier into belief functions. The binary classifier calibration is first studied then extended to multi-class problems. The calibration is then applied to the combination of pedestrian detectors in Chapter 4. The bounding boxes returned by several detectors are calibrated, clustered and combined. Finally, the fusion framework applied to a multi-modal semantic segmentation problem is described in Chapter 5. The construction of different detection modules is detailed and they are tested on data from the KITTI Vision Benchmark Suite.

Chapter 1

Information fusion for scene understanding

Intelligent vehicles are often equipped with multiple types of sensors. Cameras and LiDAR (Light Detection And Ranging) are the most common ones. Each sensor perceives the surrounding environment differently and can provide its own partial knowledge about it. The aim of multi-modal data fusion is to combine all the information retrieved from the data acquired by the available sensors. In the context of a driver assistance system, where the task is to warn drivers about potential dangers, it seems relevant to display information on an image that reflects what the driver sees.

We consider scene understanding as an image understanding problem with possibly other sources of information given by other sensors. In general, image understanding is a very broad and challenging task. The recent paper of Everingham *et al.* [43], which gives a retrospective on the PASCAL Visual Object Classes (VOC) challenge, provides a thorough view of state-of-art achievements regarding image understanding. The VOC challenge is composed of three types of tasks. The first one is a classification task, where the aim is to predict whether a specific object, among a predefined set of twenty classes, is present in an image or not. The second task is object detection. Here, the objects need to be localized, if present, in the image. The localization is done by using rectangular bounding boxes. The last challenge is to segment the images by classifying each pixel as belonging to one of the twenty potential objects or as background. Figure 1.1 shows some examples from the VOC challenge.

These three tasks are of increasing difficulty. The segmentation task can easily be used for the detection one, which can in turn be used for classification. In the VOC 2012 edition [44], the organizers built a *superclassifier* from the seven methods that were submitted to the classification challenge. The scores returned by the classifiers were concatenated into a single vector and a linear SVM was trained with it. An increase of more

(a) (b) (c)

Figure 1.1: Examples from the PASCAL VOC challenge. (a) Image containing two buses. (b) Ground truth bounding boxes of a bus and a pedestrian. (c) Ground truth segmentation of the two buses pictured in (a).

than 10% in terms of average precision was reported for certain object classes such as "bottle" or "pottedplant". Performance losses were observed for five classes, but they remained relatively limited. The VOC organizers discussed the benefits of combining multiple classifiers in [43]. They also reported some limitations on extending their combination approach to the detection and segmentation tasks. Yet, these two tasks are the only ones that may be of interest for automotive applications, a classification result being of relatively limited use.

In this chapter, we first introduce in Section 1.1 some common approaches and methods to detect objects or segment semantic regions. Then, we discuss general aspects of pattern classifiers combination in Section 1.2. Finally, in Section 1.3 we describe information representation and combination in the framework of classical probability theory.

1.1 Scene understanding

We consider scene understanding as an image understanding problem. It can be decomposed into two main tasks: object detection and semantic segmentation.

1.1.1 Object detection

Object detection is very important for driving safety applications. The detection of pedestrians, vehicles or potentially dangerous moving obstacles is a complex issue. When using cameras, the detections are often represented with rectangular bounding boxes.

Pedestrian detection Pedestrian detection is certainly the most studied and important case. A lot of pedestrian detectors can be found in the literature [40, 54, 33]. These detectors are very diverse and based on different

kinds of features, training data and classifiers. Most methods using monocular camera are based on a sliding window approach. Because the size and the position of pedestrians are unknown, the image is densely scanned over a large set of scales and positions. Then for each window some features are extracted and a classifier is used to decide whether it corresponds to a pedestrian or not.

The use of histograms of oriented gradient (HOG) has been popularized by Dalal and Triggs [20]. Almost all state-of-the-art pedestrian detectors use the HOG feature in some forms. A lot of works have contributed to improve the pioneer work of Dalal and Triggs by integrating new features. The combination of HOG feature with color [119] or motion [19] information has proved to increase the detectors performance. The addition of new information is usually handled by concatenating several feature vectors representing different kinds of information into one *super*-vector.

Using only a single camera is often not enough to achieve satisfying results. The integration of other sensors can provide additional information. Sensors such as stereo camera or LiDAR (Light Detection And Ranging) are commonly used to retrieve 3D information. This information can be used in many different ways. A few attempts to detect pedestrians directly from LiDAR 3D data can be found in the literature [3, 84]. Given the limited resolution of LiDAR sensors, the performances of such approaches remain limited. Another popular method is to use 3D information to retrieve regions of interests (ROI) which are then further analyzed from color images [50, 100]. Geometric cues like the ground plane have also been used to infer constraints for object detection [67, 72]. The use of infra-red images has also been considered in many studies [110, 10]; they can be especially useful for night vision.

Vehicle detection Many algorithms, originally designed for pedestrian detection, can also be adapted to learn models for other kinds of objects such as cars [47, 111]. Contrary to pedestrian detection, there is a high within-class variability in terms of appearance between vehicles. The combination of different models learned from distinct view-points is often useful [65]. In contrast, geometrical analysis of vehicles is easier, as compared to pedestrian detection. Many vehicle detection algorithms using radar-based [93], laser-based [121] or sonar-based [62] active sensors can be found in the literature.

Moving object detection For safety applications, the detection of moving objects or obstacles is often more important than the detection of static ones. Whether they are pedestrians, cars or bicycles, the risk of collision with these moving objects needs to be assessed. The motion of the objects can be estimated through tracking [134]. The use of multiple sources of information can also help the tracking process [94, 140, 41]. Conversely, the

tracking process itself can also help detection [2]. Wedel and Cremers [125] analyzed the 3D motion of a scene using 3D data from a stereo camera system. From two consecutive pairs of stereo images, they managed to estimate the 3D displacement of each pixel. The motion of an object in the image reference can be induced by both its own motion or by the camera motion (egomotion). Therefore, the camera motion needs to be known in order to detect moving objects in the world reference.

The camera motion, which is often rigidly attached to the vehicle, is known if the motion of the vehicle itself is known. The motion of the car can be estimated from its localization. Simultaneous localization and mapping (SLAM) is a very popular way to both estimate the position of the vehicle and detect static structures through a map [38]. Range sensors like LiDAR are commonly used for SLAM. Probabilistic occupancy grid maps [39] can be used to combine maps computed from different sensors. Wang *et al.* [122] added a moving object tracking module within the SLAM framework. Non-probabilistic formulations of occupancy grids have also been considered [80]. The additional use of GPS data, maps [66] or odometry modules [53] can further increase the performance of both localization, mapping and moving object detection.

1.1.2 Semantic segmentation

The detection of some classes, such as roads or buildings, can hardly be represented by bounding boxes. An image annotation at the pixel level is then required. The pixels of the image need to be classified and grouped into segments, i.e., a group of adjacent pixels with the same class. This task is called semantic segmentation, image labeling [68] or image parsing [46, 116]. In several works, the segmentation is only considered for some classes considered as object (*thing*), as opposed to background (*stuff*) [49, 70]. Typically, the ground and buildings are considered as background in the VOC challenge.

For automotive applications, the analysis of the background is also of high interest. In particular, the detection of the road and the navigable space is primordial for autonomous driving systems. In [57], Hoiem *et al.* recovered the 3D layout of a scene from a single image by separating the ground from the vertical structures and the sky. Several monocular image semantic segmentation frameworks have been proposed [106, 115, 46]. Ladicky *et al.* [69] extended the method of Shotton *et al.* [106] by including depth information from a stereo camera. Given the high diversity of the classes that need to be classified, an extensive set of features is often considered. These features usually encode local information, such as texture, which are well adapted to describe *stuff*. However, for object detection, they are not as powerful as more global features such as those used in sliding windows based methods.

Leibe *et al.* [73] used an object detection module to initialize object segmentation. Ladicky *et al.* [68] used the bounding boxes returned by an object detector to compute high order cliques within a conditional random field (CRF) formulation. The segments included in the bounding boxes are led to belong to similar classes. Tighe and Lazebnik [116] used per-exemplar detectors to get object segmentation from a set of segmented training data. Conversely, many detection methods submitted to VOC use semantic segmentation to get initial ROI.

Semantic segmentation can also be computed from 3D information [130, 82]. Occupancy grid maps computed from a SLAM module can, for example, easily be used to segment out the obstacle-free space [5].

1.2 Combining pattern classifiers

Many of the methods presented in the previous section combine several sources of information to attain better performance, given a specific task. In this thesis, we aim at combining directly several detection modules. Several issues need to be solved to combine different types of pattern classifiers.

1.2.1 Types of outputs

There are many possible ways to combine multiple pattern classifiers, which highly depend on the outputs of the classifiers. Xu *et al.* [131] described three principal types of outputs (see also [64, Chapter 4.1]):

- *Type 1 (The Abstract level).* The classifier returns a unique predicted class label without any confidence measure.
- *Type 2 (The Rank level).* The classifier returns a ranking of all, or the most plausible, class labels.
- *Type 3 (The Measurement level).* The classifier assigns a confidence measure, or score, to each class label.

It is clear that the third type of outputs is the most informative one. It can easily generate outputs of type 2, which in turn can lead to outputs of type 1. Outputs at the abstract level are typically provided by black box commercial tools such as MobilEye®[1] products [78]. The rank level case is rarely seen in studies dealing with intelligent vehicles. Classical classifiers, used in the machine learning community, can usually provide outputs at the measurement level. In this thesis, we will mainly deal with this last type of outputs.

[1] MobilEye products use cameras for collision warning, they can detect pedestrians as well as lane markers.

1.2.2 Trainable and non-trainable combiners

Given the classification outputs of several classifiers, the combination strategies can be separated into two kinds: trainable and non-trainable [37, 117]. In the first case, the outputs of each classifier are used as inputs in a new round of training. A simple way is to concatenate the initial outputs into a feature vector and train a new classifier using classical machine learning techniques. Using trainable approaches is appealing as the outputs of the initial classifiers can be directly supplied to a new classifier without the need of any particular processing. However, the re-training step may also be seen as a major drawback, as new classifiers can hardly be added afterward. In many practical situations, new sources of information can become available over time: new sensors can be included in the system or additional training data can be acquired. In such cases, we may not wish to train the whole combiner again every time.

The non-trainable approaches consist in combining directly the outputs of the base classifiers, using a pre-defined combination rule. The simplest example is the majority vote. It is one of the few cases where the outputs of the base classifiers do not have to be pre-processed. Otherwise, the various outputs have to be made comparable beforehand. This constitutes the major difficulty of non-trainable methods. Class membership probabilities are often used as common representations. The transformation of the outputs of a classifier into probabilities is referred to as calibration. In non-trainable approaches, the quality of the calibration often prevails over the classification accuracy of the initial classifiers [8, 37]. Compared to trainable combiners, non-trainable methods may lead to sub-optimal results, as the potential interactions among the initial classifiers are not considered. However, in contrast with the trainable case, new classifiers can be easily added and combined iteratively.

This last point is a prerequisite for a flexible multi-modal system. Therefore, only non-trainable combination approaches are considered in this thesis.

1.2.3 Class definition

The definition of the classes of objects in an image understanding context is non-trivial. Having an exhaustive list of the objects that may appear in an image may seem impossible in practice. All the semantic segmentation methods presented in the previous section deal with different sets of classes. In our case, we will only consider classes that are of interest in automotive applications. Figure 1.2 shows the set of 30 basic classes used for our ground truth annotations. Figure 1.3 shows some images we have selected and manually annotated from the KITTI Vision Benchmark Suite [51].

In our case, we do not expect to build a general classifier that would

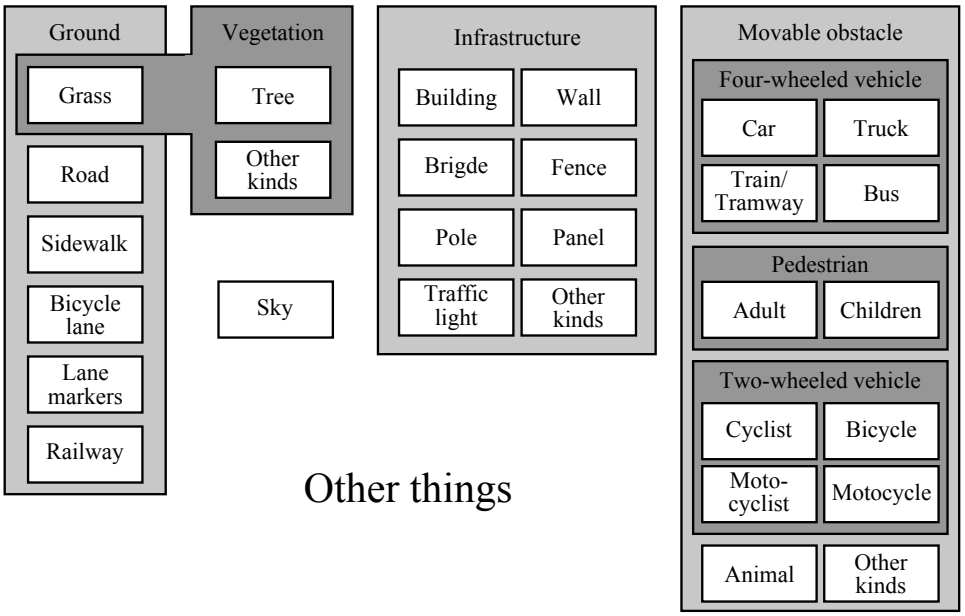


Figure 1.2: Class decomposition.

recognize each of these basic classes. Instead, we try to combine methods that may recognize only a partial number of classes or sets of classes. This set of basic classes is also allowed to evolve over time if new classes need to be introduced.

1.3 Bayesian fusion

The use of probabilistic measures is the most common way to model imperfect information. Let $\Omega = \{\omega_1, \ldots, \omega_K\}$ be a set of K mutually exclusive classes called the *frame of discernment*, which corresponds to the set of all possible classes. The imperfect knowledge about the true class $\omega \in \Omega$ of an instance, after observing some data $\mathbf{x} \in \mathbb{X}$, is modeled by an a posteriori probability distribution $P(\cdot|\mathbf{x})$ defined over Ω. This probability distribution will also be noted $P^{\Omega}_{\mathbf{x}}(\cdot)$, the superscript Ω will sometimes be omitted if there is no ambiguity about the frame of discerment.

1.3.1 Information representation

After observing some data $\mathbf{x} \in \mathbb{X}$, the probability $P^{\Omega}_{\mathbf{x}}(\omega_i)$ can be interpreted as the confidence degree of the class $\omega_i \in \Omega$.

Definition 1.1. *If there exists a singleton $\omega_j \in \Omega$ such that $P^{\Omega}_{\mathbf{x}}(\omega_j) = 1$, the information is said to be* certain. *Otherwise, it is* uncertain.

Figure 1.3: Examples of annotated images from the KITTI dataset.

The closer the probability distribution is to the uniform distribution, the less informative it is. In particular, complete ignorance is handled by the principle of indifference [60, Chapter IV, pages 41–64]:

> "The Principle of Indifference asserts that if there is no *known* reason for predicating of our subject one rather than another of several alternatives, then relatively to such knowledge the assertions of each of these alternatives have an *equal* probability. Thus equal probabilities must be assigned to each of several arguments, if there is an absence of positive ground for assigning *unequal* ones".

Ignorance can occur when the data $\mathbf{x}$ conveys no relevant information or when is it known to be unreliable. In this case, the uniform distribution U^{Ω} over Ω is used

$$U^{\Omega}(\omega_i) = \frac{1}{|\Omega|}, \qquad \forall \omega_i \in \Omega. \tag{1.1}$$

1.3.2 Combination rules

Probabilistic fusion relies mainly on Bayes' rule. Let $P(\omega_i|\mathbf{x_1})$ and $P(\omega_i|\mathbf{x_2})$, for $i = 1, \ldots, K$, be the probability distributions over Ω returned by two modules after observing some data $\mathbf{x_1} \in \mathbb{X}$ and $\mathbf{x_2} \in \mathbb{X}$, respectively. By assuming conditional independence, the following expression holds:

$$p(\mathbf{x_1}, \mathbf{x_2}|\omega_i) \;=\; p(\mathbf{x_1}|\omega_i)p(\mathbf{x_2}|\omega_i), \quad \forall i \in \{1, \ldots, K\}. \tag{1.2}$$

Bayes' rule then yields

$$\begin{aligned} P(\omega_i|\mathbf{x_1}, \mathbf{x_2}) &= \frac{P(\omega_i)p(\mathbf{x_1}, \mathbf{x_2}|\omega_i)}{p(\mathbf{x_1}, \mathbf{x_2})} && \text{(1.3a)} \\ &= \frac{P(\omega_i)}{p(\mathbf{x_1}, \mathbf{x_2})} p(\mathbf{x_1}|\omega_i)p(\mathbf{x_2}|\omega_i) && \text{(1.3b)} \\ &= \frac{p(\mathbf{x_1})p(\mathbf{x_2})}{p(\mathbf{x_1}, \mathbf{x_2})} \frac{P(\omega_i|\mathbf{x_1})P(\omega_i|\mathbf{x_2})}{P(\omega_i)} && \text{(1.3c)} \\ &\propto \frac{P(\omega_i|\mathbf{x_1})P(\omega_i|\mathbf{x_2})}{P(\omega_i)}, \; \forall i \in \{1, \ldots, K\}. && \text{(1.3d)} \end{aligned}$$

In practice, the prior class distribution $P(\omega_i)$ is difficult to estimate and is often replaced by a uniform distribution. When a uniform prior class distribution is used, the combination rule (1.3) is referred to as the product rule. Other combination rules that replace the product by the sum, the minimum or the maximum operator can be derived from the product rule by using different approximations [63]. These four combination rules will be denoted, respectively, by the operators $\boxtimes$, $\boxplus$, $\wedge$ and $\vee$. The support μ_i of the class ω_i computed from these combination rules, given L probabilities

$P(\omega_i|\mathbf{x_1}), \ldots, P(\omega_i|\mathbf{x_L})$, is summarized in Table 1.1. To retrieve a probability distribution from these supports, they are normalized to sum up to one. The combination rule resulting from the sum operator is actually the average rule.

To illustrate the differences between these rules, let us consider a binary classification problem with $\Omega = \{0, 1\}$. Figure 1.4 shows the combined probability $P(y = 1|\mathbf{x_1}, \mathbf{x_2})$ computed from two probabilities $P(y = 1|\mathbf{x_1})$ and $P(y = 1|\mathbf{x_2})$, where $y \in \{0, 1\}$ is the true class of an observed instance. The average rule is sometimes considered to be more *stable* [113] than the product rule. Indeed, we can see in Figure 1.4b that the average rule is less prone to drive the combined probability to a very low or very high value. For the average rule, a high probability can only be obtained if both the initial probabilities are high. On the contrary, with the product rule, a very low probability can still lead to a very high combined probability. For example, the combination of $P_{\mathbf{x_1}}(y = 1) = 0.1$ with $P_{\mathbf{x_2}}(y = 1) = 0.99$ gives $P_{\mathbf{x_1}} \boxtimes P_{\mathbf{x_2}}(y = 1) \approx 0.917$, while $P_{\mathbf{x_1}} \boxplus P_{\mathbf{x_2}}(y = 1) = 0.545$.

Definition 1.2. *Let P be a binary probability distribution over a binary random variable Y defined as $P(Y = 1) = q$ and $P(Y = 0) = 1 - q$ for some $q \in [0, 1]$. The entropy $H(P)$ associated to P defined as*

$$H(P) \quad = \quad -q \ln q - (1 - q) \ln(1 - q), \tag{1.4}$$

quantifies the amount of information encoded by P. The higher is the entropy, the less informative is P. In particular, the entropy is maximized by the uniform distribution.

Proposition 1.1. *Let $P_{\mathbf{x_1}}$ and $P_{\mathbf{x_2}}$ be two binary probability distributions. Their combination follows the following ordering:*

$$H(P_{\mathbf{x_1}} \boxtimes P_{\mathbf{x_2}}) \leq H(P_{\mathbf{x_1}} \wedge P_{\mathbf{x_2}}) \leq H(P_{\mathbf{x_1}} \boxplus P_{\mathbf{x_2}}) \leq H(P_{\mathbf{x_1}} \vee P_{\mathbf{x_2}}). \tag{1.5}$$

Proof. The proof is given in Appendix (see page 107).

The product rule can be seen as more committed than the other three while the maximum rule is the most cautious one. In practical situations, Tax *et al.* [113] recommended to use averaging estimated posterior probabilities when the posterior probabilities to combine are not well estimated. They would prefer the product combination rule only when good estimates of posterior class probabilities are available.

Another interesting way to visualize the differences between these combination rules is to look at the gain $P_{\mathbf{x_1},\mathbf{x_2}} - P_{\mathbf{x_1}}$ obtained after combining $P_{\mathbf{x_1}}$ with $P_{\mathbf{x_2}}$. Figure 1.5 shows the gains obtained from the different combination rules. We can remark that the uniform distribution is, as one would expect, a neutral element of the product rule. However, it is not the case for the three other rules; none of them actually has a neutral element. This

Table 1.1: Probabilistic combination rules.

Combination rules	Support of the class ω_i
Product	$\mu_i^* = \prod_{j=1}^{L} P(\omega_i \mid \mathbf{x_j})$
Average	$\mu_i^+ = \sum_{j=1}^{L} P(\omega_i \mid \mathbf{x_j})$
Minimum	$\mu_i^\wedge = \min_{j=1,\dots,L} \{P(\omega_i \mid \mathbf{x_j})\}$
Maximum	$\mu_i^\vee = \max_{j=1,\dots,L} \{P(\omega_i \mid \mathbf{x_j})\}$

result is counter-intuitive as a uniform distribution should convey no information. In a classification context, this result might be acceptable for the average rule as the ranking of the potential outcomes, with respect to their respective estimated probability, is not changed by combining with an uniform distribution. This is not the case for the minimum and maximum rules when the classification involves three or more classes.

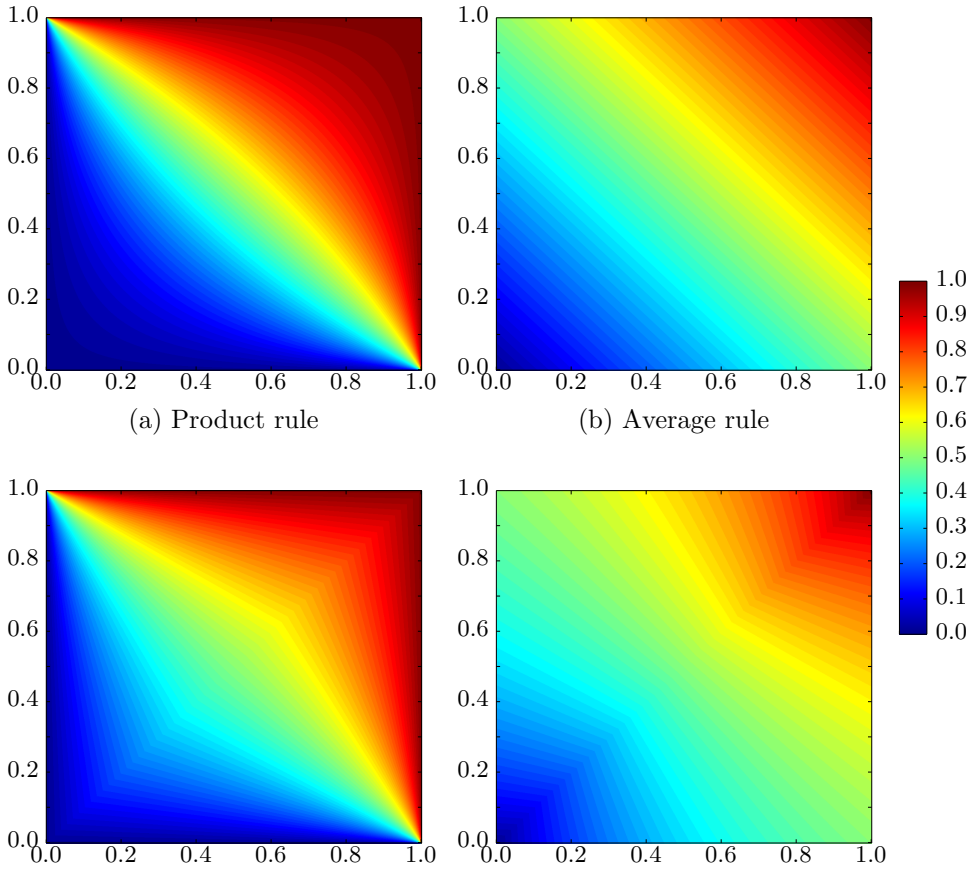
Another difference is that the average, minimum and maximum rules are idempotent. It means that a probability distribution combined with itself leads to the same probability. It is not true for the product rule. When the conditional independence property (1.2) is not satisfied, it is more cautious not to use the product rule.

Next, we can see that the gain obtained by the average and maximum rules is bounded in $[-0.5, +0.5]$. This implies that certain information cannot be well represented. A probability $P(y = 1) = 1$, for which there is no uncertainty involved, may still be changed if combined with another probability distribution with the average or maximum rules. Using the product or minimum rules, such a probability distribution is not modified as it is considered certain.

Finally, it is important to note that the product and minimum rules cannot be used to combine contradictory sources of information. Two probability distributions are contradictory if the support resulting from their combination is null for all the classes. Table 1.2 summarizes the properties of these four combination rules.

1.3.3 Reliability

When the reliability $r \in \{0, 1\}$ of the source of information is known, it can be combined with the initial probability distribution $P_{\mathbf{x}}^{\Omega}$. If the source of information is reliable, i.e., $r = 1$, then $P_{\mathbf{x}}^{\Omega}$ is kept as is, otherwise it conveys no information and should be replaced by U^{Ω}. The combined probability



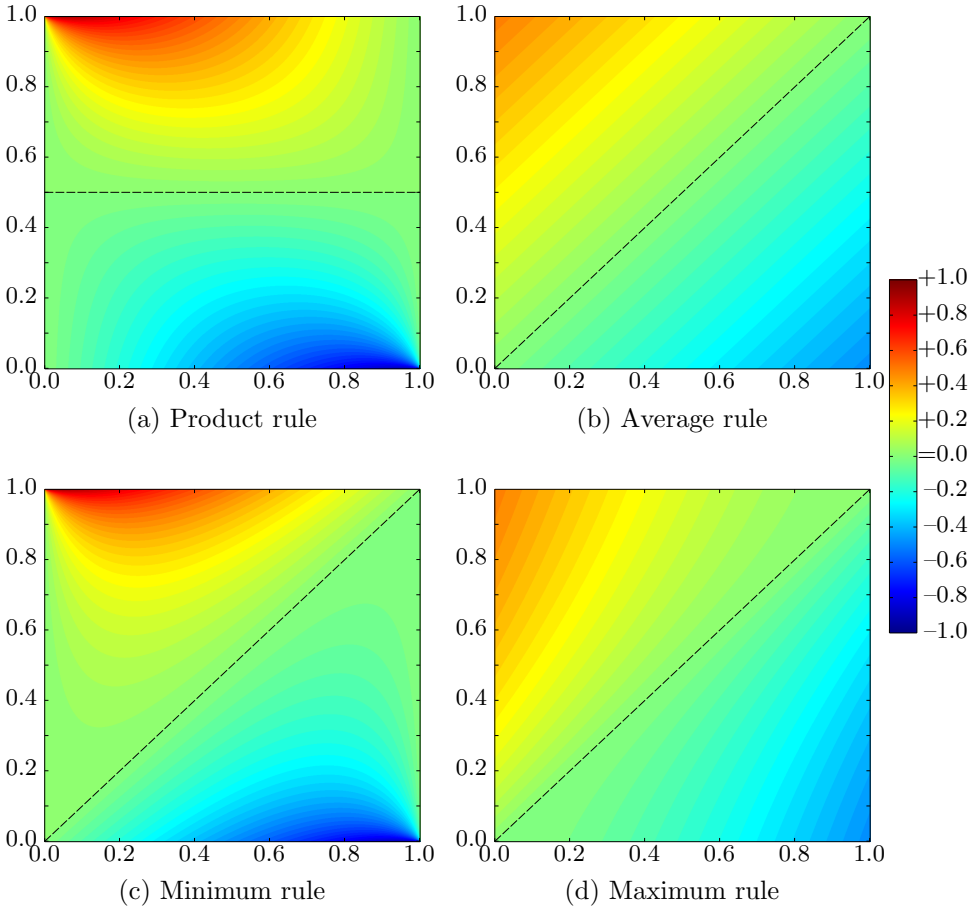
(a) Product rule (b) Average rule

(c) Minimum rule (d) Maximum rule

Figure 1.4: Combination of two probability distributions. The horizontal axis corresponds to the value of $P(y = 1|\mathbf{x_1})$ and the vertical axis corresponds to the value of $P(y = 1|\mathbf{x_2})$. The colors show the value of the combined probability $P(y = 1|\mathbf{x_1}, \mathbf{x_2})$.

Table 1.2: Properties of combination rules.

Properties	Prod.	Avg.	Min.	Max.
Uniform distribution as neutral element	√			
Representation categorical information	√		√	
Idempotence		√	√	√
Combination of contradictory information		√		√



(a) Product rule (b) Average rule

(c) Minimum rule (d) Maximum rule

Figure 1.5: Probability gain obtained from the combination. The horizontal axis corresponds to the value of $P(y = 1|\mathbf{x_1})$ and the vertical axis corresponds to the value of $P(y = 1|\mathbf{x_2})$. The colors show the value of the gain $G = P(y = 1|\mathbf{x_1}, \mathbf{x_2}) - P(y = 1|\mathbf{x_1})$. The dotted line separates the positive gains from the negative ones.

$P^{\Omega}_{\mathbf{x},r}$ is derived from the law of total probability

$$P^{\Omega}_{\mathbf{x},r}(\omega_i) = P^{\Omega}_{\mathbf{x}}(\omega_i|r=1)P_R(r=1) + P^{\Omega}_{\mathbf{x}}(\omega_i|r=0)P_R(r=0) \tag{1.6a}$$

$$= P^{\Omega}_{\mathbf{x}}(\omega_i)P_R(r=1) + U^{\Omega}(\omega_i)P_R(r=0), \tag{1.6b}$$

$$= P^{\Omega}_{\mathbf{x}}(\omega_i)P_R(r=1) + \frac{1-P_R(r=1)}{|\Omega|}, \tag{1.6c}$$

for all $\omega_i \in \Omega$. If the probability of a source of information to be reliable is low, i.e., $P_R(r=1) \approx 0$, then $P^{\Omega}_{\mathbf{x},r} \approx U^{\Omega}$. Combining $P_{\mathbf{x},r}$ with another probability distribution using the product rule will then lead to only small changes, as expected. However, it is not the case for the other combination rules. For the average and maximum rules, the probability $P_R(r=1)$ is often used as a weight w for computing the support function. The support function of the weighted average rule is defined as

$$\tilde{\mu}^{+}_i = \sum_{j=1}^{L} w_j P(\omega_i|\mathbf{x_j}), \quad \forall i \in \{1,\ldots,K\}, \tag{1.7}$$

where $w_j = P_R(r_j = 1)$ represents the reliability of j-th source of information. Similarly, the support function of the weighted maximum rule is defined as

$$\tilde{\mu}^{\vee}_i = \max_{j=1,\ldots L}\{w_j P(\omega_i|\mathbf{x_j})\}, \quad \forall i \in \{1,\ldots,K\}. \tag{1.8}$$

For the minimum rule, there is no straightforward way to take into account the reliability of a source of information.

1.3.4 Refinement

An important point to note is that probability distributions to be combined, have to be defined over the same frame of discernment. When several modules deal with different kinds of objects, it is necessary to reason with several frames of discernment with varying granularities. From a frame of discernment Ω, a refinement Θ can be defined by splitting some or all its elements into new classes.

Definition 1.3. *A refining from Ω to Θ can be defined [105, Chapter 6, Section1] by an application $\rho : 2^{\Omega} \to 2^{\Theta}$ such that*

- $\{\rho(\{\omega\}),\ \omega \in \Omega\} \subseteq 2^{\Theta}$ *is a partition of Θ;* (1.9a)
- $\forall A \subseteq \Omega,\ \rho(A) = \bigcup_{\omega \in A} \rho(\{\omega\})$. (1.9b)

The notation 2^{Ω} refers to the power set of Ω, which is the set of all subsets of Ω. Condition (1.9b) implies that the refining ρ is fully defined by the images of all the singletons $\{\omega\} \in 2^{\Omega}$ under ρ.

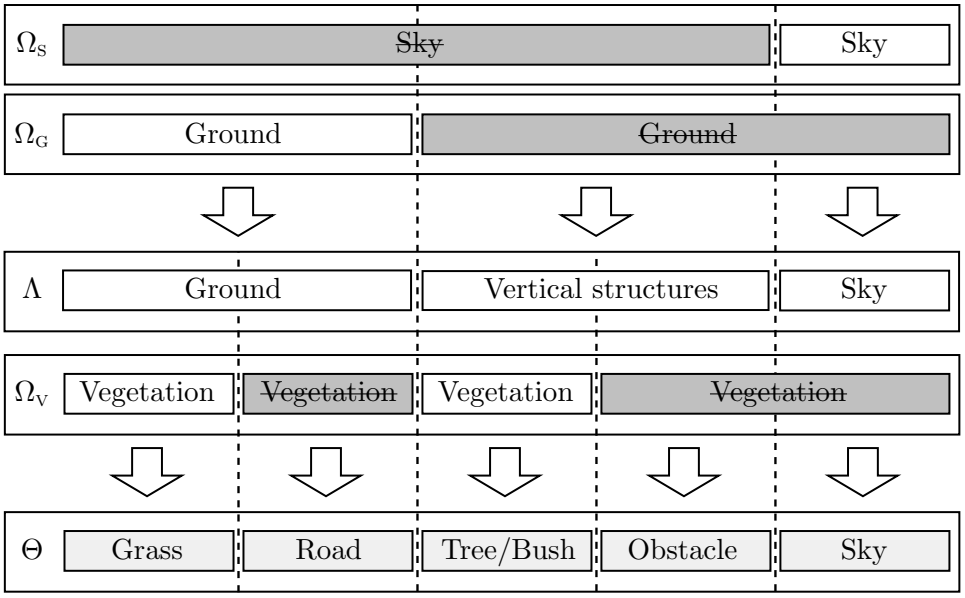


Figure 1.6: Illustration of multi-class fusion. A ground detector can be combined with a sky detector by defining the "vertical" class which refers to anything that is not the ground or the sky. The combination with a vegetation detector leads to an even finer class decomposition. The "obstacle" class refers to anything that is neither the sky, the ground nor vegetation.

Example 1. Figure 1.6 shows a typical example of a combination of detectors defined over distinct frames of discernment. This example will be used throughout this thesis and will serve as basis to show the limitations of probabilistic reasoning. If a ground detector reasoning over $\Omega_{\text{G}} = \{\text{ground}, \sout{\text{ground}}\}$ has to be combined with a sky detector reasoning over $\Omega_{\text{S}} = \{\text{sky}, \sout{\text{sky}}\}$, a common frame of discernment $\Lambda = \{\text{ground}, \text{vertical}, \text{sky}\}$ has to be defined, as illustrated in Figure 1.6. The refining from Ω_{G} to Λ is defined by

$$\left\{ \begin{array}{rcl} \rho(\{\text{ground}\}) & = & \{\text{ground}\}, \\ \rho(\{\sout{\text{ground}}\}) & = & \{\text{vertical, sky}\}. \end{array} \right. \tag{1.10}$$

The notation $\{\sout{\text{ground}}\}$ is used instead of $\overline{\{\text{ground}\}}$ whenever we want to specifically refer to the non-ground class as a singleton, but they both semantically refer to the same thing. The class "vertical" actually corresponds to everything that is neither the ground nor the sky, *i.e.*, $\{\text{vertical}\} = \overline{\{\text{ground}\}} \cap \overline{\{\text{sky}\}}$. Then, if a vegetation detector reasoning over $\Omega_{\text{V}} = \{\text{vegetation}, \sout{\text{vegetation}}\}$ has to be added, the frame Λ can be further refined to Θ by

$$\left\{ \begin{array}{rcl} \rho'(\{\text{ground}\}) & = & \{\text{grass, road}\}, \\ \rho'(\{\text{vertical}\}) & = & \{\text{tree, obstacle}\}, \\ \rho'(\{\text{sky}\}) & = & \{\text{sky}\}. \end{array} \right. \tag{1.11}$$

The "obstacle" class refers to anything that is neither the sky, the ground nor vegetation.

An important type of imperfection that occurs when dealing with refinements is imprecise information. Consider the case described in Example 1 and assume that the output of a ground detector, initially defined on Ω_{G}, is expressed in the refined frame of discernment Λ. Let $\mathbf{x}_{\text{G}} \in \mathbb{X}$ be some observed data and $P^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}$ be the probabilities returned by a ground detector defined as

$$P^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}(\text{ground}) = q, \qquad P^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}(\sout{\text{ground}}) = 1 - q, \tag{1.12}$$

where $q \in [0, 1]$. The information represented by $P^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}$ can be rewritten over the refined frame Λ as

$$P^{\Lambda}_{\mathbf{x}_{\text{G}}}(\text{ground}) = q, \qquad P^{\Lambda}_{\mathbf{x}_{\text{G}}}(\text{vertical OR sky}) = 1 - q. \tag{1.13}$$

However, expression (1.13) does not fully define the probability $P^{\Lambda}_{\mathbf{x}_{\text{G}}}$. Actually, every probability distribution P so that $P(\text{vertival}) + P(\text{sky}) = 1 - q$, verifies the constraints defined by (1.13). We say that the information represented by (1.13) is *imprecise* [120]. In such situations, the principle of indifference states that the classes "vertical" and "sky" should have an equal probability. It leads to the following probability:

$$P^{\Lambda}_{\mathbf{x}_{\text{G}}}(\text{ground}) = q, \qquad P^{\Lambda}_{\mathbf{x}_{\text{G}}}(\text{vertical}) = P^{\Lambda}_{\mathbf{x}_{\text{G}}}(\text{sky}) = \frac{1 - q}{2}. \tag{1.14}$$

As the ground detector cannot differentiate the "vertical" class from the "sky" class, the initial probability assigned the non-ground class is uniformly distributed to these two refined classes.

One major issue with such an approach is that the information represented by (1.14) is not exactly the same as (1.12). Suppose that the observation $\mathbf{x}_{\mathrm{G}}$ conveys no relevant or reliable information. The initial probability $P_{\mathbf{x}_{\mathrm{G}}}^{\Omega_{\mathrm{G}}}$ should be uniform ($q = 1/2$):

$$P_{\mathbf{x}_{\mathrm{G}}}^{\Omega_{\mathrm{G}}}(\text{ground}) = 1/2, \qquad P_{\mathbf{x}_{\mathrm{G}}}^{\Omega_{\mathrm{G}}}(\cancel{\text{ground}}) = 1/2, \tag{1.15}$$

in which case the ground detector cannot make any decision, as expected. Reasoning on another frame of discernment such as Λ does not change the information at hand, which should still be modeled by a uniform distribution. However, equation (1.14) does not define a uniform distribution but yields

$$P_{\mathbf{x}_{\mathrm{G}}}^{\Lambda}(\text{ground}) = 1/2, \qquad P_{\mathbf{x}_{\mathrm{G}}}^{\Lambda}(\text{vertical}) = P_{\mathbf{x}_{\mathrm{G}}}^{\Lambda}(\text{sky}) = 1/4. \tag{1.16}$$

Even worse, the ground detector would then be able to make a decision and choose the "ground" class as the most probable one. Paradoxically, if instead of $\{\cancel{\text{ground}}\}$, the class $\{\text{ground}\}$ had been refined into $\{\text{grass, road}\}$, the same reasoning over $\Psi = \{\text{grass, road, } \cancel{\text{ground}}\}$ leads to

$$P_{\mathbf{x}_{\mathrm{G}}}^{\Lambda}(\cancel{\text{ground}}) = 1/2, \qquad P_{\mathbf{x}_{\mathrm{G}}}^{\Lambda}(\text{grass}) = P_{\mathbf{x}_{\mathrm{G}}}^{\Lambda}(\text{road}) = 1/4. \tag{1.17}$$

The same ground detector would then chose the "non-ground" class as the most probable one. This example shows that traditional probability theory cannot properly represent imprecise information.

1.4 Conclusion

In this thesis, scene understanding is seen as an image understanding problem. Multiple sources of information from potentially different sensors are considered for both object detection and semantic segmentation. In order to be flexible enough to include new sources of information, only non-trainable combiners are considered. In such a context, all the classification modules need to provide comparable outputs. Probabilistic measures are commonly employed to represent imperfect information. However, many types of imperfect information cannot be properly represented with probability distributions. More complex theories have been developed during the last decades to better represent and combine imperfect information. One of them will be reviewed in the next chapter.

Chapter 2

Theory of belief functions

Multi-modal information fusion tasks require powerful tools to represent and combine several types of imperfect information. As shown in the previous chapter, probabilistic approaches, even though commonly employed, may be too limited. The theory of belief functions, also known as Dempster-Shafer theory [105], evidence theory or the transferable belief model [109], is a generalization of the theory of probabilities. It offers a well-founded and elegant framework to represent and combine a large variety of imperfect information [61]. It is also a generalization of possibility theory [137] and is closely linked to other theories including random sets [85], imprecise probability [120] and fuzzy sets [136].

In this chapter, we first describe in Section 2.1 how different types of information can be represented in the framework of belief functions. In Section 2.2, we discuss the combination of mass functions and present a few combination rules. Next, we describe in Section 2.3 some operations over the frame of discernment such as refinement, coarsening and conditioning. Then, in Section 2.4, we consider the issue of decision making using belief functions. Finally, statistical inference based on belief functions is described in Section 2.5.

2.1 Information representation

2.1.1 Mass function

Definition 2.1. *A* mass function*, also called* basic belief assignment *or* basic probability assignment*, over a frame of discernment* Ω *is a function* $m^\Omega : 2^\Omega \to [0, 1]$ *verifying*

$$m^\Omega(\emptyset) = 0, \qquad \sum_{A \subseteq \Omega} m^\Omega(A) = 1. \tag{2.1}$$

The superscript Ω will sometimes be omitted when there is no ambiguity about the frame of discernment. Given an object of class $\omega \in \Omega$, the belief

about its membership to some subsets of Ω can be modeled by a mass function m. The quantity $m(A)$, for a given subset $A \subseteq \Omega$, represents the belief committed exactly to the hypothesis $\omega \in A$. It is important to understand that the hypothesis $\omega \in A$ does not support the membership of ω to any subset $B \subsetneq A$. If $m(A) > 0$, then A is said to be a *focal element*, or *focal set*, of m.

Definition 2.1 imposes that the empty set cannot be a focal element. If this constraint is relaxed, the mass function is said to be *unnormalized*. Then, the quantity $m(\emptyset)$ can be interpreted as the degree of support of the hypothesis that the true class ω is actually outside of the frame Ω. In this thesis, the closed-world assumption is used [107], i.e., the frame of discernment Ω is considered exhaustive. This might seem contradictory with our discussion about the difficulty of having an exhaustive list of objects in an image understanding context (Section 1.2.3). However, because the reasoning is performed over sets of classes, it is easy to define the complement $\overline{A}$ of any set A.

For instance, in the example given in Figure 1.6, the frame of discernment $\Omega_{\text{G}} = \{\text{ground}, \sout{\text{ground}}\}$ is exhaustive as every observed object is either ground or not. Similarly, the class "vertical structures" in $\Lambda = \{\text{ground}, \text{vertical}, \text{sky}\}$ actually refers to anything that is not the ground or the sky. Therefore, only *normalized* mass function will be used in this work.

Definition 2.2. *A mass function that has Ω as unique focal element, i.e., $m(\Omega) = 1$, is said to be* vacuous.

The vacuous mass function actually represents total ignorance as the closed-world assumption implies that the hypothesis $\omega \in \Omega$ is always true.

Definition 2.3. *A non-vacuous mass function that has only one focal element is called a* categorical *mass function.*

A categorical mass function represents certain information in which no uncertainty is involved.

Definition 2.4. *A probability distribution over Ω is a particular kind of mass function that has only singletons as focal elements. Such a mass function is said to be* Bayesian.

In the theory of belief functions, probability distributions are used to represent precise information.

Definition 2.5. *A* simple *mass function is a mass function that has at most two focal elements, including Ω. For $A \subset \Omega$ and $w \in [0, 1]$, the notation A^w refers to the simple mass function m defined as*

$$m(A) = 1 - w, \qquad m(\Omega) = w. \tag{2.2}$$

Table 2.1: Examples of mass functions defined over $\Lambda = \{\text{ground, vertical, sky}\}$.

Mass function	Example	Type of information
Vacuous	$m(\Lambda) = 1$	Complete ignorance
Categorical	$m(\{\text{vertical, sky}\}) = 1$	Certain information
Bayesian	$m(\{\text{ground}\}) = 1/2$, $m(\{\text{vertical}\}) = m(\{\text{sky}\}) = 1/4$	Precise information
Simple	$m(\{\text{ground}\}) = 2/3$, $m(\Lambda) = 1/3$	Support a unique hypothesis
Dogmatic	$m(\{\text{ground}\}) = 1/3$, $m(\{\text{vertical, sky}\}) = 2/3$	

Simple mass functions represent evidences that support only one hypothesis. The mass function A^0 is categorical, while A^1 is vacuous for any $A \subset \Omega$.

Definition 2.6. *A mass function is said to be* dogmatic *if Ω is not a focal element. In particular, categorical and Bayesian mass functions are dogmatic.*

The use of dogmatic mass functions should be handled with caution as they imply strong assumption. Let $\mathcal{F} = \{F_i \subset \Omega | 1 \leq i \leq k\}$ be the set of all focal elements of a dogmatic mass function m. Let $\overline{\mathcal{F}} = \{E \subseteq \Omega | \forall i \in [|1, k|],\ E \cap F_i = \emptyset\}$ be the set of all subsets $E \subset \Omega$ than intersect none of the elements of $\mathcal{F}$. Then, for all subsets $E \in \overline{\mathcal{F}}$, the hypothesis E is completely ruled out by m. When the use is not totally justified, dogmatic mass functions may lead to somewhat counter-intuitive results when combined with conflicting information.

Table 2.1 gives some examples of mass functions and the type of information that are encoded.

2.1.2 Other representations

The information encoded by a mass function can be represented in other ways. The notions of belief, plausibility, commonality, contour and pignistic probability play important roles in many aspects of evidential reasoning.

Definition 2.7. *Belief and plausibility functions are defined, respectively, as*

$$Bel(A) = \sum_{B \subseteq A} m(B) \quad \textit{and} \quad Pl(A) = \sum_{B \cap A \neq \emptyset} m(B), \tag{2.3}$$

for all $A \subseteq \Omega$. The quantity $Bel(A)$ measures the degree of support of A, while $Pl(A) = 1 - Bel(\overline{A})$ measures the lack of support to the complement of A.

Definition 2.8. *The commonality function q associated to a mass function m is defined as*

$$q(A) = \sum_{B \supseteq A} m(B), \quad \forall A \subseteq \Omega. \tag{2.4}$$

The quantity $q(A)$ can be interpreted as the degree of belief that could potentially support any element of A if further information becomes available.

There exists a one-to-one correspondence between mass, belief, plausibility and commonality functions. The mass function can be computed from the belief, plausibility and commonality functions as

$$m(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} Bel(B) \tag{2.5a}$$

$$= \sum_{B \subseteq A} (-1)^{|A \setminus B|} \left(1 - Pl\left(\overline{B}\right)\right) \tag{2.5b}$$

$$= \sum_{B \supseteq A} (-1)^{|A \setminus B|} q(B). \tag{2.5c}$$

Definition 2.9. *The contour function $pl : \Omega \to [0, 1]$ associated to a mass function is defined as the plausibility of the singletons*

$$pl(\omega) = Pl(\{\omega\}), \quad \forall \omega \in \Omega. \tag{2.6}$$

Definition 2.10. *A mass function can be transformed into a probability distribution $BetP$ by the pignistic transformation [109] defined as*

$$BetP(\omega) = \sum_{A \subseteq \Omega, \omega \in A} \frac{m(A)}{|A|}, \quad \forall \omega \in \Omega. \tag{2.7}$$

$BetP$ is called the pignistic probability associated to m. The mass $m(A)$ is uniformly distributed to all of the elements of A.

The contour function and the pignistic probability are often used for decision making. This issue is further discussed in Section 2.4.

2.1.3 Consonant belief functions

Definition 2.11. *A mass function is said to be* consonant *if its focal elements $\mathcal{F} = \{F_i \subseteq \Omega | 1 \leq i \leq k\}$, are nested: $F_1 \subset F_2 \subset \ldots \subset F_k$.*

Consonant mass functions play an important role in the theory of belief functions. In particular, the plausibility of consonant mass function defines a possibility distribution [137]. This makes the theory of belief functions a generalization of the possibility theory.

Proposition 2.1. *Let m be a mass function and Pl its associated plausibility function, the following propositions are equivalent:*

1. *m is consonant;*
2. *$Pl(A \cup B) = \max(Pl(A), Pl(B))$, $\forall A, B \subseteq \Omega$;*
3. *$Pl(A) = \max_{\omega \in A} Pl(\{\omega\})$, $\forall A \subseteq \Omega, A \neq \emptyset$.*

From a more philosophical point of view, one may argue that the information induced by a single piece of evidence should always be modeled by a consonant mass function. The presence of non consonant mass functions is then only justified when combining several sources of information. In [105], Shafer cited the economist Shackle who argued in [104] that:

> "To assign greater than zero degrees of potential surprise to both the hypothesis and its contradictory would [...] betray an unresolved mental confusion."

The notion of potential surprise is to be understood as belief in the theory of belief functions.

In practice, consonant mass functions are also useful because they have a maximum of $|\Omega|$ focal elements instead of $2^{|\Omega|}$ in the general case. They can be used as approximations of general mass functions [18, 35]. Consonant mass functions also appear when considering belief functions ordering. In the theory of belief functions, the *Least Commitment Principle* [108] plays a role similar to the principle of indifference in probability theory. This principle indicates that, given two belief functions compatible with a set of constraints, the least informative, with respect to a given ordering, should be selected. A popular partial ordering is the q-ordering [34].

Definition 2.12. *A mass function m_1 is said to be q-less committed than m_2 and noted $m_1 \sqsubseteq_q m_2$ if and only if*

$$q_1(A) \leq q_2(A), \quad \forall A \subseteq \Omega. \tag{2.8}$$

This least commitment principle can be used to associate a non-Bayesian mass function to a probability distribution. The pignistic transformation given in Definition 2.10 transforms a mass function into a probability distribution. This transformation is, however, not invertible: different mass functions can lead to the same pignistic probability. A generalized inverse can however be defined by using the least commitment principle. Given a probability distribution P, Dubois *et al.* [36] showed that the least informative belief function with respect to the q-ordering is unique and consonant. It can be constructed as follows:

- The probability measure is first transformed into a possibility measure:

$$poss\,(\omega_i) \quad = \quad \sum_{\omega_j \in \Omega} \min(P(\omega_i), P(\omega_j)),\ \forall \omega_i \in \Omega. \tag{2.9}$$

- The possibilities $\pi_j = poss(\omega_{i_j})$ are sorted so that:

$$\pi_1 \geq \pi_2 \geq \ldots \geq \pi_{|\Omega|}. \tag{2.10}$$

- The associated consonant mass function is then defined as:

$$m(A) \quad = \quad \begin{cases} \pi_j - \pi_{j+1} & \text{if } A = \{\omega_{i_1}, \ldots, \omega_{i_j}\}, \\ \pi_{|\Omega|} & \text{if } A = \Omega, \\ 0 & \text{otherwise.} \end{cases} \tag{2.11}$$

The ignorance $m(\Omega)$ resulting from this transformation is equal to the minimum plausibility of the singletons. In particular, a uniform probability distribution leads to the vacuous mass function.

2.1.4 Discounting

In the theory of belief functions, knowledge about the reliability of a source of information can be handled by a discounting factor [105, Chapter 11, Section5]. A discounting factor is used to *weaken* a mass function by transferring some masses to the ignorance state.

Definition 2.13. *For a factor $\alpha \in [0, 1]$, the discounted mass function ${}^{\alpha}m$ is defined as*

$${}^{\alpha}m(A) = (1 - \alpha)m(A), \quad \forall A \subsetneq \Omega \tag{2.12a}$$

$${}^{\alpha}m(\Omega) = (1 - \alpha)m(\Omega) + \alpha. \tag{2.12b}$$

If $\alpha = 0$, the information is considered reliable and is kept as is. On the other hand, if $\alpha = 1$, the information is totally unreliable and leads to the vacuous mass function. Smets [108] showed that the discounting equation (2.12) can be derived by interpreting $1 - \alpha$ as the degree of belief that the information is reliable. Thus, the discounting factor α plays a role equivalent to $P_R(r = 0)$ in the probabilistic case (1.6).

2.2 Combination rules

2.2.1 Dempster's rule

Definition 2.14. *Given two mass functions m_1 and m_2 induced by two independent sources of information, one can combine them using* Dempster's

rule of combination, *or* orthogonal sum, *to compute a new mass function* $m_1 \oplus m_2$ *defined as*

$$(m_1 \oplus m_2)(\emptyset) = 0, \tag{2.13a}$$

$$(m_1 \oplus m_2)(A) = \frac{1}{1-\kappa} \sum_{B \cap C = A} m_1(B) m_2(C), \quad \forall A \neq \emptyset, \tag{2.13b}$$

where

$$\kappa = \sum_{B \cap C = \emptyset} m(B) m(C). \tag{2.14}$$

The quantity κ measures the conflict between the two mass functions. The combination rule (2.13) is valid if and only if $\kappa < 1$, otherwise, m_1 and m_2 are incompatible and cannot be combined. Dempster's rule is commutative, associative and has the vacuous mass function as unique neutral element.

Dempster's rule can be expressed very simply in terms of commonality functions. Given L commonality functions $q_1, \ldots, q_L$, their combination by Dempster's rule is defined as

$$(q_1 \oplus \ldots \oplus q_L)(A) = \frac{1}{1-K} \prod_{1 \leq i \leq L} q_i(A), \tag{2.15a}$$

$$\propto \prod_{1 \leq i \leq L} q_i(A), \quad \forall A \subseteq \Omega, \tag{2.15b}$$

where K is the overall conflict resulting from the combination of the mass functions $m_1, \ldots, m_L$. It can be expressed as

$$K = \sum_{\substack{A_1, \ldots, A_L \\ \bigcap A_i = \emptyset}} \prod_{1 \leq i \leq L} m_i(A_i). \tag{2.16}$$

We can see from (2.15) that Dempster's rule is actually equivalent to the probabilistic product rule (1.3) when combining Bayesian mass functions. Indeed, we have $\forall \omega_i \in \Omega,\ q(\{\omega_i\}) = m(\{\omega_i\})$.

2.2.2 Cautious rule

Similarly to the probabilistic product rule, Dempster's rule is not idempotent. It assumes that the mass functions to combine are induced by independent sources of information. In cases where this independence assumption is not reasonable, Denœux [26] proposed to use the cautious rule.

Every mass function m can be written as the combination by Dempster's rule of simple mass functions

$$m = \bigoplus_{\emptyset \neq A \subset \Omega} A^{w(A)}, \tag{2.17}$$

with $w(A) \in [0,1]$ for all $A \in 2^\Omega \backslash \{\emptyset\}$. This representation is called the *canonical decomposition* of m and is unique for non dogmatic mass functions. The weights $w(A)$ can be obtained from the commonalities as follows:

$$w(A) \quad = \quad \prod_{B \supseteq A} q(B)^{(-1)^{|B|-|A|+1}}, \tag{2.18}$$

for all $A \subset \Omega$.

Definition 2.15. *Let m_1 and m_2 be two non dogmatic mass functions and w_1 and w_2 their associated weights from their respective canonical decomposition. Their combination using the cautious rule is noted $m_1 \circledcirc m_2$ and is defined as*

$$m_1 \circledcirc m_2 \quad = \quad \bigoplus_{A \subset \Omega} A^{w_1(A) \wedge w_2(A)}, \tag{2.19}$$

where $\wedge$ denotes the minimum operator. The cautious rule is associative, commutative, idempotent and has the vacuous mass function as unique neutral element.

2.2.3 Triangular norm-based rules

It is possible to formulate both Dempster's rule and the cautious rule with a triangular norm-based combination rule [26, 98]. The combination with Dempster's rule of two non dogmatic mass functions m_1 and m_2 can be written as an expression similar to (2.19):

$$m_1 \oplus m_2 \quad = \quad \bigoplus_{A \subset \Omega} A^{w_1(A) w_2(A)}. \tag{2.20}$$

With Dempster's rule, the weights w_1 and w_2 are multiplied while with the cautious rule the minimum operator is used. Frank's family of t-norms generalizing these two operators is defined as

$$w_1 \top_s w_2 \quad = \quad \begin{cases} w_1 \wedge w_2 & \text{if } s = 0, \\ w_1 w_2 & \text{if } s = 1, \\ \log_s \left(1 + \frac{(s^{w_1}-1)(s^{w_2}-1)}{s-1}\right) & \text{otherwise.} \end{cases} \tag{2.21}$$

For any $s \in [0,1]$, $w_1 \top_s w_2$ returns a value between $w_1 w_2$ and $w_1 \wedge w_2$.

Definition 2.16. *The combination of two non dogmatic mass functions m_1 and m_2 with Frank's triangular norm of parameter $s \in [0,1]$ is defined as*

$$m_1 \oplus_s m_2 \quad = \quad \bigoplus_{A \subset \Omega} A^{w_1(A) \top_s w_2(A)}. \tag{2.22}$$

The degree of independence between the two mass functions m_1 and m_2 can be taken into account by varying the value of s from 0 to 1.

2.2.4 Other combination rules

Dempster's rule has been often criticized and its validity questioned as it may lead to some *counter-intuitive* results when combining conflicting evidence. Many alternatives to Dempster's rule can be found in the literature [71, 132, 114]. These rules were mainly proposed to address the issue of combining conflicting information. More specifically, the conflict κ (2.14) resulting from the combination of two mass functions is distributed differently to the various subsets of the frame of discernment.

Similarly to the use of the average or maximum rule in the probabilistic case, these alternative rules allow the combination of contradictory information but prevent the representation of certain types of information. In particular, categorical, Bayesian and dogmatic mass functions can often not be properly handled by these alternative combination rules. The use of these alternative rules thus limits the power of the belief functions theory to represent a large variety of information. In this work, we adopt the same point of view as Haenni [55] who considered that so-called counter-intuitive results that may be obtained from Dempster's rule are often due to a wrong modelisation of the pieces of evidence to combine. Efforts should thus be put on representing properly the information at hand rather than on modifying the combination rule.

2.3 Operations over the frame of discernment

2.3.1 Refinement

Because mass functions are directly defined over sets of classes, refinement and imprecise information can be easily handled.

Definition 2.17. *Given a refining $\rho : 2^{\Omega} \rightarrow 2^{\Theta}$, a mass function m^{Ω} defined over Ω can be transformed into a mass function m^{Θ} defined over Θ, such that for all $B \subseteq \Theta$:*

$$m^{\Theta}(B) \quad = \quad \begin{cases} m^{\Omega}(A) & \text{if } \exists A \subseteq \Omega,\ B = \rho(A), \\ 0 & \text{otherwise.} \end{cases} \tag{2.23}$$

Example 2. Remind Example 1, where an initial probability $P^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}$ defined over the frame $\Omega_{\text{G}} = \{\text{ground}, \sout{\text{ground}}\}$ has to be redefined over the frame $\Lambda = \{\text{ground}, \text{vertical}, \text{sky}\}$. The probability $P^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}$ can be interpreted as a Bayesian mass function $m^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}$ defined as

$$m^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}(\{\text{ground}\}) = q, \quad m^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}(\{\sout{\text{ground}}\}) = 1 - q, \quad m^{\Omega_{\text{G}}}_{\mathbf{x}_{\text{G}}}(\Omega_{\text{G}}) = 0, \tag{2.24}$$

where $q \in [0, 1]$. The refining from Ω_{G} to Λ (1.10) then simply yields

$$m^{\Lambda}_{\mathbf{x}_{\text{G}}}(\{\text{ground}\}) = q, \quad m^{\Lambda}_{\mathbf{x}_{\text{G}}}(\{\text{vertical}, \text{sky}\}) = 1 - q, \quad m^{\Lambda}_{\mathbf{x}_{\text{G}}}(\Lambda) = 0. \tag{2.25}$$

The initial mass assigned to the singleton {~~ground~~} is simply transferred to {vertical, sky} and not uniformly distributed as in the probabilistic case. The information encoded by the two mass functions $m^{\Omega_G}_{\mathbf{x}_G}$ and $m^{\Lambda}_{\mathbf{x}_G}$ is thus the same.

2.3.2 Coarsening

The opposite operation to refining is called coarsening. If a frame of discernment Θ is a refinement of Ω, then Ω is a coarsening of Θ. By definition, the cardinality of a refinement Θ is greater than the cardinality of the original frame Ω. This implies that a refining $\rho : 2^{\Omega} \to 2^{\Theta}$ cannot be bijective, thus not invertible. There are two ways to define a mass function over a coarser frame.

Definition 2.18. *The* inner reduction $\underline{\varphi} : 2^{\Theta} \to 2^{\Omega}$ *and* outer reduction $\overline{\varphi} : 2^{\Theta} \to 2^{\Omega}$ *associated to a refining ρ are defined, respectively, as*

$$\underline{\varphi}(B) = \{\omega \in \Omega | \rho(\{\omega\}) \subseteq B\}, \quad \forall B \subseteq \Theta, \tag{2.26}$$

$$\overline{\varphi}(B) = \{\omega \in \Omega | \rho(\{\omega\}) \cap B \neq \emptyset\}, \quad \forall B \subseteq \Theta. \tag{2.27}$$

Definition 2.19. *The inner and outer reduction of a mass function m^{Ω} over Θ are defined, respectively, as*

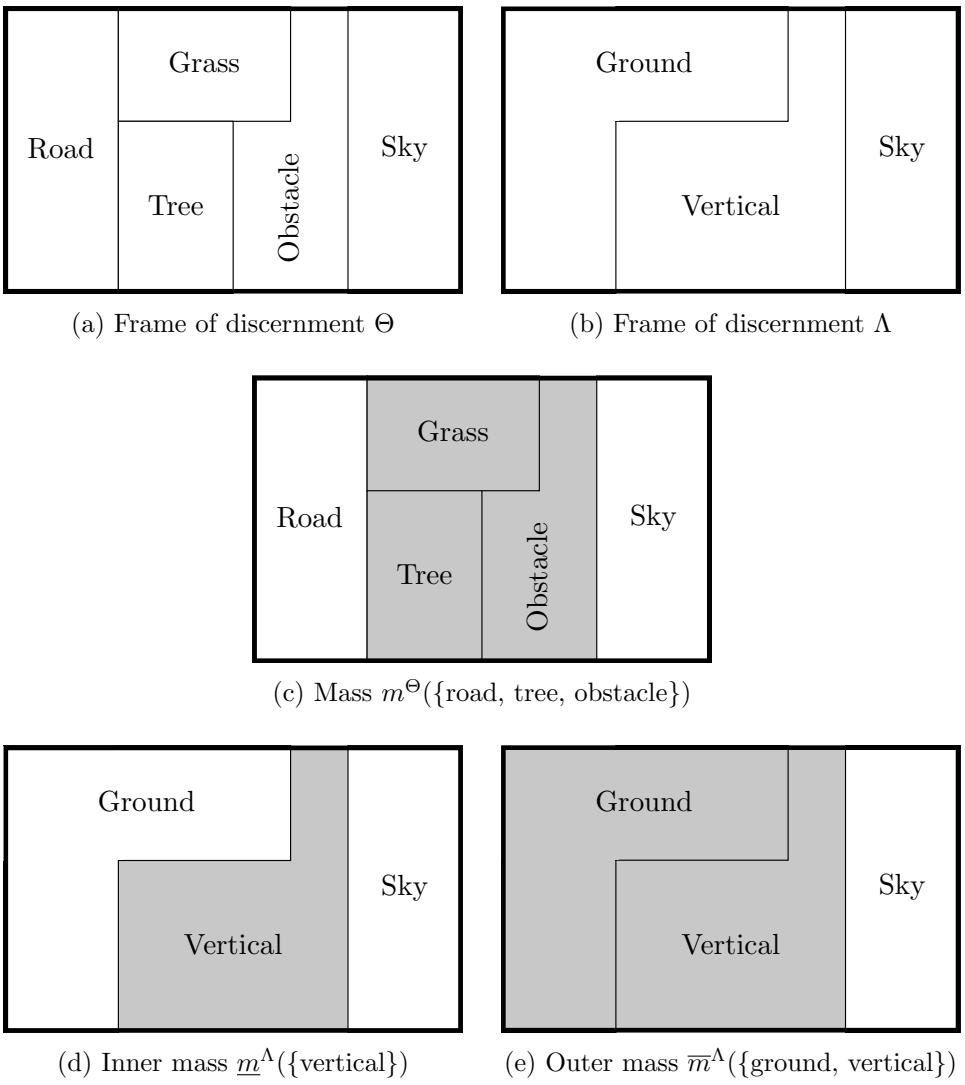
$$\underline{m}^{\Theta}(A) = \sum_{B \subseteq \Omega,\ \underline{\varphi}(B)=A} m^{\Omega}(B), \quad \forall A \subseteq \Theta, \tag{2.28}$$

$$\overline{m}^{\Theta}(A) = \sum_{B \subseteq \Omega,\ \overline{\varphi}(B)=A} m^{\Omega}(B), \quad \forall A \subseteq \Theta. \tag{2.29}$$

Remark. In general, the inner reduction $\underline{m}^{\Theta}$ defined by Equation 2.28 is not normalized, i.e., we may have $\underline{m}^{\Theta}(\emptyset) > 0$. It is, however, not the case for the outer reduction $\overline{m}^{\Theta}$ which is always a normalized mass function.

Example 3. Suppose several mass functions were combined over a common refined frame Θ = {road, grass, tree, obstacle, sky}. Suppose now that we actually only need to reason over the coarser frame Λ = {ground, vertical, sky}. These two frames are illustrated in Figure 2.1a and Figure 2.1b. The set B = {road, tree, obstacle} $\subset \Omega$ (see Figure 2.1c) has no correspondence in the frame Λ. The inner reduction $\underline{\varphi}(B)$ = {vertical} corresponds to the largest subset of Λ that is entirely included in B (see Figure 2.1d). The outer reduction $\overline{\varphi}(B)$ = {ground, vertical} corresponds to the smallest subset of Λ that intersects B (see Figure 2.1e). It is important to note that if the mass functions $\underline{m}^{\Lambda}$ and $\overline{m}^{\Lambda}$ are refined back to Θ, none of them would actually lead to the initial mass function m^{Θ}.

Proposition 2.2. *Let m^{Θ} be a mass function defined over Θ and $\overline{m}^{\Omega}$ its outer reduction over Ω. The following propositions hold:*



(a) Frame of discernment Θ

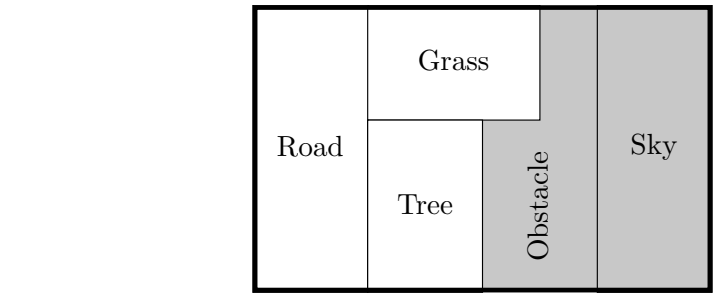
(b) Frame of discernment Λ

(c) Mass $m^{\Theta}(\{\text{road, tree, obstacle}\})$

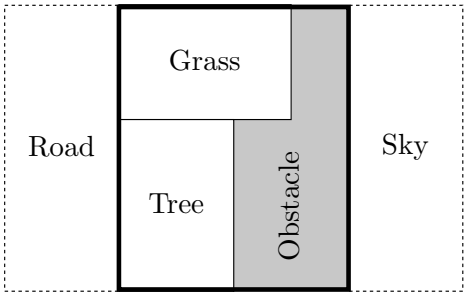
(d) Inner mass $\underline{m}^{\Lambda}(\{\text{vertical}\})$

(e) Outer mass $\overline{m}^{\Lambda}(\{\text{ground, vertical}\})$

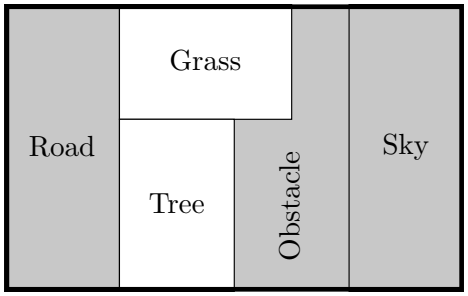
Figure 2.1: Coarsening of a frame of discernment.



(a) $m^{\Theta}(\{\text{obstacle, sky}\})$



(b) $m\,[B]^{\Theta}\,(\{\text{obstacle}\})$



(c) $m^{B\Uparrow\Theta}(\{\text{road, obstacle, sky}\})$

Figure 2.2: Conditioning and deconditioning.

1. $\overline{Bel}^{\Omega}(A) = Bel^{\Theta}(\rho(A)), \quad \forall A \subseteq \Omega$;
2. $\overline{Pl}^{\Omega}(A) = Pl^{\Theta}(\rho(A)), \quad \forall A \subseteq \Omega$.

In practice, the outer reduction is often preferred as it is consistent with respect to the belief and plausibility functions.

2.3.3 Conditioning

The conditioning of a mass function can be formulated as its combination with a categorical mass function.

Definition 2.20. *Let m^{Ω} be a mass function. The conditioning $m^{\Omega}[B]$ of m with respect to $B \subseteq \Omega$ is defined as*

$$m^{\Omega}[B] \quad = \quad m^{\Omega} \oplus B^{0}, \tag{2.30}$$

where B^0 is the categorical mass function with unique focal element B.

The mass function $m^{\Omega}[B]$ encodes the information represented by m^{Ω} given that hypothesis B is true. In particular, we have $m^{\Omega}[B](A) = 0$ for all $A \not\subseteq B$. Suppose now that we are given a conditional mass function $m^{\Omega}[B]$. In general, there are multiple mass functions whose conditioning

with respect to B would lead to $m^{\Omega}[B]$. The least commitment principle tells that the least committed one should be used.

Definition 2.21. *Given a conditional mass function $m^{\Omega}[B]$, its ballooning extension $m^{B\Uparrow\Omega}$ is defined as*

$$\begin{cases} m^{B\Uparrow\Omega}\left(A \cup \overline{B}\right) = m^{\Omega}[B](A) & , \forall A \subseteq B, \\ m^{B\Uparrow\Omega}(C) = 0 & , \forall C \subset \overline{B}. \end{cases} \tag{2.31}$$

Example 4. Figure 2.2 shows an example of conditioning and ballooning extension. The initial mass $m^{\Theta}(\{\text{obstacle, sky}\})$ is transferred to the singleton $\{\text{obstacle}\}$ after conditioning with respect to the set $B = \{\text{grass, tree, obstacle}\}$. The ballooning extension of this conditional mass function would then transfer this mass to the set $\{\text{road, obstacle, sky}\}$.

2.4 Decision making

The final aim of a classification task is to decide a class from the frame of discernment. There exist several strategies for decision making [24] when reasoning within the theory of belief functions. Two simple strategies consist in choosing the singleton with maximum belief or plausibility. They are called, respectively, the pessimistic and optimistic strategy. Another widely used strategy is to use the pignistic probability transformation and selected the singleton with maximum probability. In this thesis, the optimistic strategy will be used. Two principal arguments can be stated in favor of this choice. The first is that selecting the singleton with maximum plausibility is computationally efficient [7]. The second argument is that the decisions made from the optimistic strategy remain coherent with frame refinement.

2.4.1 Computation considerations

In general, the computation of Dempster's rule (2.13) requires a number of operations exponential in $|\Omega|$. This can easily lead to intractable computation when $|\Omega|$ becomes large, typically as soon as $|\Omega| \geq 10$. Barnett [7] showed that for certain subsets of Ω, the plausibility can be computed efficiently.

Definition 2.22. *Let $\mathcal{E} = \{m_i | 1 \leq i \leq L\}$ be a set of L mass functions to combine and $\mathcal{F}_i$ the set of focal elements associated to each m_i. A set $A \subset \Omega$ is said to be* atomic *with respect to $\mathcal{E}$ if and only if*

$$1) \quad A \neq \emptyset, \tag{2.32a}$$

$$2) \quad \forall i \in \{1, \ldots, L\},\ \forall F \in \mathcal{F}_i,\ A \subseteq F \text{ or } A \cap F = \emptyset. \tag{2.32b}$$

Intuitively, it means that a non-empty set $A \subset \Omega$ is atomic if none of the mass functions from $\mathcal{E}$ can distinguish an element of A from another one. It can easily be shown that a singleton $\{\omega\} \subset \Omega$ is always atomic. Barnett [7] showed that if A is atomic with respect to $\mathcal{E}$ then

$$\forall i \in \{1, \ldots, L\}, \qquad Pl_i(A) = q_i(A), \tag{2.33}$$

which leads to

$$\begin{aligned}(Pl_1 \oplus \ldots \oplus Pl_L)(A) &= (q_1 \oplus \ldots \oplus q_L)(A) && \text{(2.34a)}\\ &\propto \prod_{1 \leq i \leq L} q_i(A) && \text{(2.34b)}\\ &\propto \prod_{1 \leq i \leq L} Pl_i(A), && \text{(2.34c)}\end{aligned}$$

where q_i and Pl_i are the commonality and plausibility associated to m_i. As singletons are always atomic, we finally get

$$\begin{aligned}(Pl_1 \oplus \ldots \oplus Pl_L)(\{\omega\}) &\propto \prod_{1 \leq i \leq L} Pl_i(\{\omega\}) && \text{(2.35a)}\\ &\propto \prod_{1 \leq i \leq L} pl_i(\omega). && \text{(2.35b)}\end{aligned}$$

To find the singleton with maximum plausibility resulting from the combinations of m_i, we only need to multiply the associated contour functions pl_i. In theory, the computation of pl_i from m_i requires $O(|\mathcal{F}_i|)$ time which can be as large as $O(2^{|\Omega|})$. However, in many practical situations, we have $\mathcal{F}_i \ll 2^{|\Omega|}$. Moreover, in practice, the information at hand can be directly encoded as a contour function without the need of explicitly representing the underlying mass function. In such situations, the optimistic decision reached from L sources of information can be computed in $O(L|\Omega|)$ time.

2.4.2 Decision making example

To show the differences between different decision making strategies, let us consider the following mass function defined on $\Omega = \{\text{grass, road, } \sout{\text{ground}}\}$:

$$\begin{aligned}m^\Omega(\{\text{grass, road}\}) &= 0.2, && \text{(2.36a)}\\ m^\Omega(\{\text{grass, } \sout{\text{ground}}\}) &= 0.3, && \text{(2.36b)}\\ m^\Omega(\{\text{road, } \sout{\text{ground}}\}) &= 0.5. && \text{(2.36c)}\end{aligned}$$

Table 2.2 shows the beliefs, plausibilities and pignistic probabilities on the singletons. Here, the pessimistic strategy cannot lead to any decision: actually, in the worst case scenario, any decision could be wrong given the current

Table 2.2: Bel^{Ω}, Pl^{Ω} and $BetP^{\Omega}$ from mass function (2.36).

	{grass}	{road}	{~~ground~~}
Bel^{Ω}	0	0	0
Pl^{Ω}	0.5	0.7	**0.8**
$BetP^{\Omega}$	0.25	0.35	**0.4**

Table 2.3: Bel^{Θ}, Pl^{Θ} and $BetP^{\Theta}$ from mass function (2.37).

			{~~ground~~}		
	{grass}	{road}	{tree}	{obst.}	{sky}
Bel^{Θ}	0	0	0	0	0
Pl^{Θ}	0.5	0.7	**0.8**	**0.8**	**0.8**
$BetP^{\Theta}$	0.175	**0.225**	0.2	0.2	0.2

mass function. Choosing {grass} instead of {~~ground~~} would be wrong if the masses m^{Ω}({grass, ~~ground~~}) and m^{Ω}({road, ~~ground~~}) were actually entirely related to {~~ground~~}. Conversely, the other decision would also be wrong if the same masses were now related respectively to {grass} and {road}. On the other hand, both pl^{Ω} and $BetP^{\Omega}$ would lead to {~~ground~~}, which seems quite reasonable.

Now, if the singleton {~~ground~~} is refined into {tree, obstacle, sky}, the mass function (2.36) will simply be rewritten as

$$m^{\Theta}(\{\text{grass, road}\}) = 0.2, \tag{2.37a}$$
$$m^{\Theta}(\{\text{grass, tree, obstacle, sky}\}) = 0.3, \tag{2.37b}$$
$$m^{\Theta}(\{\text{road, tree, obstacle, sky}\}) = 0.5. \tag{2.37c}$$

Table 2.3 shows the measures induced by this new mass function. Following $BetP^{\Theta}$, the decision is changed and now leads to {road}. In contrast, the plausibility criterion does not discriminate between {tree}, {obstacle} and {sky}, which are still more plausible than {grass} and {road}. The optimistic strategy thus remains coherent with its previous decision. The optimistic strategy is thus more conclusive than the pessimistic one and more coherent than the pignistic one.

2.5 Statistical inference

The theory of belief functions can also be used for statistical inference. Shafer [105] originally proposed to use a "likelihood-based" belief function for statistical inference. This approach was further justified by Denœux [27]. Knowledge about some parameters can then be used for prediction as in [59].

2.5.1 Likelihood-based belief function

Let $X \in \mathbb{X}$ be some observable data and $\theta \in \Theta$ the unknown parameter of the density function $f_\theta(x)$ generating the data. Information about θ can be inferred given the outcome x of a random experiment. Shafer [105] proposed to build a belief function Bel_x^Θ on Θ from the likelihood function. After observing $X = x$, the likelihood function $L_x : \theta \mapsto f_\theta(x)$ is normalized to yield the following contour function:

$$pl_x^\Theta(\theta) \quad = \quad \frac{L_x(\theta)}{\sup_{\theta' \in \Theta} L_x(\theta')}, \quad \forall \theta \in \Theta, \tag{2.38}$$

where sup denotes the supremum operator. The consonant plausibility function associated to this contour function is

$$Pl_x^\Theta(A) \quad = \quad \sup_{\theta \in A} pl_x^\Theta(\theta), \quad \forall A \subseteq \Omega. \tag{2.39}$$

The focal sets of Bel_x^Θ are defined as

$$\Gamma_x(\gamma) \quad = \quad \{\theta \in \Theta \mid pl_x^\Theta(\theta) \geq \gamma\}, \quad \forall \gamma \in [0,1]. \tag{2.40}$$

The random sets formalism [85] can be used to represent the belief and plausibility functions on Θ. Given the Lebesgue measure λ on $[0,1]$ and the multi-valued mapping $\Gamma_x : [0,1] \to 2^\Theta$, we have

$$\begin{array}{rcl} Bel_x^\Theta(A) & = & \lambda\left(\{\gamma \in [0,1] \mid \Gamma_x(\gamma) \subseteq A\}\right) \\ Pl_x^\Theta(A) & = & \lambda\left(\{\gamma \in [0,1] \mid \Gamma_x(\gamma) \cap A \neq \emptyset\}\right) \end{array}, \qquad \forall A \subseteq \Theta. \tag{2.41}$$

A complete description of the theory of random sets and its relation to Dempster-Shafer theory can be found in [85].

2.5.2 Forecasting

Suppose that we now have some knowledge about θ after observing some training data x. The *forecasting* problem consists in making some predictions about some random quantity $Y \in \mathbb{Y}$ whose conditional distribution $g_{x,\theta}(y)$ given $X = x$ depends on θ. A belief function on $\mathbb{Y}$ can be derived from the sampling model proposed by Dempster [21]. For some unobserved auxiliary variable $Z \in \mathbb{Z}$ with known probability distribution μ independent of θ, we define a function φ so that

$$Y \quad = \quad \varphi(\theta, Z). \tag{2.42}$$

A multi-valued mapping $\Gamma'_x : [0,1] \times \mathbb{Z} \to 2^\mathbb{Y}$ is defined by composing Γ_x with φ

$$\begin{array}{rccc} \Gamma'_x : & [0,1] \times \mathbb{Z} & \to & 2^\mathbb{Y} \\ & (\gamma, z) & \mapsto & \varphi(\Gamma_x(\gamma), z). \end{array} \tag{2.43}$$

A belief function on $\mathbb{Y}$ can then be derived from the product measure $\lambda \otimes \mu$ on $[0,1] \times \mathbb{Z}$ and the multi-valued mapping Γ'_x

$$Bel_x^{\mathbb{Y}}(A) = (\lambda \otimes \mu)\left(\{(\gamma, z) \mid \varphi\left(\Gamma_x(\gamma), z\right) \subseteq A\}\right), \tag{2.44a}$$

$$Pl_x^{\mathbb{Y}}(A) = (\lambda \otimes \mu)\left(\{(\gamma, z) \mid \varphi\left(\Gamma_x(\gamma), z\right) \cap A \neq \emptyset\}\right), \tag{2.44b}$$

for all $A \subseteq \mathbb{Y}$.

2.5.3 Binary case example

In the particular case where Y is a random variable with a Bernoulli distribution $\mathcal{B}(\omega)$, it can be generated by a function φ defined as

$$Y = \varphi(\omega, Z) = \begin{cases} 1 & \text{if } Z \leq \omega, \\ 0 & \text{otherwise,} \end{cases} \tag{2.45}$$

where Z has a uniform distribution on $[0,1]$. Assume that the belief function Bel_x^{Ω} on Ω is induced by a random closed interval $\Gamma_x(\gamma) = [U(\gamma), V(\gamma)]$. In particular, it is the case if it is the consonant belief function associated to a unimodal contour function. We get

$$\Gamma'_x(\gamma, z) = \varphi\left([U(\gamma), V(\gamma)], z\right) \quad = \quad \begin{cases} 1 & \text{if } Z \leq U(\gamma), \\ 0 & \text{if } Z > V(\gamma), \\ \{0,1\} & \text{otherwise.} \end{cases} \tag{2.46}$$

The *predictive* belief function $Bel_x^{\mathbb{Y}}$ can then be computed as

$$Bel_x^{\mathbb{Y}}(\{1\}) = (\lambda \otimes \mu)(\{(\gamma, z) \mid Z \leq U(\gamma)\}) \tag{2.47a}$$

$$= \int_0^1 \mu(\{z \mid z \leq U(\gamma)\}) f(\gamma) d\gamma \tag{2.47b}$$

$$= \int_0^1 U(\gamma) f(\gamma) d\gamma = \mathbb{E}(U) \tag{2.47c}$$

and

$$Bel_x^{\mathbb{Y}}(\{0\}) = (\lambda \otimes \mu)(\{(\gamma, z) \mid Z > V(\gamma)\}) \tag{2.48a}$$

$$= 1 - (\lambda \otimes \mu)(\{(\gamma, z) \mid Z \leq V(\gamma)\}) \tag{2.48b}$$

$$= 1 - \mathbb{E}(V). \tag{2.48c}$$

As U and V take only non-negative values, these quantities have the following expressions:

$$Bel_x^{\mathbb{Y}}(\{1\}) = \int_0^{+\infty} (1 - F_U(u)) du \tag{2.49a}$$

$$= \int_0^{\hat{\omega}} (1 - pl_x^{\Omega}(u)) du \tag{2.49b}$$

$$= \hat{\omega} - \int_0^{\hat{\omega}} pl_x^{\Omega}(u) du \tag{2.49c}$$

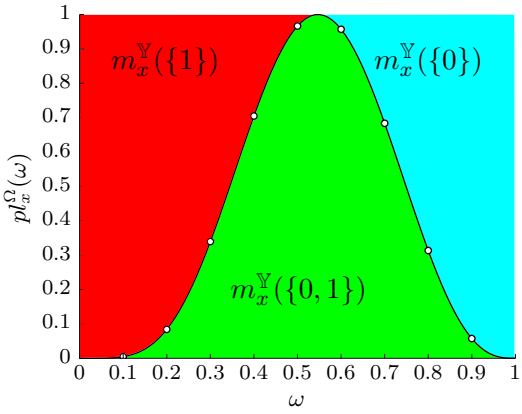


Figure 2.3: Predictive mass function $m_x^{\mathbb{Y}}$ based on the contour function pl_x^{Ω}.

and

$$Pl_x^{\mathbb{Y}}(\{1\}) = 1 - Bel_x^{\mathbb{Y}}(\{0\}) \tag{2.50a}$$

$$= \int_0^{+\infty} (1 - F_V(v))dv \tag{2.50b}$$

$$= \hat{\omega} + \int_{\hat{\omega}}^{1} pl_x^{\Omega}(v)dv, \tag{2.50c}$$

where $\hat{\omega}$ is the value maximizing pl_x^{Ω}. In many practical situations, the belief function $Bel_x^{\mathbb{Y}}$ cannot be expressed analytically. However, they can be approximated either by Monte Carlo simulation using Equations (2.47) and (2.48) or by numerically estimating the integrals of Equations (2.49) and (2.50). The predictive mass function $m_x^{\mathbb{Y}}$ can be represented by the areas of regions delimited by the contour function, as shown in Figure 2.3.

2.6 Conclusion

The theory of belief functions is a generalization of the theory of probability. It can be used to represent many types of imperfect information. In particular, imprecision and ignorance are better modeled by belief functions compared to classical probabilities. Several combination rules are also defined to combine information coming from different sources. The definition of new classes through refinement is easily handled as the reasoning is done over sets of classes. This theory will be used in the next three chapters to model different kinds of imperfection information.

Chapter 3

Calibration of classifiers

The combination of pattern classifiers is an important issue in machine learning. In many practical situations, different kinds of classifiers have to be combined. If the outputs of the classifiers are of the same nature, such as probability measures or belief functions, they can be combined directly. Several classification methods such a k-nearest neighbor rule, neural network or decision trees can return probabilities. Evidential versions of these methods can be found in the literature [23, 25, 112]. Other methods like support vector machines (SVM) or boosting may only return some classification scores. In order to combine different types of classifiers, their outputs need to be made comparable.

The transformation of the output of a classifier into a posterior class probability is called calibration. Several methods can be found in the literature [95, 138, 139]. In this chapter, we first review in Section 3.1 three binary probabilistic calibration methods and study their evidential extension. Then, in Section 3.2, we extend the calibration to multi-class problems. Finally, some experimental results are presented in Section 3.3.

3.1 Binary classifier calibration

Let us consider a binary classification problem. Let $\mathcal{X} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ be some training data, where $x_i \in \mathbb{R}$ is the score returned by a pre-trained classifier for the i-th training sample whose label is $y_i \in \{0, 1\}$. Given a test sample of score $s \in \mathbb{R}$ and unknown label $y \in \{0, 1\}$, the aim of calibration is to estimate the posterior class probability $P(y = 1|s)$. Several calibration methods can be found in the literature. Binning [138], isotonic regression [139] and logistic regression [95] are the most commonly used.

3.1.1 Binning

Binning [138] is a rather simple way to do calibration by partitioning the score space into bins. For the j-th bin, which is an interval $[\underline{s}_j, \overline{s}_j]$, we count the number of positive examples k_j over all the n_j training examples whose score falls into this particular bin. Given a test sample of score $s \in [\underline{s}_j, \overline{s}_j]$ and unknown label $y \in \{0, 1\}$, estimating $P\left(y = 1 | s \in [\underline{s}_j, \overline{s}_j]\right)$ can be formulated as a simple binomial proportion estimation problem. Given n_j trials with k_j successes, the probability $P\left(y = 1 | s \in [\underline{s}_j, \overline{s}_j]\right)$ of having a success from a new trial is the unknown binomial proportion $\tau_j \in [0, 1]$ associated to the j-th bin. The best estimate of τ_j from a Bayesian point of view is simply $\hat{\tau}_j = k_j / n_j$, which yields

$$P_{\text{B}}\left(y = 1 | s \in [\underline{s}_j, \overline{s}_j]\right) = \frac{k_j}{n_j}, \quad P_{\text{B}}\left(y = 0 | s \in [\underline{s}_j, \overline{s}_j]\right) = \frac{n_j - k_j}{n_j}. \tag{3.1}$$

To avoid a crisp 0 or 1 probability, especially in case of few data, the Laplace estimator can be used instead. It assumes that both a success and a failure have already been observed prior to the n_j trials. This leads to the following estimator:

$$P_{\text{L}}\left(y = 1 | s \in [\underline{s}_j, \overline{s}_j]\right) = \frac{k_j + 1}{n_j + 2}, \quad P_{\text{L}}\left(y = 0 | s \in [\underline{s}_j, \overline{s}_j]\right) = \frac{n_j - k_j + 1}{n_j + 2}. \tag{3.2}$$

One limitation of these two probabilistic estimators is that the uncertainty due to the number of samples is not taken into account. For example, a bin containing 10 positive examples out of 20 and another one with 100 positive examples out of 200 will be given the same Bayesian and Laplacian estimates. Yet, in the second case, the estimate is much more certain. This kind of uncertainty can be better handled by using belief functions instead of probabilities.

One way to get belief functions is to transform the probabilistic outputs into mass functions using the inverse pignistic transformation (see Section 2.1.3). However, it does not solve the issue mentioned previously. Another simple way is to use Dempster's model [22] which leads to the following mass function:

$$m_{\text{D}}(\{1\}) = \frac{k_j}{n_j + 1}, \quad m_{\text{D}}(\{0\}) = \frac{n_j - k_j}{n_j + 1}, \quad m_{\text{D}}(\{0, 1\}) = \frac{1}{n_j + 1}. \tag{3.3}$$

Similarly to Laplace estimator, it can be interpreted as having observed one sample prior to the trial but with unknown label. The amount of ignorance $m_{\text{D}}(\{0, 1\})$ is inversely proportional to $n_j + 1$.

From a statistical inference point of view, confidence intervals are often used to better model the uncertainty due to a small sample size. A confidence interval $[\underline{\tau}_j, \overline{\tau}_j]$ at confidence level $1-\alpha \in [0, 1]$, i.e., $P\left(\underline{\tau}_j \leq \tau_j \leq \overline{\tau}_j\right) =$

$1-\alpha$, can be represented by the following contour function defined over $\tau_j \in T$:

$$pl^T_{\text{CI}}(\tau_j) = \begin{cases} 1 & \text{if } \underline{\tau}_j \leq \tau_j \leq \overline{\tau}_j, \\ \alpha & \text{otherwise.} \end{cases} \tag{3.4}$$

This contour function can be used within the Equations (2.49) and (2.50) to derive the associated predictive mass function. The associated predictive mass function is then defined as

$$m_{\text{CI}}(\{1\}) = (1-\alpha)\underline{\tau}_j, \quad m_{\text{CI}}(\{0\}) = (1-\alpha)(1-\overline{\tau}_j). \tag{3.5}$$

For the Clopper-Person interval [16], the bounds are defined as

$$\underline{\tau}_j = B\left(\frac{\alpha}{2}; k_j, n_j - k_j + 1\right), \quad \overline{\tau}_j = B\left(1-\frac{\alpha}{2}; k_j + 1, n_j - k_j\right), \tag{3.6}$$

where $B(q;\beta,\gamma)$ is the q-th quantile of a beta distribution with shape parameters β and γ. The choice of the confidence level is often arbitrary, a confidence level of 95% is a common one.

An alternative to confidence intervals is the use of the likelihood function as proposed by Denœux [27]. If the relative likelihood function is used as contour function for τ_j, we get

$$pl^T_{\text{L}}(\tau_j) = \frac{\tau_j^{k_j}(1-\tau_j)^{n_j-k_j}}{\hat{\tau}_j^{k_j}(1-\hat{\tau}_j)^{n_j-k_j}}, \tag{3.7}$$

which gives the following predictive mass function (see Section 2.5.3):

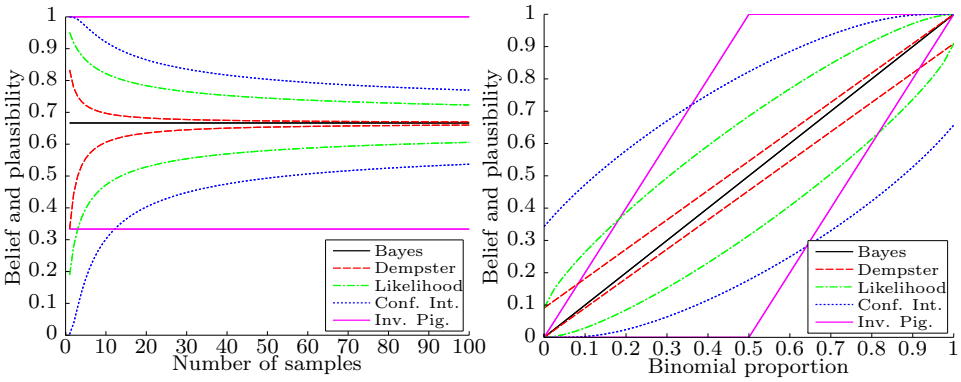
$$m_{\text{L}}(\{1\}) = \begin{cases} 0 & \text{if } \hat{\tau}_j = 0, \\ \hat{\tau}_j - \dfrac{\underline{B}(\hat{\tau}_j; k_j+1, n_j-k_j+1)}{\hat{\tau}_j^{k_j}(1-\hat{\tau}_j)^{n_j-k_j}} & \text{if } 0 < \hat{\tau}_j < 1, \\ \dfrac{n_j}{n_j+1} & \text{if } \hat{\tau}_j = 1, \end{cases} \tag{3.8a}$$

$$m_{\text{L}}(\{0\}) = \begin{cases} \dfrac{n_j}{n_j+1} & \text{if } \hat{\tau}_j = 0, \\ 1-\hat{\tau}_j - \dfrac{\overline{B}(\hat{\tau}_j; k_j+1, n_j-k_j+1)}{\hat{\tau}_j^{k_j}(1-\hat{\tau}_j)^{n_j-k_j}} & \text{if } 0 < \hat{\tau}_j < 1, \\ 0 & \text{if } \hat{\tau}_j = 1, \end{cases} \tag{3.8b}$$

where $\underline{B}$ and $\overline{B}$ are, respectively, the lower and upper incomplete beta function defined as

$$\underline{B}(z;a,b) = \int_0^z t^{a-1}(1-t)^{b-1}dt, \tag{3.9a}$$

$$\overline{B}(z;a,b) = \int_z^1 t^{a-1}(1-t)^{b-1}dt = \underline{B}(1-z;b,a). \tag{3.9b}$$



(a) (b)

Figure 3.1: (a) Belief and plausibility of success given a proportion of 2/3 w.r.t the number of samples. (b) Belief and plausibility of success given 10 samples w.r.t the binomial proportion. The Clopper-Pearson confidence interval was computed with a confidence level of 95%.

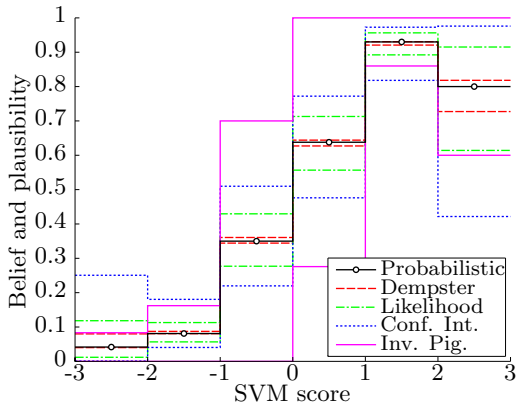
They can be computed exactly for integer values of a and b as

$$\underline{B}(z;a,b) = \sum_{j=a}^{a+b-1} \frac{(a-1)!(b-1)!}{j!(a+b-1-j)!} z^j (1-z)^{a+b-1-j}. \tag{3.10}$$

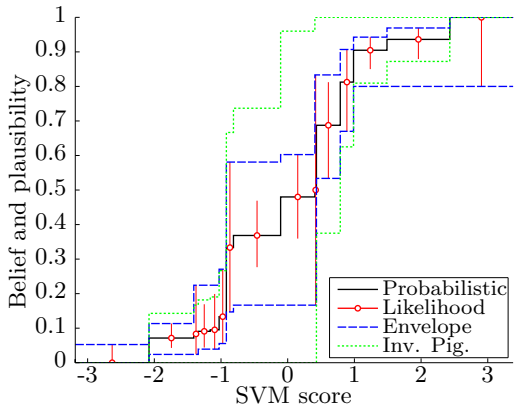
Figure 3.1 illustrates the belief and plausibility of success obtained with different number of samples and binomial proportions. In Figure 3.1a, we can see that when the number of sample grows, Dempster's model converges very rapidly to the Bayesian estimate while the Clopper-Pearson interval is more conservative. The likelihood-based approach gives intermediate results. As stated earlier, the probabilistic Bayesian estimator and its associated inverse pignistic transform does not take into account the number of samples. In Figure 3.1b, it is interesting to note that, when the empirical proportion $\hat{\tau}$ is equal to 0 or 1, the likelihood-based approach gives Dempster's model.

One difficulty of binning is the choice of the number and size of the bins. The number of bins can be set arbitrarily or optimized by cross-validation. In the later case, it may become problematic if only a few training data are available. Given a fixed number of bins, the boundaries of each bin are usually chosen so that all the bins have the same size or about the same number of samples. Figure 3.2 shows the results obtained after calibrating a SVM classifier on the UCI[1] Australian dataset. The binning was done using the bins $(-\infty,-3]$, $(-3,-2]$,..., $(+2,+3]$, $(+3,+\infty)$.

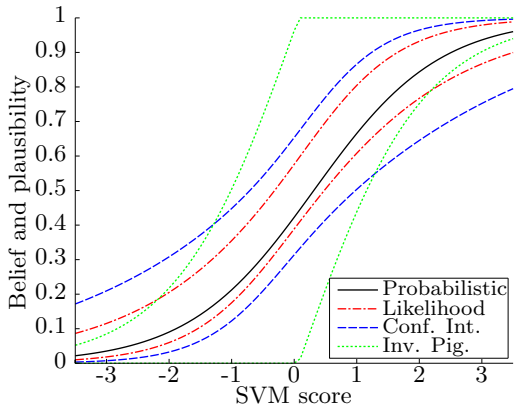
[1] http://archive.ics.uci.edu/ml



(a) Binning



(b) Isotonic regression



(c) Logistic regression

Figure 3.2: Belief and plausibility of having a positive example given an SVM score trained on the *Australian* dataset.

3.1.2 Isotonic regression

In many practical situations, the scores are seen as confidence measures. This implies that the transformation from a score to a probability measure should be done using a non-decreasing function. This assumption is strong, but it remains reasonable in many practical situations. An alternative to binning that incorporates such prior constraint is isotonic regression [139]. It consists in fitting a stepwise-constant non-decreasing, i.e., isotonic, function $g : \mathbb{R} \to [0, 1]$ to the training data by minimizing the mean-squared error

$$MSE(g, \mathcal{X}) \quad = \quad \frac{1}{n} \sum_{i=1}^{n} \left[g\left(x_i\right) - y_i \right]^2 , \tag{3.11}$$

The optimal function $\hat{g}$ can be computed efficiently using the pair-adjacent violators (PAV) algorithm [4], which is detailed in Algorithm 1.

Algorithm 1 Pair-adjacent violators (PAV) algorithm

Require: training data $(x_1, y_1), \ldots, (x_i, y_i), \ldots, (x_n, y_n)$ sorted w.r.t. x_i
  $\hat{g}_i \leftarrow 0$, $w_i \leftarrow 0$
  $\hat{g}_1 \leftarrow y_1$, $w_1 \leftarrow 1$
  $i \leftarrow 1$
  for $j = 2 : n$ **do**
    $i \leftarrow i + 1$
    $\hat{g}(x_i) \leftarrow y_j$
    $w_i \leftarrow w_j$
    while $i \geq 2$ **and** $\hat{g}(x_{i-1}) \geq \hat{g}(x_i)$ **do**
      $\hat{g}(x_{i-1}) \leftarrow (w_{i-1} \cdot \hat{g}(x_{i-1}) + w_i \cdot \hat{g}(x_i))/(w_{i-1} + w_i)$
      $w_{i-1} \leftarrow w_{i-1} + w_i$
      $i \leftarrow i - 1$
    end while
  end for
  return $\hat{g}(s) = \hat{g}_{i,j}$, for $x_i < s \leq x_j$

The calibration result from isotonic regression can also be seen as a particular case of binning. All the previous methods can thus be used. Figure 3.2b shows the likelihood-based interval for each bin defined by the isotonic regression. We can see, however, that the lower and upper envelopes defined by the intervals are not isotonic. A simple way to get an isotonic envelope is to first scan the bins in increasing order and keep the highest upper bound seen so far to define the upper envelope, then to scan the bins in decreasing order and keep the lowest lower bound to define the lower envelope. Figure 3.2b also illustrates the obtained belief and plausibility functions.

3.1.3 Logistic regression

Platt [95] further constrains the calibration problem by using a parametrized formulation using logistic regression. Niculescu-Mizil and Caruana [86] showed that logistic regression is well-adapted for calibrating maximum margin methods like SVM. Moreover, it is less prone to over-fitting as compared to binning and isotonic regression, especially when relatively few training data are available. Logistic regression calibration consists in fitting a sigmoid function

$$P\left(y=1|s\right) \approx h_s(\theta) = \frac{1}{1+\exp\left(\theta_0+\theta_1 s\right)}. \tag{3.12}$$

The parameter $\theta = (\theta_0, \theta_1) \in \mathbb{R}^2$ of the sigmoid function is determined by maximizing the likelihood function on the training data,

$$L_{\mathcal{X}}(\theta) = \prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{1-y_i} \quad \text{with} \quad p_i = \frac{1}{1+\exp(\theta_0+\theta_1 x_i)}. \tag{3.13}$$

To reduce over-fitting and prevent θ_0 from becoming infinite when the training examples are perfectly separable, Platt proposed to use an out-of-sample data model by replacing y_i and $1-y_i$ by t_+ and t_- defined as

$$t_+ = \frac{n_+ + 1}{n_+ + 2} \quad \text{and} \quad t_- = \frac{1}{n_- + 2}, \tag{3.14}$$

where n_+ and n_- are respectively the number of positive and negative training samples. This ensures $L_{\mathcal{X}}$ to have a unique supremum $\hat{\theta} = (\hat{\theta}_0, \hat{\theta}_1)$. By formulating the logistic regression as a generalized linear model [56], normal approximation intervals can be used to compute a confidence interval over $h_s(\hat{\theta})$. The mass function (3.5) can then be used.

The likelihood function $L_{\mathcal{X}}$ can be used to define a plausibility function $Pl^{\Theta}_{\mathcal{X}}$ over the parameter $\theta \in \Theta$ as follows:

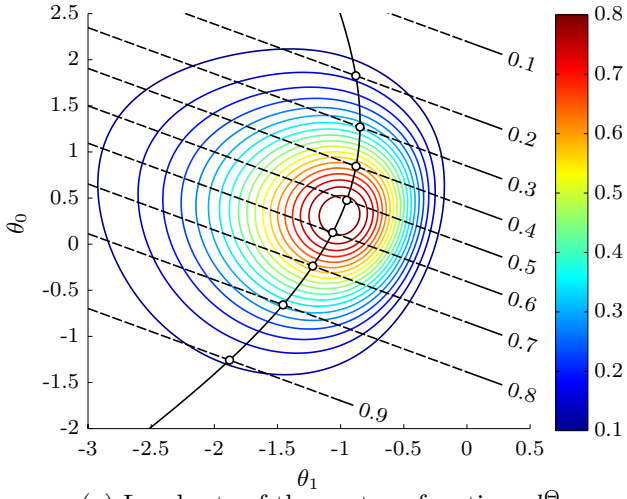
$$Pl^{\Theta}_{\mathcal{X}}(A) \;=\; \sup_{\theta\in A} pl^{\Theta}_{\mathcal{X}}(\theta), \quad \forall A \subseteq \Theta, \tag{3.15}$$

where

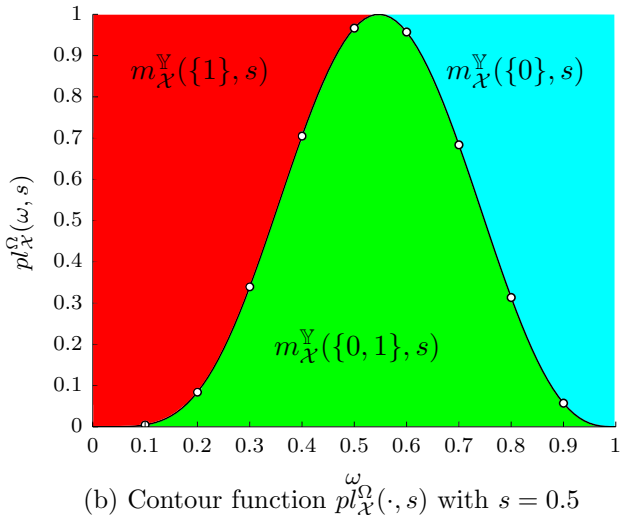
$$pl^{\Theta}_{\mathcal{X}}(\theta) \;=\; \frac{L_{\mathcal{X}}(\theta)}{L_{\mathcal{X}}(\hat{\theta})}, \quad \forall \theta \in \Theta, \tag{3.16}$$

After observing the score s of a test sample, its label $y \in \{0,1\}$ can be seen as the realisation of a random variable Y with a Bernoulli distribution $\mathcal{B}(\omega)$, where $\omega = h_s(\theta) \in [0,1]$. As described in Section 2.5.3, a predictive belief function $Bel^{\mathbb{Y}}_{\mathcal{X}}(\cdot, s)$ can be derived from the contour function $pl^{\Omega}_{\mathcal{X}}(\cdot, s)$. The function $pl^{\Omega}_{\mathcal{X}}(\cdot, s)$ can be computed from $Pl^{\Theta}_{\mathcal{X}}$ as

$$pl^{\Omega}_{\mathcal{X}}(\omega, s) \;=\; \begin{cases} 0 & \text{if } \omega \in \{0,1\} \\ Pl^{\Theta}_{\mathcal{X}}\left(h_s^{-1}(\omega)\right) & \text{otherwise,} \end{cases} \tag{3.17}$$



(a) Level sets of the contour function $pl^{\Theta}_{\mathcal{X}}$



(b) Contour function $pl^{\Omega}_{\mathcal{X}}(\cdot, s)$ with $s = 0.5$

Figure 3.3: Calibration results on the Australian dataset. (b) The three colored areas correspond to the predictive mass function $m^{\mathbb{Y}}_{\mathcal{X}}(\cdot, s)$.

where

$$h_s^{-1}(\omega) = \left\{ (\theta_0, \theta_1) \in \Theta \;\middle|\; \frac{1}{1 + \exp(\theta_0 + \theta_1 s)} = \omega \right\} \tag{3.18}$$

$$= \left\{ (\theta_0, \theta_1) \in \Theta \;\middle|\; \theta_0 = \ln\left(\omega^{-1} - 1\right) - \theta_1 s \right\}, \tag{3.19}$$

which finally yields

$$pl^{\Omega}_{\mathcal{X}}(\omega, s) = \sup_{\theta_1 \in \mathbb{R}} pl^{\Theta}_{\mathcal{X}}\left(\ln\left(\omega^{-1} - 1\right) - \theta_1 s, \theta_1\right), \quad \forall \omega \in (0, 1). \tag{3.20}$$

Figure 3.3 illustrates the computation of the predictive belief function $Bel^{\mathbb{Y}}_{x,s}$. Figure 3.3a shows level sets of the contour function $pl^{\Theta}_{\mathcal{X}}$ computed

from the scores of an SVM classifier trained on the Australian dataset. The value of $pl^{\Omega}_{\mathcal{X}}(\omega, s)$ is defined as the maximum value of $pl^{\Theta}_{\mathcal{X}}(\cdot, s)$ along the line $\theta_0 = \ln\left(\omega^{-1} - 1\right) - \theta_1 s$ represented by the doted lines. It can be approximated by a gradient descent algorithm. Figure 3.3b shows the contour function $pl^{\Omega}_{\mathcal{X}}(\cdot, s)$ from which $Bel^{\mathbb{Y}}_{\mathcal{X}}(\cdot, s)$ can be computed using Equations (2.49) and (2.50).

3.1.4 Discounting and keeping decision

As explained in Section 2.1.4, a mass function can be discounted by a factor $\delta \in [0, 1]$ representing the belief about its reliability. By cross-validation, an accuracy estimate can be computed from the training data. It is simply defined as the ratio of correct predictions over all the training data. It is thus a binomial proportion estimation as in the binning case. The methods presented in Section 3.1.1 returns a mass function about the accuracy. The belief of having a correct prediction is then used as a discounting factor. As illustrated on Figure 3.1a, the belief is low when the sample is small; discounting by it would thus result in a highly uncertain belief function as it is desired.

Another important aspect is the fact that calibration may change the final decision. In Figure 3.2 we can see that, after calibration, a zero score does not exactly lead to a probability of $1/2$. In the case of an SVM classifier, as only the sign of a score matters, one may argue that a positive score only supports the hypothesis of having a positive example but with more or less certainty depending on its value. In this sense, there should be no mass allocated to the singleton $\{0\}$ when the score is positive. Similarly, no mass should be assigned to the singleton $\{1\}$ when the score is negative.

A simple way to keep the decisions unchanged is to calibrate the positive and negative scores separately. For the positive scores, the mass assigned to $\{0\}$ is set to zero and added to the unknown state $\{0, 1\}$. A similar procedure is applied to the negative scores. In this way, the belief of having a success when the score is negative will always be zero while the plausibility of having a success when the score is positive will always be one.

3.2 Multi-class problem

Let us now consider a set of C classes $\Omega = \{\omega_1, \ldots, \omega_C\}$. Multi-class classifiers are often built by decomposing the initial problem into multiple binary classification ones. The two most common decompositions are the *one-vs-all* and *one-vs-one* strategies.

3.2.1 Multi-class probability from binary sub-problems

For a *one-vs-all* decomposition, a decision is made by choosing the binary classifier with the highest score. When the binary classifiers are calibrated, a simple normalization is a simple and efficient way to get a multi-class probability [139].

For a *one-vs-one* decomposition, majority vote is usually used for decision making. Getting a multi-class probability is more challenging that in the *one-vs-all* case. The output r_{ij} of a calibrated classifier trained on the classes $\{\omega_i, \omega_j\}$ can be seen as an estimate of $P(\omega = \omega_i | \omega \in \{\omega_i, \omega_j\}, \mathbf{x})$ for an observation $\mathbf{x}$ of class ω. From these r_{ij} it is necessary to estimate the multi-class probability $\mathbf{p}$ so that $\mathbf{p}(i) = P(\omega = \omega_i | \mathbf{x})$. Many approaches, based on different kinds of hypotheses, can be found in the literature. Wu et al. [129] compared several methods and proposed a more efficient one consisting in minimizing the following cost function:

$$\min_{\mathbf{p}} \sum_{i=1}^{C} \sum_{j \neq i} \left(r_{ji} \mathbf{p}(i) - r_{ij} \mathbf{p}(j) \right)^2 \quad \text{subject to} \quad \sum_{i=1}^{C} \mathbf{p}(i) = 1. \tag{3.21}$$

Again, it is important to note that the decisions made from the raw SVM scores and the ones made from the estimated probabilities may be different. A simple majority vote with raw SVM scores is actually a very competitive strategy and may, in certain cases, perform better than any probabilistic calibration methods. Wu et al. [129] reported that the higher the number of classes was the smaller improvement was observed from their probabilistic combination. In their experiments with the *letter* dataset (26 classes), the majority vote on the raw SVM scores out-performed by at least 3% in accuracy all other approaches. But in their experiments, they used a fixed number of training samples 300 and 800, which would lead to have about 20 and 60 training samples for each pairwise binary classifier. This will result in a calibration of high uncertainty which may explain the decrease in performance. Therefore the calibration of a classifier not always improves the performance of a single binary or multi-class classifier but may still be necessary if several classifiers are to be combined.

3.2.2 From binary to multi-class belief functions

Just as in the probabilistic case, it is possible to derive a multi-class belief function from a binary decomposition. Quost et al. proposed an evidential combination for both *one-vs-all* [96] and *one-vs-one* [97] decompositions.

Let m_{ij} be the mass function obtained from a classifier trained on $\Omega_{ij} = \{\omega_i, \omega_j\}$ through a *one-vs-one* decomposition. This mass function can be seen as an estimate of a multi-class mass function m^{Ω} conditioned on the knowledge that the true label is in Ω_{ij}. The conditional mass function given

$B \subseteq \Omega$, noted $m^{\Omega}[B]$, is defined as

$$m^{\Omega}[B](A) = \begin{cases} \sum_{C \cap B = A} m^{\Omega}(C) & \text{if } A \subseteq B, \\ 0 & \text{otherwise.} \end{cases} \tag{3.22}$$

This conditional mass function is not always normalized, that is $m^{\Omega}[B](\emptyset)$ may not be null. The normalized conditional mass function $m^{\Omega}[B]^*$ can be obtained by

$$m^{\Omega}[B]^*(A) = \begin{cases} \frac{m^{\Omega}[B](A)}{1 - m^{\Omega}[B](\emptyset)} & \text{if } A \subseteq \Omega,\ A \neq \emptyset, \\ 0 & \text{if } A = \emptyset. \end{cases} \tag{3.23}$$

The goal is then to find a mass function m^{Ω} whose conditioning on each Ω_{ij} would lead to m_{ij}. This is equivalent to have a mass function satisfying

$$m^{\Omega}[\Omega_{ij}](A) = m_{ij}(A)(1 - m^{\Omega}[\Omega_{ij}](\emptyset)), \quad \forall A \in 2^{\Omega_{ij}} \backslash \emptyset,\ \forall i > j \tag{3.24}$$

The constraints (3.24) actually define a system of $3C(C-1)/2$ linear equations with $2^K - 2$ unknowns. Unfortunately, this system rarely has a solution. Quost et al. [97] thus suggested to have an approximate solution by minimizing the following quadratic problem:

$$\min_{m^{\Omega}} \sum_{i>j} \sum_{\emptyset \neq A \subseteq \Omega} \left(m^{\Omega}\left[\Omega_{ij}\right](A) - m_{ij}(A) \left(1 - m^{\Omega}\left[\Omega_{ij}\right](\emptyset)\right)\right)^2, \tag{3.25}$$

subject to

$$m^{\Omega}(A) \geq 0, \quad \forall A \subseteq \Omega,\ A \neq \emptyset, \tag{3.26a}$$

$$m^{\Omega}(\emptyset) = 0, \tag{3.26b}$$

$$\sum_{A \subseteq \Omega} m^{\Omega}(A) = 1. \tag{3.26c}$$

A similar approach can also be used in the case of a *one-vs-all* decomposition [96].

3.3 Experimental evaluations

Experimental evaluations were conducted on several binary and multi-class classification problems from UCI. For each dataset, three independent classifiers were trained on non-overlapping subsets of different sizes. Two of them were trained using a fixed number of data while the third one was trained with a variable number of data. Table 3.1 shows the number of samples used to train and test each classifier. For each experiment, a 5-fold

Table 3.1: Number of samples using for training and testing on different datasets from UCI.

Dataset	Classes	Train #1	Train #2	Train #3	Test
autralian	2	30	70	10–200	390
diabetes	2	30	70	10–200	468
heart	2	20	40	10–140	70
ionosphare	2	20	40	10–190	101
liver-disorders	2	20	40	10–190	95
sonar	2	20	40	10–90	58
dna	3	200	400	100–700	1000
MNIST	10	200	400	100–700	1000
satimage	6	200	400	100–700	1000
segment	7	200	400	100–700	1000
USPS	10	200	400	100–700	1000

cross validation was conducted to get both the score data and the accuracy estimates. The whole process was repeated for 100 rounds on each dataset. The LibSVM[2] library [14] was used to learn the base classifiers.

The results for binary classification are shown in Table 3.2, 3.3 and 3.4. In the case of relatively few training data, when the decisions are allowed to change through calibration, majority vote from raw SVM scores almost always gave better results than probabilistic or evidential calibration methods. This can be explained by the fact that the calibration step can be of very poor quality and may lead to high bias on the decision value. Only when the decisions were kept, that is the sign of the raw SVM scores were still considered, the calibrated classifiers managed to reach performances close or better than majority vote. Especially, logistic regression, which was the least prone to over-fitting, performed almost always better than majority vote when the decisions were kept unchanged.

When the third classifier was trained with much more data than the two others, the combination of the three tended to decrease the performance of the third one, which was the best out of the three. In the case of majority vote, the combination always lead to worse results than the best single classifier. This could be expected as the lower accuracies of the first two classifiers were not taken into account. The use of a calibrated methods without decision keeping could in some cases out-perform majority vote, but it was not guaranteed. However, they hardly reached the performance of the best single classifier. The best results were obtained when the decisions were kept unchanged, the results were then close or better than the best single classifier. Compared to majority vote, the gain could go up to 3% in accuracy.

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvm

The use of belief functions other than the inverse pignistic transformation gave better results than probabilistic calibrations. Especially when the classifiers were trained with an unbalanced amount of data, in which case it was important to model the uncertainty relative to the calibration step. The inverse pignistic based calibration gave results similar to probabilistic calibration but could be used to keep the decisions unchanged which lead to improvements. For both binning and isotonic regression, Dempster's model gave the best results closely followed by confidence interval based model. For logistic regression, the likelihood based calibration approach reached the highest accuracy. Overall, the best results were obtained by using a logistic regression calibration with a likelihood-based discounted belief function while keeping the decisions unchanged.

In the case of multi-class classification, if the class distribution in the training data is uniform, each binary classifier from a *one-vs-one* decomposition are trained with a similar amount of data. In such a case, the evidential combination proposed by Quost et al. [96] was shown to perform as well as the probabilistic approach from Wu et al. [129]. To show the potential gain from the evidential approach, we transformed the multi-class classification shown on Table 3.1 into three class problems by keeping two classes and grouping the rest. Table 3.5 shows the results obtained with the multi-class classification. Only logistic regression has been considered as it has been shown to perform better than both binning and isotonic regression. While considering single multi-class classifiers, the use of evidential calibration while keeping the decisions gave the best performances except for the *dna* dataset. For the *dna* dataset, which was originally a three class classification problem, the class distribution was actually uniform. The same conclusions were reached when combining the three multi-class classifiers although the results from the probabilistic and evidential approaches were very close.

One drawback of the evidential calibration methods is the higher computational cost, especially when dealing with a high number of classes. Solving (3.25) may be problematic when the number of classes is large. The use of one-class classifier can, however, help to keep it tractable [97].

3.4 Conclusion

In this chapter, we showed how to extend three classical probabilistic calibration methods using belief functions. Belief functions can better represent the uncertainty of the calibration procedure, especially when few data are available. It also allows to keep the decisions of the original classifiers unchanged. We applied our calibration to SVM classifiers but the calibration can actually be used with any classifier. In the next chapter, we will exploit these calibration methods to combine multiple pedestrian detectors which

use different types of classifier. We will also show that in the particular case of pedestrian detection, keeping the decisions of the classifiers unchanged becomes crucial.

Table 3.2: Results using binning. The second line (# Samples) refers to the number of samples used to train the third classifier. The methods marked by an asterisk are those using an additional discounting. Underlined bold figures are the best results, bold ones are the second best.

	Dataset	australian			diabetes			heart			ionosphere			liver-disorders			sonar		
	# Samples	10	50	200	10	50	200	10	30	140	10	30	190	10	30	190	10	30	90
Raw	Vote	**84.2**	**84.6**	85.9	73.1	**74.1**	75.7	**80.8**	**81.6**	82.7	81.1	86.4	91.4	**62.9**	**65.2**	69.1	72.9	**77.5**	81.3
	Best single	83.0	83.0	86.1	**73.5**	73.5	**76.6**	79.9	79.9	**83.3**	86.6	86.6	93.6	**62.9**	62.9	**71.1**	**76.4**	76.4	**84.7**
Prob.	Product	74.1	82.4	83.5	64.9	70.1	72.2	68.6	77.7	78.7	72.2	83.7	87.2	56.1	57.1	60.1	56.4	69.0	74.5
	Sum	80.0	83.5	85.4	66.5	70.8	73.4	74.1	78.7	80.9	75.2	84.1	91.0	54.9	56.4	61.3	60.6	71.1	77.7
	Weighted Sum	81.1	83.6	85.4	67.2	70.9	73.9	75.5	79.0	81.0	77.8	84.9	92.4	54.7	57.0	63.1	62.9	71.6	79.1
Belief function	Inv. Pign.	74.1	82.5	83.5	64.9	70.2	72.4	68.6	77.7	78.7	72.2	83.7	87.2	56.1	57.1	60.1	56.4	69.0	74.5
	Inv. Pign.*	81.4	83.5	85.4	67.9	71.1	74.1	76.3	79.0	81.0	79.6	84.7	93.0	54.9	57.3	64.0	65.4	72.1	80.2
	Dempster	82.0	83.8	85.9	68.3	70.9	74.4	76.3	78.9	82.1	79.7	85.6	93.7	55.5	57.1	65.1	65.1	72.0	80.8
	Dempster*	82.6	83.8	86.0	69.5	71.5	75.2	77.1	79.0	82.2	83.1	85.8	93.6	56.2	57.8	67.1	67.7	72.3	82.0
	Conf. Int.	82.3	83.9	**86.2**	69.7	71.4	75.6	76.6	78.7	82.5	84.5	86.2	**93.9**	56.8	57.4	69.0	68.9	71.9	82.9
	Conf. Int.*	82.4	83.9	86.1	70.1	71.8	75.8	76.7	78.8	82.5	84.8	86.7	93.8	56.8	57.8	69.3	69.3	72.3	82.9
	Likelihood	81.3	83.6	85.8	67.6	70.7	74.3	76.2	79.0	81.8	79.5	85.4	93.5	54.7	56.6	63.6	63.7	71.7	80.2
	Likelihood*	82.6	83.7	85.9	69.4	71.4	75.1	77.1	79.2	82.0	82.6	85.6	93.5	56.1	57.5	66.8	67.5	72.6	81.7
Keep decision	Inv. Pign.	81.0	83.7	84.4	70.1	72.8	74.5	76.6	79.6	80.9	76.1	85.5	89.6	59.4	61.9	66.0	66.5	74.4	79.1
	Inv. Pign.*	83.1	84.2	85.6	71.1	72.9	75.1	78.8	80.1	82.1	81.5	85.7	93.2	59.0	61.7	67.3	69.5	75.6	81.7
	Dempster	**84.0**	**84.7**	**86.2**	73.0	**73.9**	76.0	**80.4**	**81.4**	83.0	84.2	88.2	**93.9**	**61.9**	**64.0**	70.0	73.9	**77.5**	83.6
	Dempster*	**84.0**	**84.7**	**86.2**	**73.5**	**73.9**	76.2	80.3	81.2	83.2	85.9	88.4	93.8	61.6	**64.0**	70.6	73.6	**77.2**	84.0
	Conf. Int.	83.6	**84.6**	**86.3**	73.3	73.6	**76.5**	79.7	80.5	**83.4**	**87.4**	**88.7**	**94.0**	61.8	62.9	70.8	74.3	76.6	**84.7**
	Conf. Int.*	83.5	84.5	**86.2**	**73.4**	73.7	**76.5**	79.5	80.5	**83.4**	**87.6**	**88.6**	**93.9**	61.8	62.9	**71.0**	**74.4**	76.9	**84.6**
	Likelihood	83.7	84.4	86.1	72.5	73.6	75.9	79.9	80.9	83.0	83.3	88.2	93.7	60.9	63.2	69.2	72.5	76.9	83.1
	Likelihood*	83.9	84.5	**86.3**	73.0	73.6	76.2	79.8	80.8	83.0	84.9	88.1	93.7	61.1	63.3	70.2	72.9	77.0	83.8

Table 3.3: Results using isotonic regression. The second line (# Samples) refers to the number of samples used to train the third classifier. The methods marked by an asterisk are those using an additional discounting. Underlined bold figures are the best results, bold ones are the second best.

	Dataset	australian			diabetes			heart			ionosphere			liver-disorders			sonar		
	# Samples	10	50	200	10	50	200	10	30	140	10	30	190	10	30	190	10	30	90
Raw	Vote	**84.2**	**84.6**	85.9	73.1	**74.1**	75.7	**80.8**	**81.6**	**82.7**	81.1	86.4	91.4	**62.9**	**65.2**	69.1	72.9	77.5	81.3
	Best single	83.0	83.0	86.1	**73.5**	73.5	**76.6**	79.9	79.9	**83.3**	86.6	86.6	93.6	**62.9**	62.9	**71.1**	**76.4**	76.4	**84.7**
Prob.	Product	81.9	83.9	84.9	70.1	73.1	75.0	78.1	80.0	81.0	88.0	91.0	92.7	59.6	62.5	67.9	73.1	76.8	81.6
	Sum	82.9	84.3	85.5	71.0	73.3	75.4	79.3	80.4	81.5	86.1	89.1	92.5	59.9	62.5	68.3	73.3	77.4	82.5
	Weighted Sum	83.1	84.4	85.7	71.1	73.4	75.6	79.2	80.4	81.6	86.2	89.1	92.8	59.9	62.5	68.3	73.2	77.2	82.7
Belief function	Inv. Pign.	81.9	84.0	85.0	70.2	73.0	74.9	78.0	80.1	80.9	87.7	**91.1**	92.8	59.5	62.5	67.8	73.0	76.9	81.7
	Inv. Pign.*	82.9	84.2	85.7	71.4	73.2	75.5	78.9	80.3	81.5	86.4	89.1	93.4	60.6	62.4	68.4	73.4	76.9	83.2
	Dempster	83.1	84.3	85.6	71.6	73.1	75.5	79.4	80.5	81.7	87.7	90.1	93.5	60.7	63.0	68.8	73.6	77.5	83.3
	Dempster*	83.3	84.3	85.9	71.7	73.3	75.8	79.4	80.2	81.8	87.4	89.1	93.7	61.1	62.7	68.9	73.1	77.0	83.3
	Conf. Int.	83.0	84.1	85.9	71.2	72.8	75.6	79.7	80.4	82.0	88.2	90.3	93.6	60.0	61.9	69.2	72.2	76.6	83.1
	Conf. Int.*	83.1	84.2	85.9	71.6	73.1	75.9	79.2	80.3	82.0	87.6	89.5	**93.9**	60.3	61.8	69.6	72.0	76.1	83.3
	Likelihood	82.9	84.2	85.7	71.3	73.0	75.5	79.6	80.5	81.8	88.2	90.3	93.4	60.2	62.9	68.8	73.4	77.5	83.1
	Likelihood*	83.4	84.2	85.8	71.5	73.3	75.7	79.3	80.2	81.9	87.5	89.4	93.6	60.7	62.5	68.9	73.3	77.0	83.3
Keep decision	Inv. Pign.	82.3	84.1	85.0	71.0	73.4	75.2	78.7	80.6	81.3	88.1	**91.7**	93.2	60.5	62.9	68.3	73.8	77.5	82.0
	Inv. Pign.*	83.6	84.3	85.8	72.0	73.6	75.7	79.8	81.0	81.9	87.2	90.2	93.7	61.0	62.9	68.9	74.1	77.6	83.5
	Dempster	**84.2**	**84.6**	**86.1**	72.9	**74.1**	76.1	**80.3**	**81.5**	82.5	88.2	91.0	93.7	**62.4**	**64.7**	69.7	74.5	**78.3**	83.6
	Dempster*	**84.2**	**84.7**	**86.1**	**73.3**	**74.1**	76.1	80.2	81.1	82.6	87.3	90.1	**93.8**	62.1	64.6	70.3	74.4	78.0	83.7
	Conf. Int.	83.7	84.4	**86.1**	72.7	73.5	76.1	**80.3**	81.2	**82.7**	**88.5**	90.8	93.7	**62.4**	63.3	**70.5**	74.0	77.0	83.7
	Conf. Int.*	83.8	84.5	**86.2**	73.1	73.6	**76.3**	80.1	81.0	82.6	88.2	89.9	**93.9**	61.7	63.0	70.4	73.2	76.9	**84.2**
	Likelihood	83.7	84.5	85.8	72.6	73.8	76.0	80.1	81.3	82.2	**88.6**	**91.1**	93.7	61.8	64.3	69.6	**74.8**	**78.1**	83.4
	Likelihood*	**84.0**	**84.6**	**86.1**	73.1	**73.9**	76.2	79.9	81.1	82.4	87.8	90.4	93.7	61.7	64.2	70.1	74.5	77.8	83.9

Table 3.4: Results using logistic regression. The second line (# Samples) refers to the number of samples used to train the third classifier. The methods marked by an asterisk are those using an additional discounting. Underlined bold figures are the best results, bold ones are the second best.

	Dataset	australian			diabetes			heart			ionosphere			liver-disorders			sonar		
	# Samples	10	50	200	10	50	200	10	30	140	10	30	190	10	30	190	10	30	90
Raw	Vote	84.2	84.6	85.9	73.1	74.1	75.7	**80.8**	81.6	82.7	81.1	86.4	91.4	**62.9**	**65.2**	69.1	72.9	77.5	81.3
	Best single	83.0	83.0	86.1	**73.5**	73.5	**76.6**	79.9	79.9	**83.3**	86.6	86.6	93.6	**62.9**	62.9	**71.1**	**76.4**	76.4	84.7
Prob.	Product	83.8	84.6	86.1	69.5	70.7	74.9	79.9	81.7	82.8	83.0	89.6	93.8	57.0	60.1	65.7	70.5	**78.7**	84.4
	Sum	83.8	84.7	86.1	69.6	70.6	74.9	80.3	81.7	82.8	83.3	89.6	93.5	57.3	60.2	65.9	70.9	78.4	84.2
	Weighted Sum	84.0	84.7	86.1	69.8	70.7	75.2	80.0	81.5	82.8	84.2	89.6	93.6	58.0	60.5	66.7	71.9	78.5	84.4
Belief func- tion	Inv. Pign.	82.4	84.4	85.8	69.1	71.2	74.8	78.3	81.0	82.4	83.4	**90.2**	93.5	56.9	60.0	65.3	69.1	**78.7**	84.0
	Inv. Pign.*	83.5	84.6	86.0	70.2	71.3	75.5	79.6	80.8	82.6	85.7	89.4	93.9	58.6	60.8	67.2	72.6	78.5	84.4
	Conf. Int.	83.7	84.4	86.2	70.1	70.9	75.7	80.3	81.3	**83.2**	**87.0**	89.3	93.2	59.7	60.5	67.8	**75.9**	**78.7**	**85.4**
	Conf. Int.*	83.7	84.6	**86.3**	70.5	71.2	76.0	80.3	81.1	83.1	86.8	89.2	93.4	59.9	60.9	68.9	75.4	78.2	**85.3**
	Likelihood	84.0	84.7	86.2	69.6	70.5	75.4	80.3	**81.8**	**83.2**	84.2	89.3	93.9	58.1	60.2	66.8	71.8	78.5	84.9
	Likelihood*	84.0	**84.8**	86.2	70.2	70.8	75.9	80.0	81.7	**83.2**	85.3	89.3	93.9	59.0	60.7	68.3	73.1	78.5	85.1
Keep decision	Inv. Pign.	83.0	84.5	85.9	71.1	72.8	75.7	79.2	81.1	82.5	85.4	**90.4**	93.5	60.2	62.8	67.7	73.3	**79.2**	84.3
	Inv. Pign.*	83.6	84.7	86.0	71.6	72.8	76.1	80.0	80.9	82.6	85.9	89.8	93.9	60.9	62.8	68.4	73.6	**78.7**	84.5
	Conf. Int.	83.7	84.7	**86.4**	73.1	73.6	76.3	80.5	81.6	**83.2**	**87.4**	**90.2**	92.9	62.4	63.7	70.6	75.7	78.5	84.5
	Conf. Int.*	83.6	84.7	**86.4**	**73.4**	73.7	**76.4**	80.2	81.5	83.1	86.7	89.3	93.2	61.8	63.6	**70.8**	75.5	78.2	84.8
	Likelihood	**84.5**	**84.9**	**86.3**	73.2	**74.0**	76.3	**80.7**	**81.9**	83.0	84.0	88.8	**94.0**	**62.6**	**64.5**	70.1	73.9	78.3	84.1
	Likelihood*	**84.3**	**84.8**	**86.3**	**73.5**	**74.0**	76.2	80.5	81.7	**83.2**	85.3	88.9	**94.1**	62.1	64.4	70.5	73.7	78.3	84.5

Table 3.5: Results using logistic regression for multi-class classification. The second line (# Samples) refers to the number of samples used to train the third classifier. The methods marked by an asterisk are those using an additional discounting. Underlined bold figures are the best results, bold ones are the second best. The upper part (Single) corresponds the performances of the third classifier alone while the lower part (Multiple) is the results from the combination of the three classifiers.

	Dataset	dna			MNIST			satimage			segment			USPS		
	# Samples	50	150	500	50	150	500	50	150	500	50	150	500	50	150	500
Single	Wu et al. [129]	**74.3**	**86.6**	**92.0**	**25.4**	**29.4**	31.0	**50.9**	**52.1**	**52.9**	**41.7**	**43.3**	**43.5**	36.6	**39.2**	**39.9**
	Quost et al. [96]	**73.8**	**86.6**	**92.0**	25.2	**29.4**	31.0	50.6	52.0	**52.9**	41.1	**43.3**	**43.5**	36.4	39.1	**39.9**
	Quost*	73.7	**86.6**	**92.0**	25.1	**29.3**	31.0	50.6	52.0	**52.9**	40.9	**43.3**	**43.5**	36.3	39.1	**39.9**
	Quost + Keep	72.1	85.5	91.6	**25.5**	**29.4**	31.0	**51.8**	**52.6**	**53.2**	**42.8**	**43.5**	**43.6**	**37.5**	**39.5**	**40.1**
	Quost*+ Keep	72.4	85.6	91.6	**25.5**	**29.4**	31.0	**51.8**	**52.6**	**53.2**	**42.7**	**43.5**	**43.6**	**37.4**	**39.5**	**40.1**
Multiple	Wu et al. [129]	89.6	**90.7**	**92.3**	**29.0**	**29.9**	30.5	**52.5**	**52.7**	**52.9**	**43.4**	**43.5**	**43.5**	**39.1**	**39.5**	**39.7**
	Quost et al. [96]	**89.8**	**90.7**	**92.5**	**29.0**	29.7	30.5	52.4	52.5	**52.9**	**43.3**	**43.5**	**43.5**	39.0	**39.4**	**39.7**
	Quost*	**89.9**	**90.7**	**92.5**	**29.0**	29.7	30.5	52.4	52.5	**52.9**	**43.3**	**43.5**	**43.5**	**39.1**	**39.4**	**39.7**
	Quost + Keep	88.0	89.8	92.2	**29.2**	**29.8**	30.5	**52.7**	**52.8**	**53.0**	**43.4**	**43.5**	**43.6**	**39.2**	**39.5**	**39.8**
	Quost*+ Keep	88.0	89.7	**92.3**	**29.2**	**29.8**	30.5	**52.7**	**52.8**	**53.0**	**43.4**	**43.5**	**43.6**	**39.2**	**39.5**	**39.8**

Chapter 4

Combination of pedestrian detectors

Pedestrian detection is an important issue in the development of safe intelligent vehicles. In computer vision, it is the most studied case of object detection. There exist many pedestrian datasets; INRIA [20], ETH [42], TUD-Brussels [128] and Caltech Pedestrian Detection Benchmark [33] are among the most popular ones. The last one is the largest. More than 30 state-of-the-art detectors have been tested on it and their outputs are publicly available. Moreover, the high diversity of the evaluated methods makes their combination an ever more interesting issue. Table 4.1 lists the detectors evaluated on the Caltech dataset.

Diversity, and thus potential complementarity of the detectors exists because of mainly three reasons. The first one is related to the features used to represent pedestrians. Haar-like features [118], shapelets [101], shape context [81] and histograms of oriented gradient (HOG) [20] features are commonly used. The last one is the most popular and almost all detectors use it in some forms. Wojek and Schiele [127] concatenated all the previously mentioned features and trained a new model outperforming all individual ones. Other features such as local binary pattern (LBP) [123] or motion features [119] were also considered in addition to HOG. However, even though the HOG feature is used in these methods, it is not guaranteed that a pedestrian detected by the original 'HOG' detector [20] would still be detected by the other methods. Nevertheless, the use of multiple types of features as in [31, 30, 92] or features learned in very large spaces [32, 6] have led to significant improvements.

The second source of diversity comes from the classifier. Linear SVM and AdaBoost are often considered. The use of latent variables in SVM has been popularized by Felzenszwalb *et al.* [47] for part-based approaches. Non-linear SVM [77], Partial Least Squares analysis [102] or boosting optimizing directly the area under the receiver operating characteristic (ROC) curve [90]

Table 4.1: List of algorithms evaluated on the Caltech Pedestrian Benchmark. This table only lists the algorithms that were submitted before 2014.

#	Algorithm	Features	Classifier	Training
1	'VJ' [118]	Haar	AdaBoost	INRIA
2	'HOG' [20]	HOG	linear SVM	INRIA
3	'HikSvm' [77]	HOG	HIK SVM	INRIA
4	'LatSvm-V1' [47]	HOG	latent SVM	PASCAL
5	'LatSvm-V2' [47]	HOG	latent SVM	INRIA
6	'MultiResC' [91]	HOG	latent SVM	Caltech
7	'MultiResC+2Ped' [88]	HOG	latent SVM	Caltech
8	'MT-DPM' [133]	HOG	latent SVM	Caltech
9	'MT-DPM+Context' [133]	HOG	latent SVM	Caltech
10	'PoseInv' [76]	HOG	AdaBoost	INRIA
11	'MLS' [83]	HOG	AdaBoost	INRIA
12	'DBN-Isol' [87]	HOG	DeepNet	INRIA
13	'DBN-Mut' [89]	HOG	DeepNet	INRIA/Caltech
14	'HOG-LBP' [123]	HOG+LBP	linear SVM	INRIA
15	'MOCO' [15]	HOG+LBP	latent SVM	Caltech
16	'pAUCBoost' [90]	HOG+COV	pAUCBoost	INRIA
17	'FtrMine' [32]	channels	AdaBoost	INRIA
18	'ChnFtrs' [31]	channels	AdaBoost	INRIA
19	'FPDW' [30]	channels	AdaBoost	INRIA
20	'CrossTalk' [29]	channels	AdaBoost	INRIA
21	'Roerei' [9]	channels	AdaBoost	INRIA
22	'ACF' [31]	channels	AdaBoost	INRIA
23	'ACF-Caltech' [31]	channels	AdaBoost	Caltech
24	'ACF+SDt' [92]	channels	AdaBoost	Caltech
25	'MultiFtr' [127]	multiple	AdaBoost	INRIA
26	'MultiFtr+CSS' [119]	multiple	linear SVM	TUD-Motion
27	'MultiFtr+Motion' [119]	multiple	linear SVM	TUD-Motion
28	'MF+Motion+2Ped' [88]	multiple	linear SVM	TUD-Motion
29	'FeatSynth' [6]	multiple	linear SVM	INRIA
30	'AFS' [74]	multiple	linear SVM	INRIA
31	'AFS+Geo' [74]	multiple	linear SVM	INRIA
32	'Pls' [102]	multiple	PLS+QDA	INRIA
33	'Shapelet' [101]	gradients	AdaBoost	INRIA
34	'ConvNet' [103]	pixels	DeepNet	INRIA

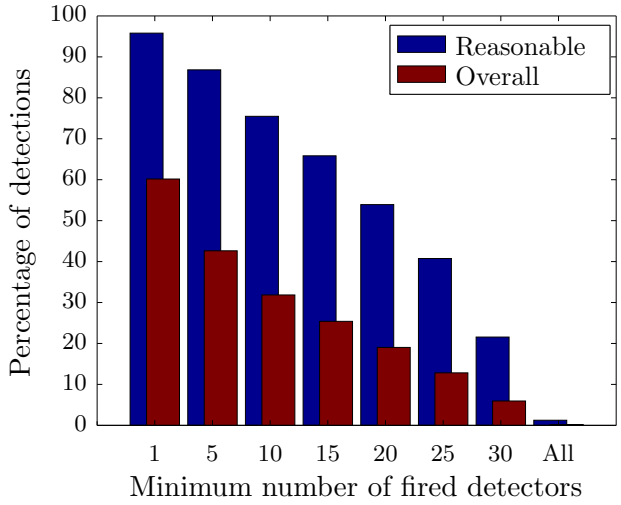


Figure 4.1: Percentage of detected pedestrians by at least $k \in \{1, 5, \ldots, 34\}$ detectors at 1 FPPI. The detections were done on the Caltech-Test dataset with the "Reasonable" and "Overall" scenarios.

were also used. More recently, deep learning was also considered [87, 89, 103]. Finally, the choice of the training data, if not the same for all detectors, is an additional source of diversity.

Different forms of detectors combination can be found in the literature. The use of multiple sensors in robotics has often led to the combination of several detectors. The easiest way is to use a first weak detector to gather a set of regions of interest, which are then more deeply analyzed by a more efficient one. The 'FeatSynth' [6] algorithm actually only processes the detections returned by 'FtrMine' [32]. Some works make use of other object detectors such as cars [133] or 2-pedestrians detectors [88]. Recently, Denoeux *et al.* [28] applied an optimal object association algorithm to combine the outputs of two object detectors in polynomial time. However, the optimal association problem with more than two detectors is NP-hard.

To figure out the potential gain from combining multiple detectors, we show in Figure 4.1 some detection statistics for the Caltech dataset. We can see that, at one False Positive Per Image (FPPI), more than 95% of the pedestrians in the "Reasonable" scenario were detected by at least one detector. The "Reasonable" scenario corresponds to pedestrians over 50 pixels tall and with an occlusion rate lower than 35%. As a comparison, the currently best performing algorithm ('ACF+SDt' [92]) has a recall of about 80% at 1 FPPI. Similarly, in the "Overall" scenario where all the pedestrians were considered, about 60% of the pedestrians were detected by at least one detector. The 'MT-DPM+Context' [133] algorithm, which outperforms 'ACF+SDt' in this scenario, hardly reached a 40% recall. The

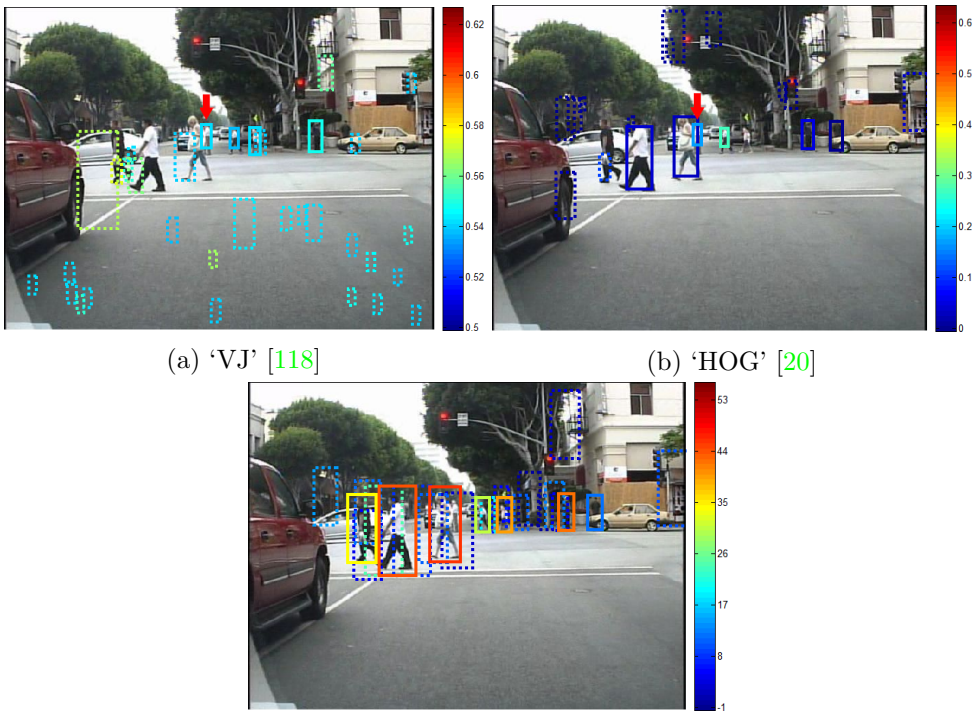
potential gain of combining in a proper way all those detectors is thus fairly significant.

In this thesis, a pedestrian detector is seen as a *black box* which returns a set of bounding boxes (BBs) with associated scores as illustrated in Figure 4.2. As explained in Section 1.2.2, there exist two main combination strategies: trainable and non-trainable combinations. For detection problems, trainable approaches are limited in several aspects. Contrary to classification problems, the information retrieved from a set of classifiers, for a given entity, may only be partial in a detection context. A BB returned by one detector may not be detected by other detectors. Most trainable approaches need the responses of all the detectors to reach a final decision. One possible way to cope with such issues would be to use a common sliding windows exploration for all the detectors and combine their outputs at each position and scale. However, that would be very computationally demanding and many detection algorithms actually do not use exhaustive sliding windows search. Finally, using trainable combination methods would imply new training every time a new detector needs to be combined. In the pedestrian detection field, where many new algorithms are designed every year, that may be seen as an important drawback. Therefore, in this thesis, we will focus on non-trainable combination approaches.

In this chapter, we propose a combination framework that models the outputs of the detectors with the theory of belief functions and combines them with a pre-defined rule. In Section 4.1 we describe how the BBs returned by multiple detectors are associated, calibrated and combined. Then, in Section 4.2 we compare different combination methods using the Caltech dataset.

4.1 Combination of bounding boxes

The outputs of most pedestrian detectors are given as bounding boxes. To each of them is associated a score representing the confidence of the detector. The range of these scores depends on the features and the classifier used for detection. Figure 4.2 shows some detection results from three algorithms, applied to one particular image frame. The 'VJ' algorithm gives pretty poor results with a very high false detection rate. Even worse, the BBs with the highest scores are actually false positives. The 'HOG' algorithm gives relatively good results with few false positives. It can be noticed that the two detected pedestrians in the foreground have very low scores. The 'ACF+SDt' algorithm is the one with the highest recall. Even though it returns more false positives than 'HOG', most of the true positives have a higher score than the false negatives. It is, however, interesting to notice that on the particular image shown in Figure 4.2, the only pedestrian missed by 'ACF+SDt' was actually detected by both the 'VJ' and 'HOG' algorithms.



(a) 'VJ' [118] (b) 'HOG' [20]

(c) 'ACF+SDt' [92]

Figure 4.2: Pedestrian detection results from three algorithms. The colour of the bounding boxes represents the score. Solid boxes are true positives and doted boxes are false positives. The red arrow points to a pedestrian detected by both 'VJ' and 'HOG' but not 'ACF+SDt'.

4.1.1 Clustering of bounding boxes

In a sliding windows approach, a single pedestrian is often detected at several nearby positions and scales. A non-maximal suppression (NMS) step is often needed in order to select only one BB per pedestrian. In our context, the same issue occurs but instead of having multiple detections from a single detector they are returned by several ones. As reported by Dollár *et al.* [33], there exist two dominant NMS approaches: mean shift mode estimation [20] and pairwise maximum suppression [47]. For the former it is necessary to define a covariance matrix representing the uncertainty in position and size of the BBs. This can be difficult considering the high variety of detectors. Felzenszwalb *et al.* [47] proposed a simpler way by suppressing the least confident of any pair of BBs that overlap sufficiently. Given two bounding boxes BB_i and BB_j, their area of overlap is defined as follows:

$$a_{\text{union}} = \frac{\text{area}(BB_i \cap BB_j)}{\text{area}(BB_i \cup BB_j)}. \tag{4.1}$$

Dollár *et al.* (see addendum to [31]) proposed to replace the above definition with:

$$a_{\min} = \frac{\text{area}(BB_i \cap BB_j)}{\min(\text{area}(BB_i), \text{area}(BB_j))}. \tag{4.2}$$

Using a_{union} or $a_{\min}$ as a distance measure between BBs, a simple hierarchical clustering can be used to group them until the overlap exceeds a certain threshold. The distance between two clusters is defined as the maximum distance between every pairs of BBs. This guarantees that, within a cluster, the overlapping area between two BBs is always sufficient. Dollár *et al.* [31] showed that proceeding greedily leads to the best results. They processed the detections in decreasing order of scores; when two BBs are associated, the one with the lowest score would no longer be used for further associations. In our clustering formulation, this last point is equivalent to defining the distance between two clusters as the distance between their respective highest-scored BBs.

4.1.2 Calibration and combination

The BBs greedy clustering method presented in the previous section supposes that the scores returned by the detectors are comparable. In practice, it is rarely the case. The detectors need to be calibrated first. All the calibration methods proposed in the previous chapter can be used. Figure 4.3a shows the isotonic and logistic regression based calibration of the 'HOG' detector. One particularity of object detection is the relatively high false positive rate. For example with the 'HOG' algorithm, more than 99% of the detections have a score less than 0.1, as illustrated on Figure 4.3b. Less than 0.1% of these detections are true positives. As a result, most detections have an associated probability lower than 0.1. From a Bayesian perspective, multiple sources of information returning low probabilities would actually lead to an even lower one. This would also be the case with belief functions expect if we consider that the detected BBs only support the presence of pedestrians (see Section 3.1.4). Thus only simple mass functions of the form $\{1\}^{\alpha}$, with $\alpha \in [0, 1]$, are used as outputs of the evidential calibration.

The combination of two simple mass functions $\{1\}^{\alpha_1}$ and $\{1\}^{\alpha_2}$ can be expressed very easily. Their combination with Dempter's rule $\oplus$, the cautious rule $\textcircled{\wedge}$ and Frank's family of t-norms based rule $\bigoplus_s$ are defined as

$$\{1\}^{\alpha_1} \oplus \{1\}^{\alpha_2} = \{1\}^{\alpha_1 \alpha_2}, \tag{4.3a}$$

$$\{1\}^{\alpha_1} \textcircled{\wedge} \{1\}^{\alpha_2} = \{1\}^{\alpha_1 \wedge \alpha_2}, \tag{4.3b}$$

$$\{1\}^{\alpha_1} \bigoplus_s \{1\}^{\alpha_2} = \{1\}^{\alpha_1 \top_s \alpha_2}. \tag{4.3c}$$

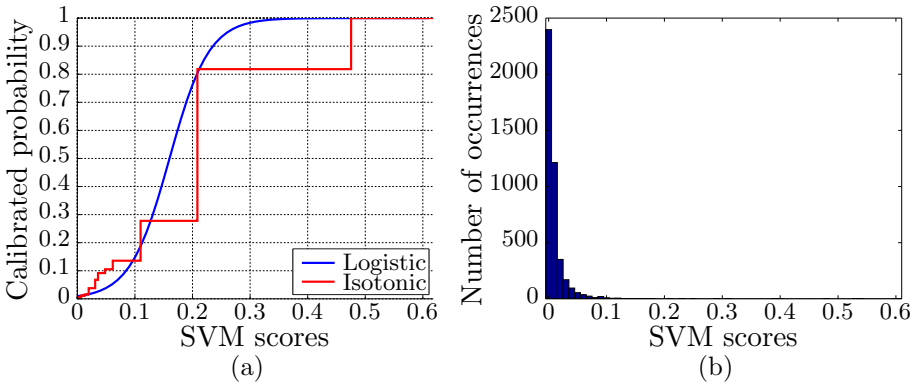


Figure 4.3: (a) Logistic and isotonic calibration of the scores from the 'HOG' pedestrian detector. (b) Histogram of the scores.

4.1.3 Clustering detectors

Detectors that return similar mass functions are likely to use similar information. Their combination should thus be handled more cautiously [98]. To define a measure between classifiers, a distance between mass functions has to be defined first. A survey of such distances can be found in [58]. A commonly used distance measure $d(m_1, m_2)$ between two mass functions m_1 and m_2 is defined as

$$\sqrt{\frac{1}{2}\sum_{A,B\subseteq\Omega\setminus\{\emptyset\}}\frac{|A\cap B|}{|A\cup B|}(m_1(A)-m_2(A))(m_1(B)-m_2(B))}. \tag{4.4}$$

For two simple mass functions, we get

$$0 \leq d\left(\{1\}^{\alpha_1},\{1\}^{\alpha_2}\right) = \frac{|\alpha_1-\alpha_2|}{\sqrt{2}} \leq \frac{1}{\sqrt{2}}. \tag{4.5}$$

The average distance for all detections is then used as a distance between the detectors $\mathcal{C}_{(k)}$ and $\mathcal{C}_{(\ell)}$:

$$\mathcal{D}(\mathcal{C}_{(k)},\mathcal{C}_{(\ell)}) = \frac{1}{n}\sum_{i=1}^{n} d\left(m_{(k),i}, m_{(\ell),i}\right), \tag{4.6}$$

where $m_{(k),i}$ and $m_{(\ell),i}$ refer to the mass functions associated to the i-th BB cluster provided by $\mathcal{C}_{(k)}$ and $\mathcal{C}_{(\ell)}$, respectively. The above definition actually assumes that, for every BB returned by $\mathcal{C}_{(k)}$ there is an associated one returned by $\mathcal{C}_{(\ell)}$. It is actually not the case. When one of the detector does not provide any BB, the distance is set to

$$\frac{1}{\sqrt{2}} \leq d\left(\{1\}^{\alpha_1},\emptyset\right) = d\left(\{1\}^{\alpha_1},\{0\}^{0}\right) = \sqrt{\frac{1+(1-\alpha_1)^2}{2}} \leq 1. \tag{4.7}$$

Using this pairwise distance, the detectors can be grouped through hierarchical clustering.

4.2 Experimental results

We conducted our experiments on the Caltech Pedestrian Detection Benchmark. The dataset consists in six training sets (set00-set05) that have been used to train detectors (see Table 4.1), and five testing sets (set06-set10). For our experiments we kept one of the testing sets (set06) as a validation set for calibration and the remaining four sets were used for testing. A five-fold cross-validation step was also conducted on the validation set to tune the different parameters of the combination system. As a performance measure, we used the *log-average miss rate* as proposed in [33]. It corresponds to the average of the miss rates computed at nine FPPI rates evenly spaced in log-space in the range 10^{-2} to 10^{0}.

4.2.1 Calibration and association of detectors

The first step of the combination is the calibration of the detectors. Figures 4.4 and 4.5 show the calibration of the pedestrian detectors, as well as the score distribution on the validation set. The white dots represent the calibration results using the binning approach with ten bins of equal size. The binning calibration is only used to have an empirical view of the shape of the posterior probability function. We can see that, for most of the methods, the posterior probability has a sigmoid shape and tends to increase with score. The distributions of the scores seem to have a Laplacian shape. It is interesting to note that the methods that combine an existing method with some new information, i.e., 'MultiResC+2Ped', 'DBN-Mut', 'MultiFtr+CSS', 'MultiFtr+Motion' and 'MF+Motion+2Ped', tend to concentrate the scores in one point. Apart from the 'VJ' and 'PoseInv' methods, we can see that the isotonic calibration returns a categorical mass function $\{1\}^0$ when the score is high enough. Having such certain information may be a sign of over-fitting. Logistic regression does not provide categorical mass functions but may still lead to very high confidence even for a detector with relatively low performance. In order to take into account the global performance of a detector, its log-average miss rate is used as a discounting factor.

Once the detectors have been calibrated, a BBs association strategy needs to be set before combining the BBs. Figure 4.6 shows the influence of the threshold used for the associations. The results were much better when doing the association greedily after score calibration. The best performance was obtained using the area of overlap a_{union} with a threshold of 0.45, although using a_{min} with a threshold of 0.8 gave very close results.

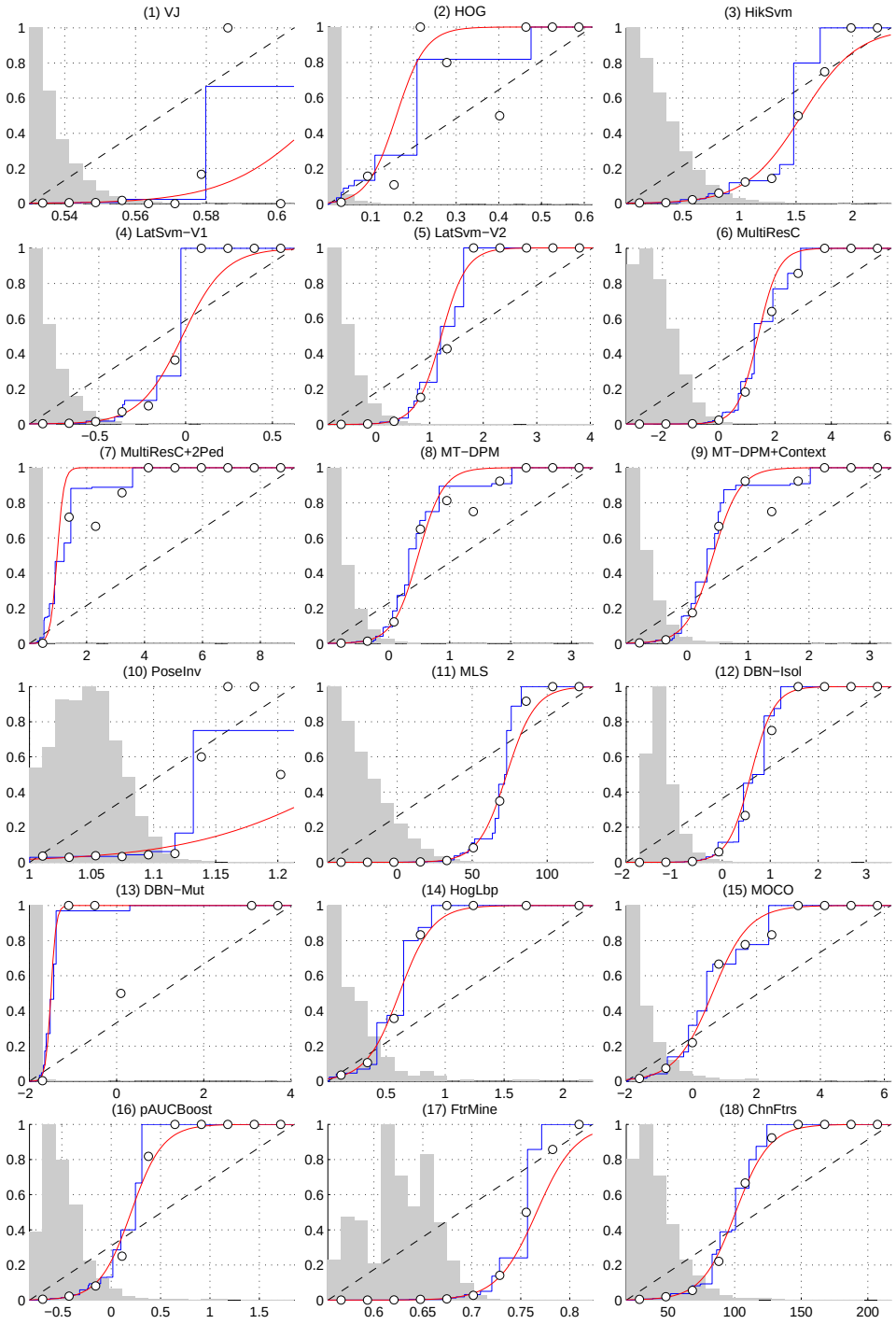


Figure 4.4: Calibration of pedestrian detectors. The x-axis corresponds to the raw scores returned by the detectors and the y-axis corresponds to the calibrated posterior probability. The white dots correspond to binning using ten equally spaced bins. Isotonic and logistic regression based calibration are represented, respectively, by the blue and red curves. The gray bars represent the empirical normalized distribution of the scores.

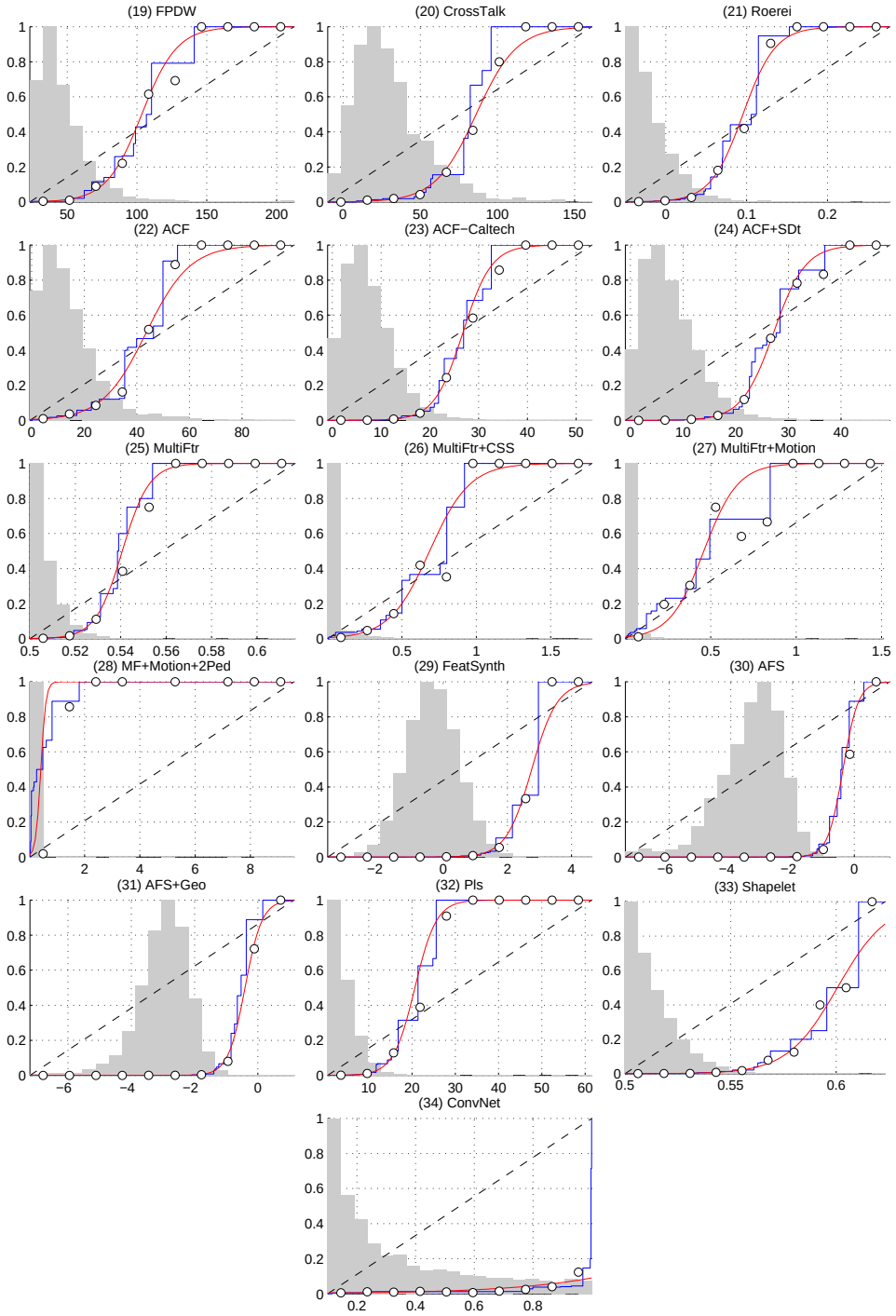


Figure 4.5: Calibration of pedestrian detectors. The x-axis corresponds to the raw scores returned by the detectors and the y-axis corresponds to the calibrated posterior probability. The white dots correspond to binning using ten equally spaced bins. Isotonic and logistic regression based calibration are represented, respectively, by the blue and red curves. The gray bars represent the empirical normalized distribution of the scores.

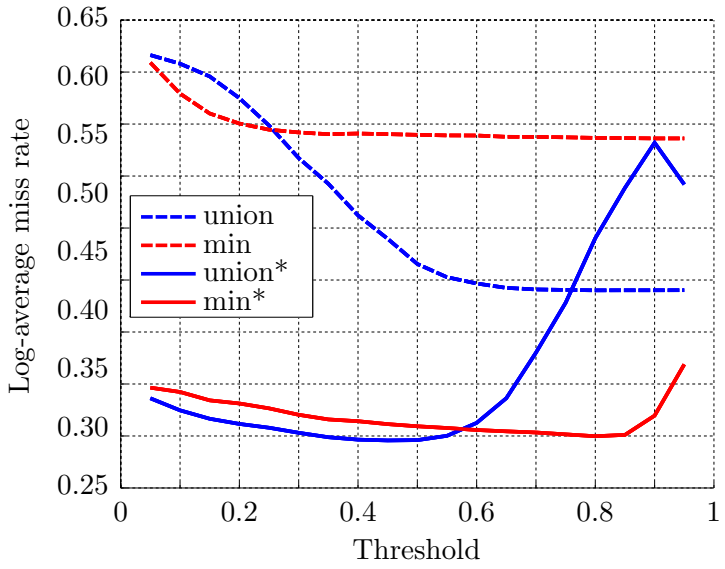


Figure 4.6: Log-average miss rate for different values of the overlapping threshold. The methods marked with a star correspond to greedy box association after a logistic calibration.

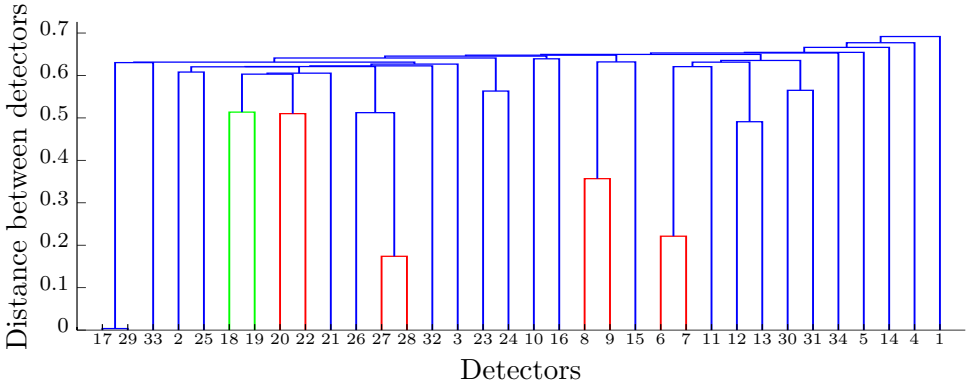


Figure 4.7: Detectors hierarchical clustering. The blue branches correspond to Dempster's rule, the red ones to the cautious rule and the green ones to a t-norm rule with parameter different from 0 and 1.

For the t-norm-based rule, the detectors were combined following the hierarchical clustering shown in Figure 4.7. For each pairwise combination the parameter of the t-norm was computed from the validation set. For most pairs of clusters, the best results were obtained using Dempster's rule ($p = 1$). The detectors #7 and #26 are, respectively, the combination of #6 and #27 with a 2-pedestrian detector while #9 uses a car detector with #8. For these three pairs, the cautious rule ($p = 0$) was optimal. The only case where the t-norm parameter was different from 0 and 1 was the combination between 'ChnFtrs' and 'FPDW'. The relatively high diversity of the evaluated detectors explains the limited gain from the t-norm rule compared to Dempster's rule.

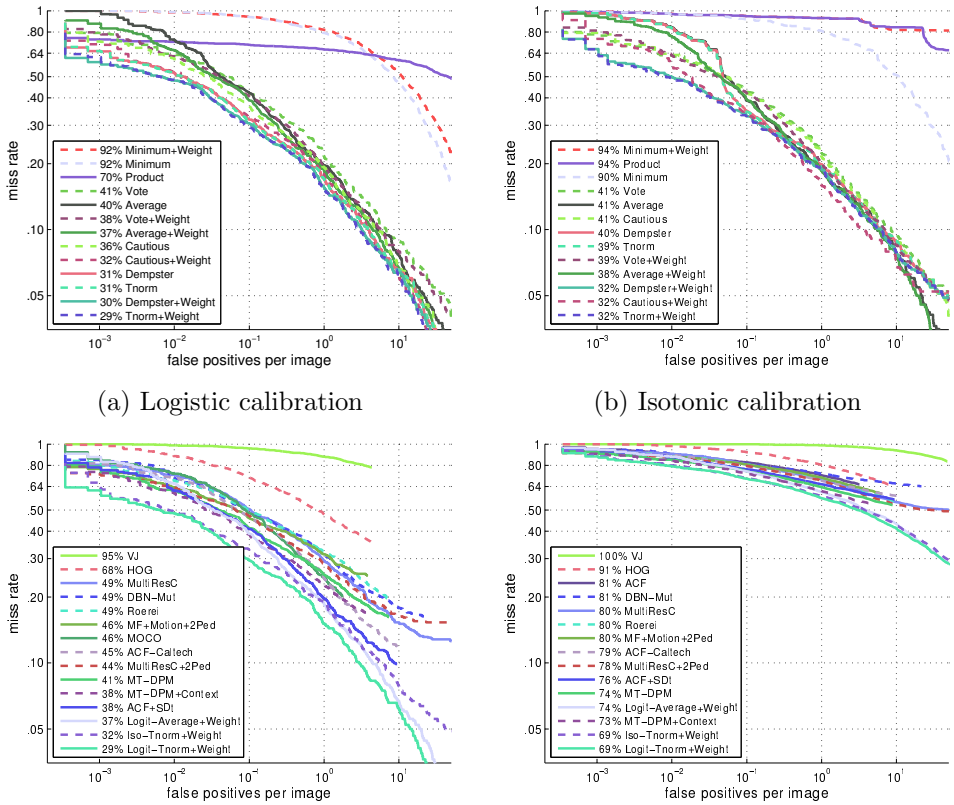
4.2.2 Combination performances

We compared the probabilistic combination rules with the evidential ones. Figure 4.8a and 4.8b show the results obtained on the "Reasonable" case scenario using a logistic and isotonic calibration, respectively. For the weighted version of the combinations, the average precision estimated on the validation set was used as weight, i.e., discounting factor. We can see that the product and minimum rules performed very poorly. The average rule performed better than the majority vote. The cautious rule, which is equivalent to the maximum rule, performed better than all the other probabilistic rules but worse than Dempster's rule and the t-norm based rule. Using an additional weight led to better results for all combination methods except the minimum combination rule. The logistic calibration always gave better results than the isotonic calibration.

Figures 4.8c and 4.8d compare the 12 best detectors, including 'VJ' and 'HOG', to the logistic and isotonic weighted t-norm and the logistic weighted average. In the "Reasonable" scenario, the logistic weighted t-norm led to an improvement of 9% in terms of log-average miss rate and 6% for the isotonic one. The weighted average only led to 1% improvement. In the "Overall" scenario, the logistic and isotonic t-norm have very similar results with a performance improvement of 4% while the weighted average performed worse than the 'MT-DPM+Context' alone.

Results for the other scenarios are detailed in Figures 4.9, 4.10 and 4.11. The logistic weighted t-norm combination always gave the best results. The logistic and isotonic regression based t-norm rules always gave better results than the best single detector. The logistic weighted average rule performed worse than the best single detector in certain scenarios: (4.10b), (4.10c), (4.11b) and (4.11c).

Figure 4.12 shows some typical combination results where the confidence of some true positives was increased after the combination. The results from the logistic weighted t-norm methods are displayed next to the results from the ACF+SDt and MT-DPM+Context methods, which are the two best



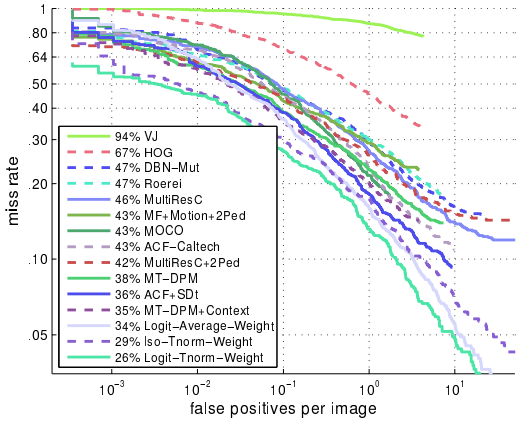
(a) Logistic calibration

(b) Isotonic calibration

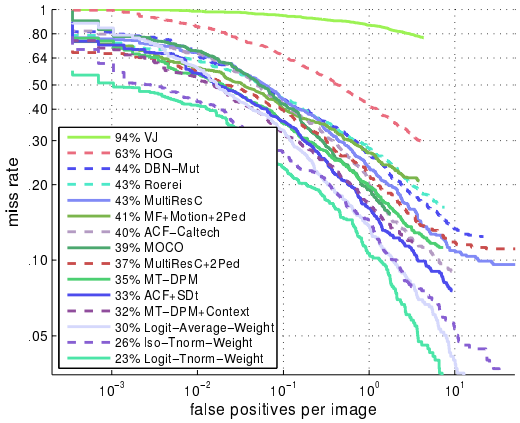
(c) Reasonable scenario

(d) Overall scenario

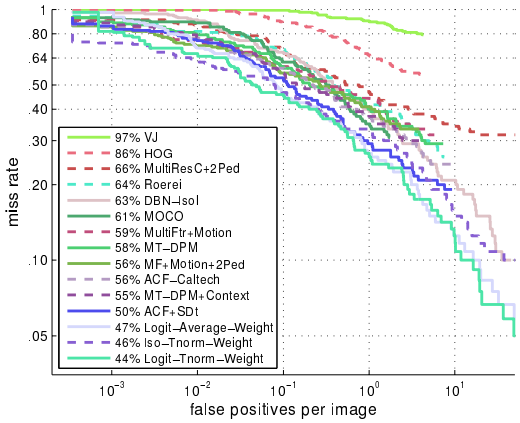
Figure 4.8: (a-b) Results of different combination strategies using a logistic and isotonic regression calibration methods on the "Reasonable" scenario. (c) Results on the "Reasonable" scenario. (d) Results on the "Overall" scenario.



(a) Overall aspect ratios

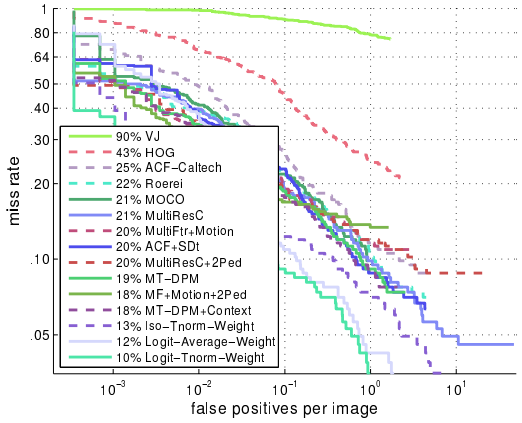


(b) Typical aspect ratios

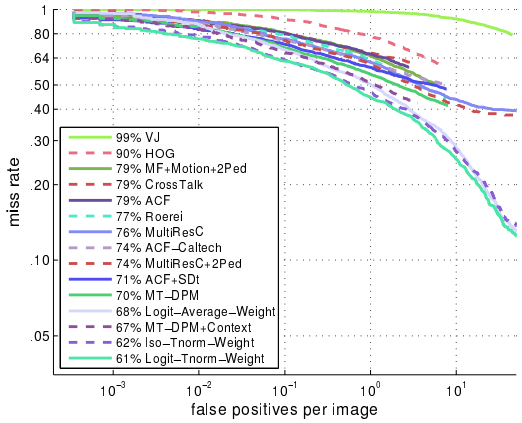


(c) Atypical aspect ratios

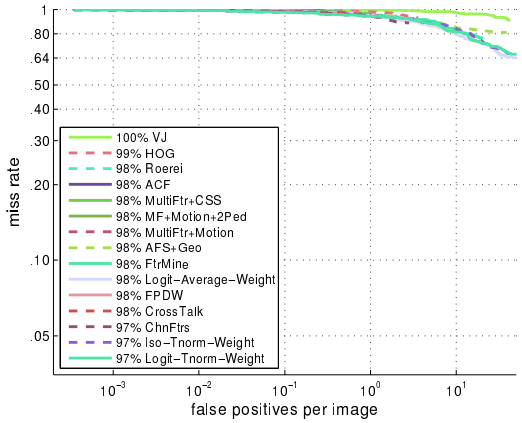
Figure 4.9: Results with respect to the aspect ratios.



(a) Near scale

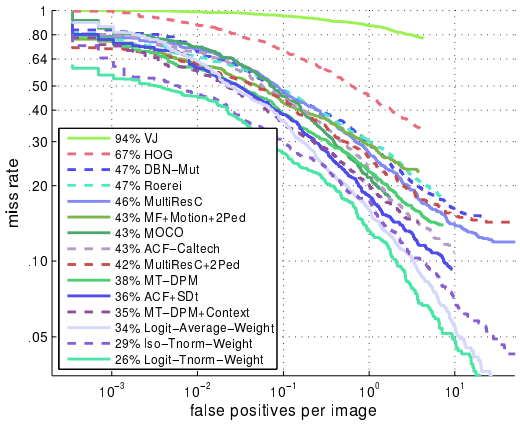


(b) Medium scale

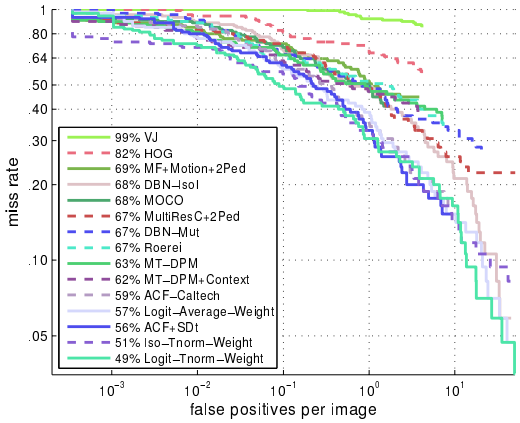


(c) Far scale

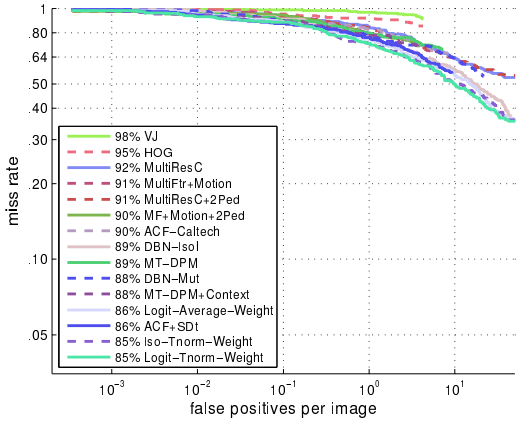
Figure 4.10: Results with respect to the scale.



(a) No occlusion



(b) Partial occlusion



(c) Heavy occlusion

Figure 4.11: Results with respect to the occlusion.

detectors. Typical examples where the confidence of some false positives detected by multiple detectors was increased are shown in Figure 4.13.

4.3 Conclusion

In this chapter, we proposed and evaluated an evidential framework for combining pedestrian detectors, noticing it could also be applied directly to detect other classes of objects. The use of belief functions and evidential combination rules yielded much better results than classical probabilistic approaches. One novelty of our approach relies on the use of an optimized t-norm rule, which can take into account the dependencies between detectors. This property can become critical if many new detectors are to be added. As optimized pairwise rules may provide only sub-optimal results, a global optimization will be investigated in future work. An important advantage of the proposed approach is that it allows us to easily include a new detector regardless of the features, training data and classifier it uses. Moreover, this modularity allows new detectors to rely on existing state-of-the-art ones. Therefore, one may focus future research on the development of detectors specially designed to detect *hard* examples without risking an overall recall loss.

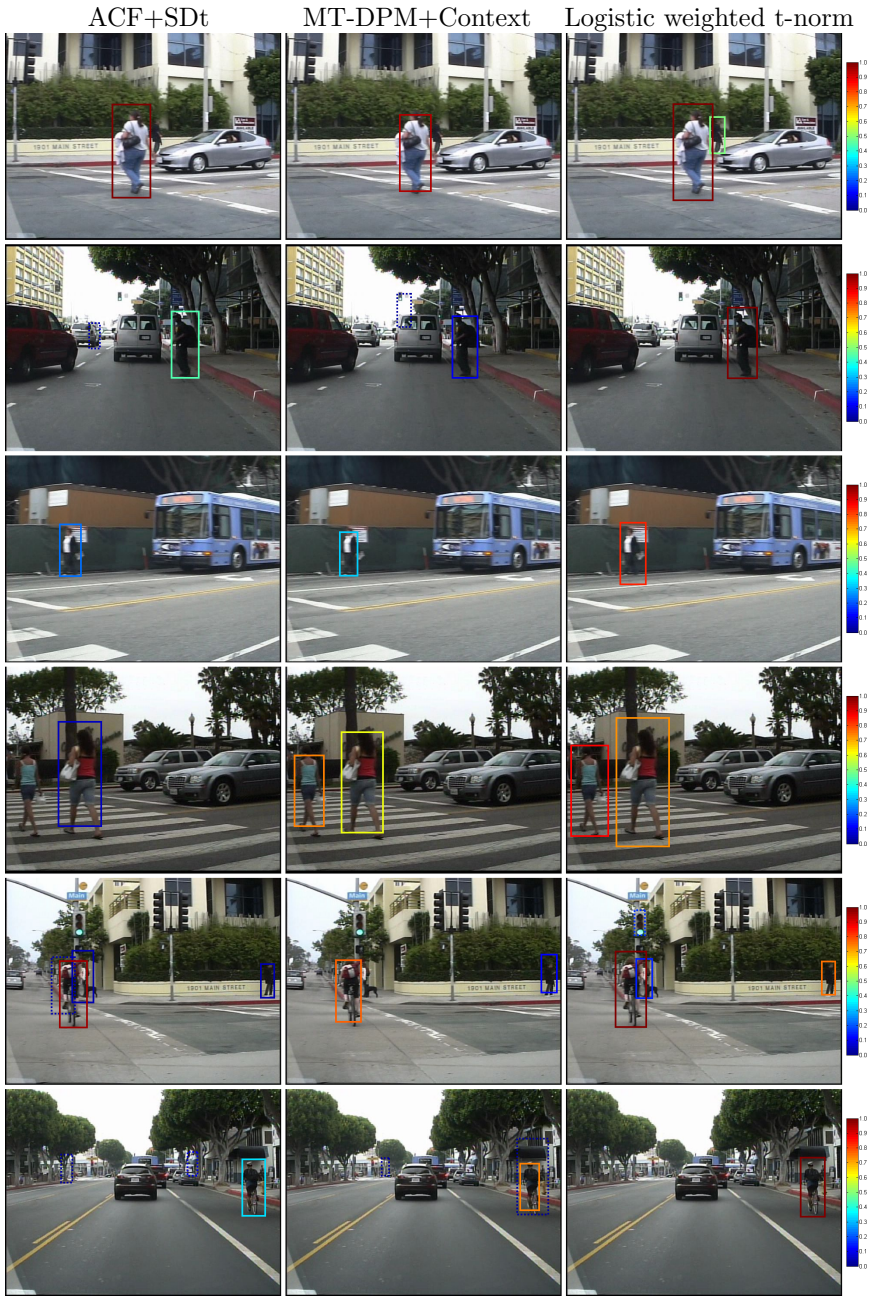


Figure 4.12: Detection results at 1 FPPI in the "Reasonable" scenario. Solid BBs correspond to true positives and doted BBs to false positives, the color represents the belief of containing a pedestrian after calibration. Those images show some combination results where the confidence of some true positives was increased. Some false positives were also discarded after combination.

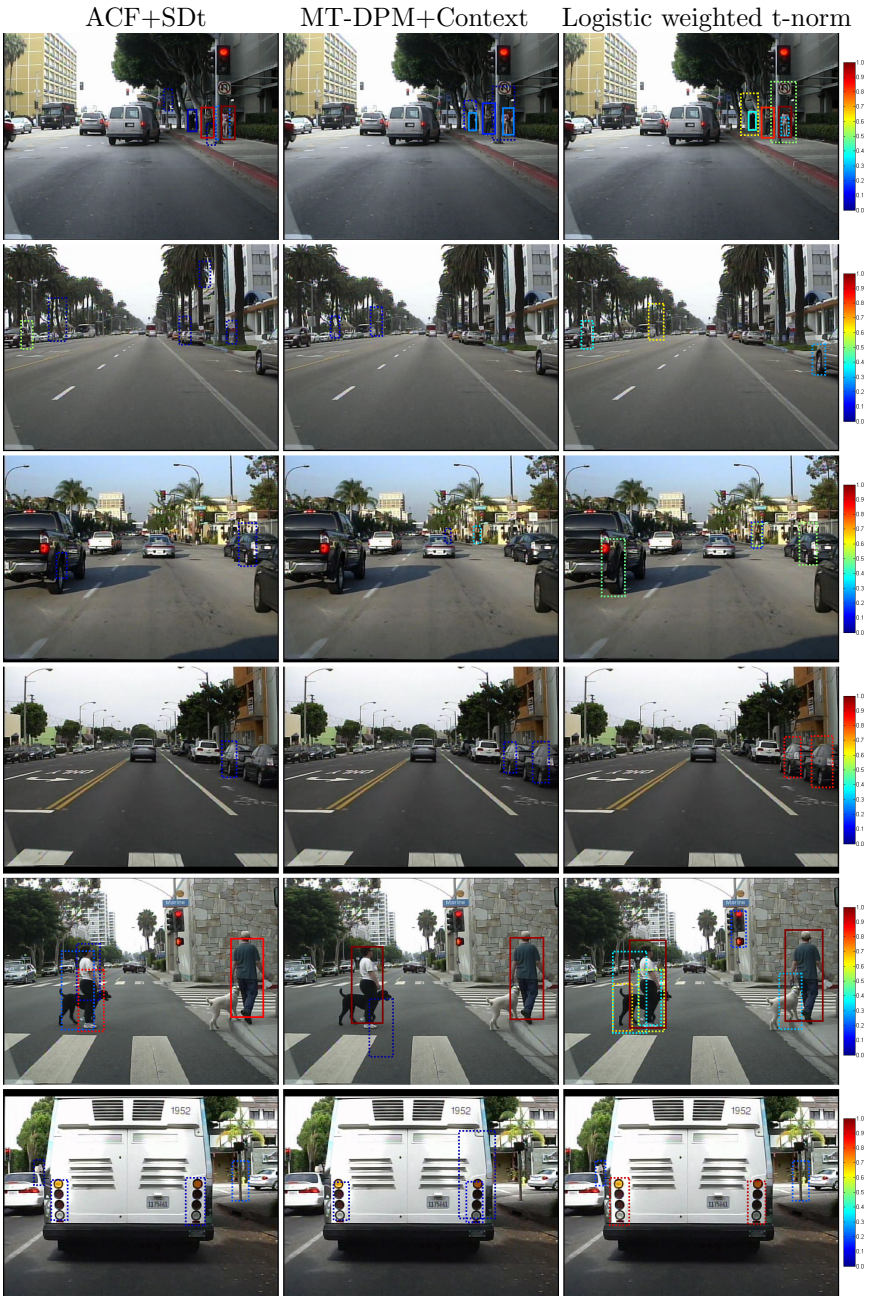


Figure 4.13: Detection results at 1 FPPI. Those images show some combination results where the confidence of some false positives returned by multiple detectors was increased.

Chapter 5

Local fusion in over-segmented images

The use of bounding boxes is limited to the detection of certain kinds of objects such as pedestrian or vehicles. Other classes of objects such as road, lane marking or sky, can only be detected at a segmentation level. Reasoning at the pixel level, however, may be too local and difficult. Semantic segmentation is often done by over-segmenting the image as in [57, 68]. The common task of all the modules, whatever the data representation they use (image, 3D points cloud or optical flow), then becomes to label each individual image segment. Figure 5.1 provides an overview of our fusion framework. Several sensors observe an urban scene, including a camera that produces an over-segmented image. Each sensor provides data to one or more modules, which are executed totally or partially in parallel to classify each image segment.

In this chapter, we first discuss the problem of over-segmentation for classification. We then describe the construction of several detection modules using probabilistic and evidential formulations. Finally, the whole multimodal system is evaluated on real urban driving scene data in Section 5.3.

5.1 Image over-segmentation

Many over-segmentation algorithms based on the mean-shift algorithm [17], graphs [48] or the k-means algorithm [1] can be found in the literature. The graph-based method proposed by Felzenszwalb and Huttenlocher [48] has been used in several works dealing with scene understanding [57, 68, 46]. Figure 5.2a shows the over-segmentation obtained by this method. We can see that the size and shape of the segments are very heterogeneous. It is relatively difficult to describe these kinds of segments with geometric notions such as height or depth. Other over-segmentation approaches [75, 1, 79] can provide a grid-like segmentation with a relatively uniform distribution in size

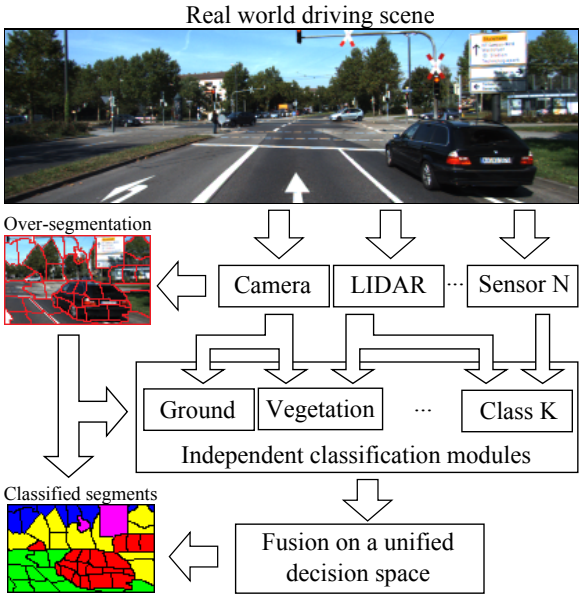


Figure 5.1: Overview of the fusion framework. N sensors, including a camera, observe the scene and provide data to K independent modules. The classification outputs are then fused in a unified decision space built from an over-segmented image.

and in shape. We chose to use the SLIC (Simple Linear Iterative Clustering) algorithm [1] for its simplicity and speed. We can see in Figure 5.2b that the over-segmentation obtained using the SLIC algorithm is more regular.

The formulation of the SLIC algorithm is very simple. Each pixel is described by a vector $\mathbf{x}_i = [l_i\ a_i\ b_i\ u_i\ v_i]$, $i = 1, \ldots, n$, where the components l_i, a_i and b_i are the color in the CIELAB color space, u_i, v_i are the pixel coordinates and n is the number of pixels. The k-means algorithm is then used to group pixels into k clusters, which correspond to the segments. The distance between two pixels $\mathbf{x}_i$ and $\mathbf{x}_j$ combines two distances: a color distance d_c and a spatial distance d_s defined as

$$d_c(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(l_j - l_i)^2 + (a_j - a_i)^2 + (b_j - b_i)^2}, \tag{5.1a}$$

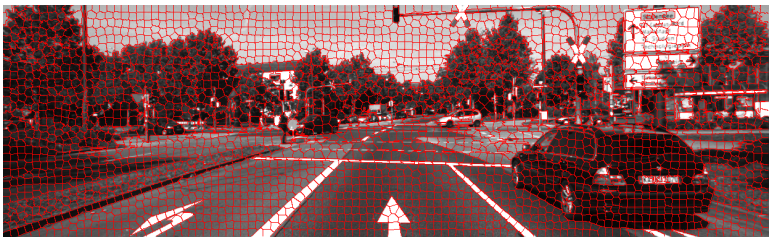
$$d_s(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}. \tag{5.1b}$$

These two distance measures are then normalized and combined, leading to

$$D(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\left(\frac{d_c(\mathbf{x}_i, \mathbf{x}_j)}{n_c}\right)^2 + \left(\frac{d_s(\mathbf{x}_i, \mathbf{x}_j)}{n_s}\right)^2}. \tag{5.2}$$

The spatial normalization constant n_s corresponds to the maximum spatial distance expected within a given cluster, which can be approximated by

(a) Graph-based approach proposed by Felzenszwalb and Huttenlocher [48]



(b) SLIC algorithm [1]

Figure 5.2: Over-segmentation results.

$n_s = \sqrt{n/k}$. One can either set the desired number of clusters k or the desired average size of the segments n_s. The color normalization constant n_c is more difficult to estimate as the distance d_c can vary significantly from cluster to cluster and image to image. Achanta *et al.* [1] proposed to fix n_c to a constant in the range $[1, 40]$. The parameter n_c weights the relative importance between color similarity and spatial proximity. When n_c is large, spatial proximity is more important and yields more compact superpixels, i.e., they have a lower area to perimeter ratio. When n_c is small, the superpixels adhere more tightly to image boundaries, but may have less regular size and shape. In our experiments, we choose $n_s = 12$ and $n_c = 20$ which realize a good trade-off between of size and shape.

5.2 Belief functions for semantic segmentation

We applied a belief function-based fusion framework to a multi-modal system including a stereo camera and a LiDAR sensor, which are supposed to be calibrated [51]. Several modules independently process the outputs of these sensors to classify each segment of the image in Figure 5.2b. Some simple classification rules are first applied directly using pixel coordinates. The 3D information from the stereo images and the LiDAR are then used to detect the ground. Next, two monocular-based approaches allow us to infer the scene layout and further extend it by including a vegetation class. Finally, a temporal propagation module is used to link two consecutive images. The

inputs of the different modules described below are shown in Figure 5.3.

5.2.1 Classification from pixel location

Some very simple rules can be directly inferred from pixel coordinates. For example, we are certain that the "lower" part of the image cannot be the sky and the "upper" part cannot be the ground. By assuming a maximum pitch angle of $\pm 5°$, upper ($V_{\max}$) and lower ($V_{\min}$) bounds of the horizon line can be computed as illustrated by the blue and green lines in Figure 5.4a. This assumption may not hold in certain complex situations such as uphills or downhills, for which a robust horizon line estimator would be needed. A segment in the image can be described by its minimum and maximum vertical coordinate $(\underline{v}, \overline{v})$. Two distinct mass functions can then be constructed. The first one is defined over the frame of discernment $\Omega_{\mathrm{S}} = \{\text{sky}, \cancel{\text{sky}}\}$ as

$$m^{\Omega_{\mathrm{S}}}_{\overline{v}}(\{\cancel{\text{sky}}\}) = \begin{cases} 1 & \text{if } \overline{v} \leq V_{\min}, \\ 0 & \text{otherwise}, \end{cases} \tag{5.3a}$$

$$m^{\Omega_{\mathrm{S}}}_{\overline{v}}(\{\text{sky}\}) = 0, \tag{5.3b}$$

$$m^{\Omega_{\mathrm{S}}}_{\overline{v}}(\Omega_{\mathrm{S}}) = 1 - m^{\Omega_{\mathrm{S}}}_{\overline{v}}(\{\cancel{\text{sky}}\}). \tag{5.3c}$$

This mass function states that, if the maximum vertical coordinate is lower that the lower bound $V_{\min}$, then the segment cannot be the sky. Otherwise, we do not know if the segment corresponds to the sky or not, which is represented by the vacuous mass function $m^{\Omega_{\mathrm{S}}}_{\overline{v}}(\Omega_{\mathrm{S}}) = 1$. Similarly, a second mass function is defined over $\Omega_{\mathrm{G}} = \{\text{ground}, \cancel{\text{ground}}\}$ as

$$m^{\Omega_{\mathrm{G}}}_{\underline{v}}(\{\cancel{\text{ground}}\}) = \begin{cases} 1 & \text{if } \underline{v} \geq V_{\max}, \\ 0 & \text{otherwise}, \end{cases} \tag{5.4a}$$

$$m^{\Omega_{\mathrm{G}}}_{\underline{v}}(\{\text{ground}\}) = 0, \tag{5.4b}$$

$$m^{\Omega_{\mathrm{G}}}_{\underline{v}}(\Omega_{\mathrm{G}}) = 1 - m^{\Omega_{\mathrm{G}}}_{\underline{v}}(\{\cancel{\text{ground}}\}). \tag{5.4c}$$

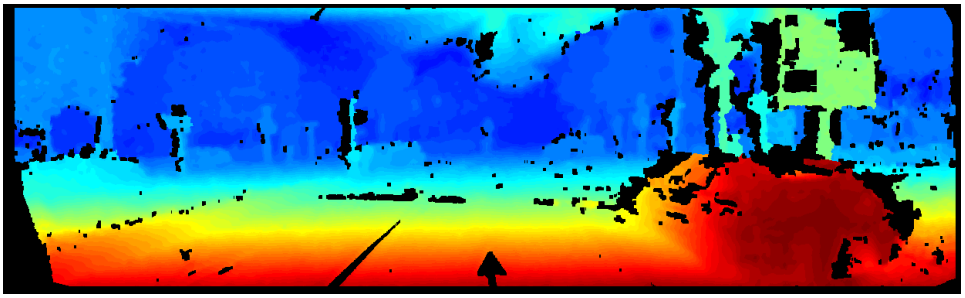
These two mass functions can be combined by Dempster's rule on a common refinement $\Lambda = \{\text{ground, vertical, sky}\}$, yielding

$$m^{\Lambda}_{\underline{v},\overline{v}}\left(\overline{\{\text{sky}\}}\right) = \begin{cases} 1 & \text{if } \overline{v} \leq V_{\min}, \\ 0 & \text{otherwise}, \end{cases} \tag{5.5a}$$

$$m^{\Lambda}_{\underline{v},\overline{v}}\left(\overline{\{\text{ground}\}}\right) = \begin{cases} 1 & \text{if } \underline{v} \geq V_{\max}, \\ 0 & \text{otherwise}, \end{cases} \tag{5.5b}$$

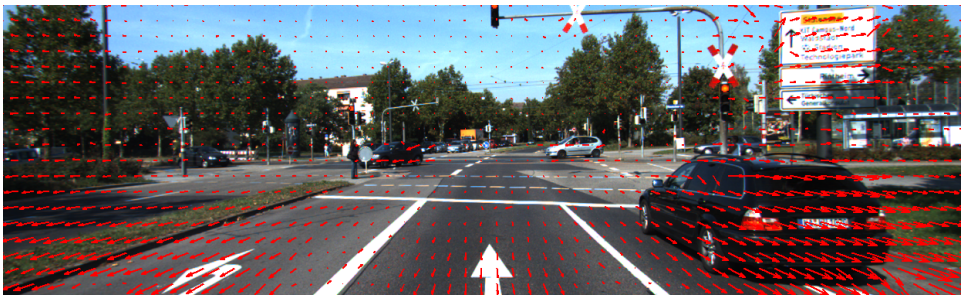
$$m^{\Lambda}_{\underline{v},\overline{v}}(\Lambda) = 1 - m^{\Lambda}_{\underline{v},\overline{v}}\left(\overline{\{\text{sky}\}}\right) - m^{\Lambda}_{\underline{v},\overline{v}}\left(\overline{\{\text{ground}\}}\right), \tag{5.5c}$$

where $\overline{\{\text{sky}\}} = \{\text{ground, vertical}\}$ and $\overline{\{\text{ground}\}} = \{\text{vertical, sky}\}$. Figure 5.4b illustrates the combined mass functions.



(a) Disparity map

(b) Laser points



(c) Optical flow

Figure 5.3: Inputs to the multi-sensor system. (a) The disparity map is computed from the ELAS algorithm [52]. (b) A single laser layer is extracted from a Velodyne LiDAR. (c) The optical flow is computed using the TV-L1 formulation as implemented in OpenCV [135].

Following the same reasoning, a probabilistic approach would lead to the following probabilities:

$$P_{\overline{v}}^{\Omega_S}(\text{~~sky~~}) = \begin{cases} 1 & \text{if } \overline{v} \leq V_{\min}, \\ 1/2 & \text{otherwise,} \end{cases} \tag{5.6a}$$

$$P_{\overline{v}}^{\Omega_S}(\text{sky}) = \begin{cases} 0 & \text{if } \overline{v} \leq V_{\min}, \\ 1/2 & \text{otherwise,} \end{cases} \tag{5.6b}$$

and

$$P_{\underline{v}}^{\Omega_G}(\text{~~ground~~}) = \begin{cases} 1 & \text{if } \underline{v} \geq V_{\max}, \\ 1/2 & \text{otherwise,} \end{cases} \tag{5.7a}$$

$$P_{\underline{v}}^{\Omega_G}(\text{ground}) = \begin{cases} 0 & \text{if } \underline{v} \geq V_{\max}, \\ 1/2 & \text{otherwise.} \end{cases} \tag{5.7b}$$

By using the product rule $\boxtimes$ over $\Lambda = \{\text{ground, vertical, sky}\}$, with a uniform prior distribution, the combined probability is defined as follows:

$P_{\underline{v},\overline{v}}^{\Lambda}$	$\overline{v} \leq V_{\min}$	$\underline{v} \geq V_{\max}$	otherwise
ground	2/3	0	2/5
vertical	1/3	1/3	1/5
sky	0	2/3	2/5

(5.8)

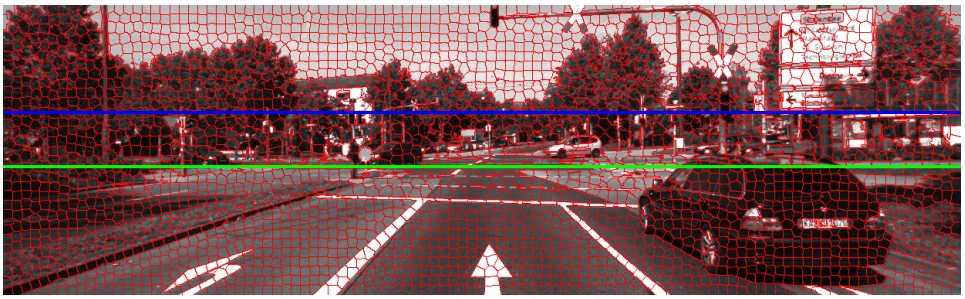
The resulting probability $P_{\underline{v},\overline{v}}^{\Lambda} = P_{\overline{v}}^{\Omega_S} \boxtimes P_{\underline{v}}^{\Omega_G}$ does not encode the same information as $P_{\overline{v}}^{\Omega_S}$ and $P_{\underline{v}}^{\Omega_G}$. In particular, complete ignorance, which is represented differently by $P_{\overline{v}}^{\Omega_S} = U_{\overline{v}}^{\Omega_S}$ and $P_{\underline{v}}^{\Omega_G} = U_{\underline{v}}^{\Omega_G}$, is not encoded by a uniform distribution in Λ. By reasoning directly on Λ, the principle of indifference would actually lead to the following probability:

$Q_{\underline{v},\overline{v}}^{\Lambda}$	$\overline{v} \leq V_{\min}$	$\underline{v} \geq V_{\max}$	otherwise
ground	1/2	0	1/3
vertical	1/2	1/2	1/3
sky	0	1/2	1/3

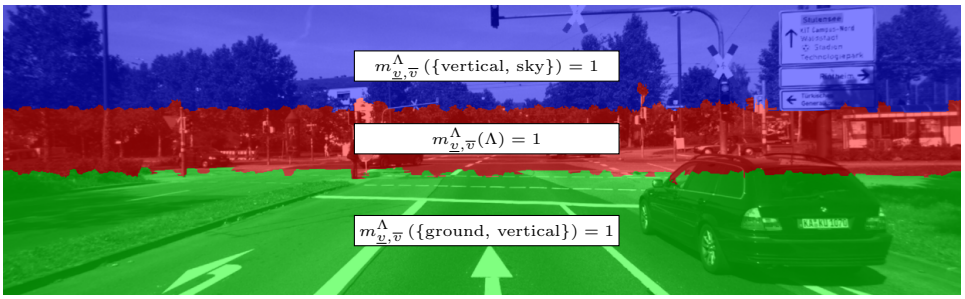
(5.9)

The probability distributions $Q_{\underline{v},\overline{v}}^{\Lambda}$ ans $P_{\underline{v},\overline{v}}^{\Lambda}$ are illustrated in Figure 5.4c and Figure 5.4d, respectively. The probability $Q_{\underline{v},\overline{v}}^{\Lambda}$ seems much more reasonable than $P_{\underline{v},\overline{v}}^{\Lambda}$. In particular, the red zone, where nothing can actually be inferred, is well represented by a uniform distribution with $Q_{\underline{v},\overline{v}}^{\Lambda}$ but not with $P_{\underline{v},\overline{v}}^{\Lambda}$.

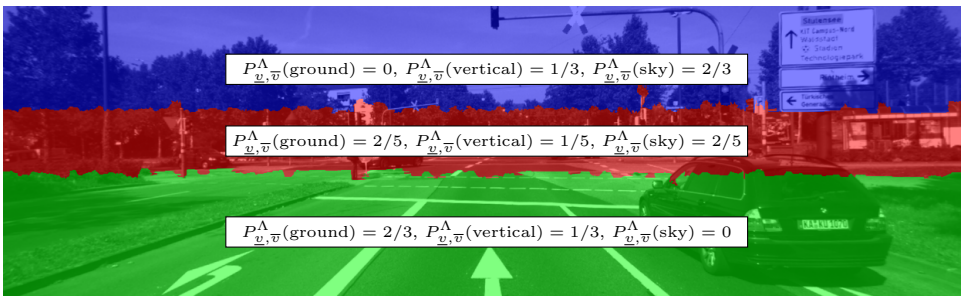
However, if a new class, such as vegetation, has to be added, neither $P_{\underline{v},\overline{v}}^{\Lambda}$ or $Q_{\underline{v},\overline{v}}^{\Lambda}$ would actually be correct. This example clearly shows that imprecise information cannot be properly represented by probabilities. Moreover, the information on the upper and lower part of the image remains certain when using belief functions while it is encoded as uncertain with probabilities.



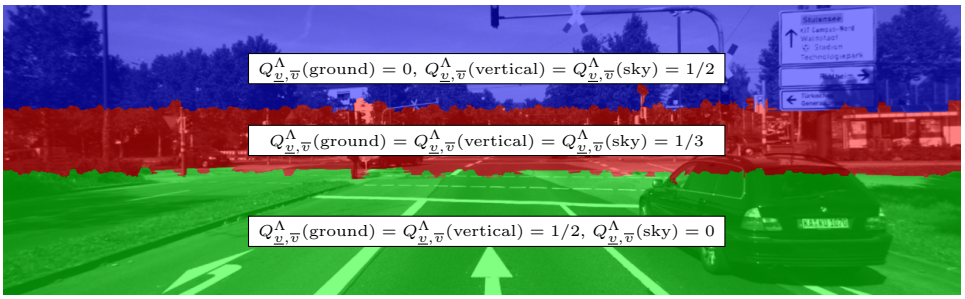
(a) Lower and upper bounds



(b) Mass function



(c) Combined probability



(d) Uniform probability

Figure 5.4: Classification from pixel coordinates. (a) Lower and upper bounds of the horizon line: $V_{\min}$ (green), $V_{\max}$ (blue). (b) Mass function $m^{\Lambda}_{\underline{v},\overline{v}} = m^{\Omega_{\mathrm{G}}}_{\underline{v}} \oplus m^{\Omega_{\mathrm{S}}}_{\overline{v}}$. (c) Probability $P^{\Lambda}_{\underline{v},\overline{v}} = P^{\Omega_{\mathrm{S}}}_{\overline{v}} \boxtimes P^{\Omega_{\mathrm{G}}}_{\underline{v}}$. (d) Probability $Q^{\Lambda}_{\underline{v},\overline{v}}$.

5.2.2 Stereo-based classification

3D information is very useful for scene understanding. A disparity map encoding the depth of each pixel (Figure 5.3a) can be estimated using a stereo camera. We used the ELAS algorithm [52] which is designed for fast high-resolution image processing.

A Euclidean 3D point cloud is first generated from the disparity map and used to estimate the ground surface. We programmed a RANSAC robust plane estimator to detect the ground plane. The assumption of a planar ground turns out to be reasonable in practice. For greater robustness, the use of more complex models such as B-splines [124] could also be considered.

The estimated ground plane Π is used to build a ground detector. Each segment is seen as a set of 3D points: $\mathbf{x} = \{p_1, \ldots, p_k, p^*_{k+1}, \ldots, p^*_n\}$, where the points denoted by p^*_i are those for which no disparity has been estimated. A segment is classified as ground or non-ground depending on its distance to the ground plane. The distance d between the observation $\mathbf{x}$ and the plane Π is defined as the median distance of the valid points p_i to Π, while forgetting the invalid ones p^*_j:

$$d(\mathbf{x}, \Pi) \quad = \quad \underset{i=1,\ldots,k}{\text{med}}\, \delta(p_i, \Pi), \tag{5.10}$$

where $\delta(p_i, \Pi)$ is the Euclidean distance from p_i to Π. Figure 5.5a illustrates the distance to the ground obtained for each pixel. Figure 5.5b shows the median distance computed for each segment.

To get a probability measure from the distance d, logistic regression is used by assuming that

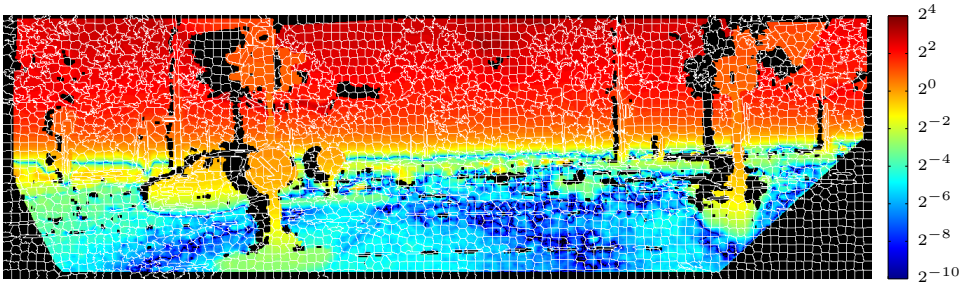
$$P_d^{\Omega_G}(\text{ground}) \quad = \quad \frac{1}{1 + \exp(ad + b)}, \tag{5.11}$$

where the sigmoid parameters $a, b \in \mathbb{R}$ can be optimized given some training data. As only k out of n points are visible, the reliability of the observation is modeled by $P_R(r = 1) = k/n$. When no disparity estimates are available, *i.e.* $P_R(r = 1) = 0$, we get the uniform distribution $P_{d,0}^{\Omega_G}(\text{ground}) = P_{d,0}^{\Omega_G}(\cancel{\text{ground}}) = 1/2$. Figure 5.5c shows the probability obtained from the distance to the ground plane.

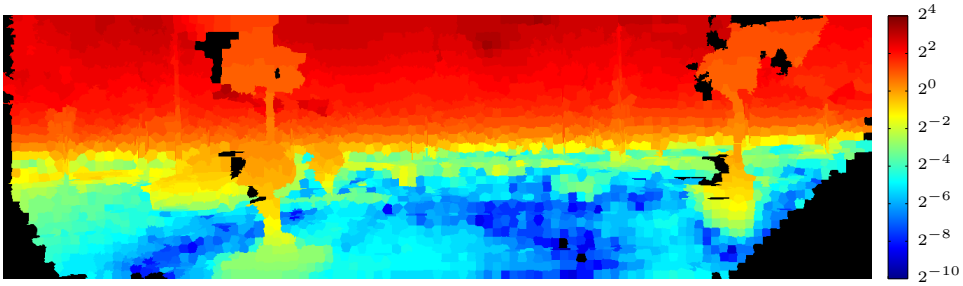
With belief functions, a more cautious model can be used. Instead of using the median distance, two distances $\underline{d}$ and $\overline{d}$ were considered. They correspond, respectively, to the minimum and maximum distance from the segment to the ground plane and are defined as

$$\underline{d} = \min_{i=1,\ldots,k} \delta(p_i, \Pi) \quad \text{and} \quad \overline{d} = \max_{i=1,\ldots,k} \delta(p_i, \Pi). \tag{5.12}$$

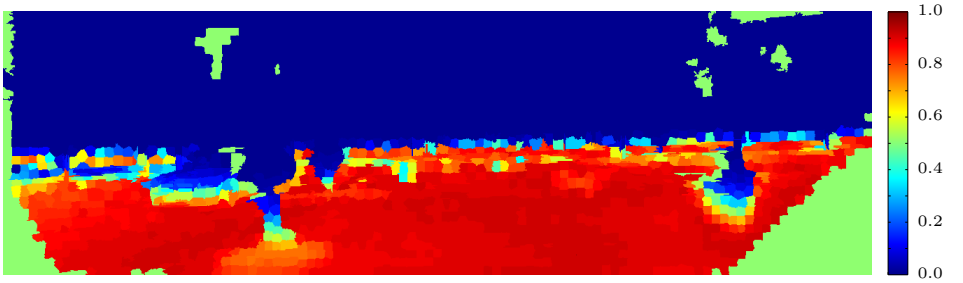
The minimum distance $\underline{d}$ was used to build a mass function $m_{\underline{d}}^{\Omega_G}$ that only supports the non-ground class. If the minimum distance is large, then we



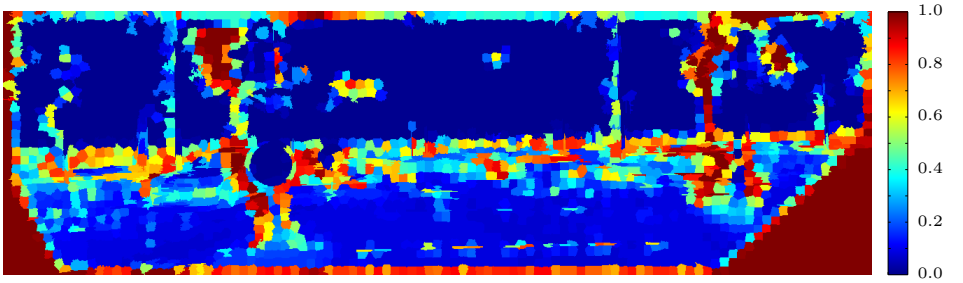
(a) Distance to ground plane (m)



(b) Median distance to ground plane (m)



(c) Probability of the ground class



(d) Amount of ignorance

Figure 5.5: Stereo-based ground classification. (a) Distance to the ground plane for each pixel. (b) Median distance of segments to the ground plane. (c) Probability of the ground class $P_d^{\Omega_G}$. (d) Amount of ignorance $m_{\underline{d},\overline{d}}^{\Omega_G}(\Omega_G)$.

are confident about the non-ground class. However, if the minimum distance is small, nothing can actually be said. The mass function $m_{\underline{d}}^{\Omega_G}$ was defined in a way similar to (5.11):

$$m_{\underline{d}}^{\Omega_G}(\{\cancel{\text{ground}}\}) = \frac{1}{1+\exp(\underline{a}\underline{d}+\underline{b})}, \tag{5.13a}$$

$$m_{\underline{d}}^{\Omega_G}(\{\text{ground}\}) = 0, \tag{5.13b}$$

$$m_{\underline{d}}^{\Omega_G}(\Omega_G) = 1 - m_{\underline{d}}^{\Omega_G}(\{\cancel{\text{ground}}\}), \tag{5.13c}$$

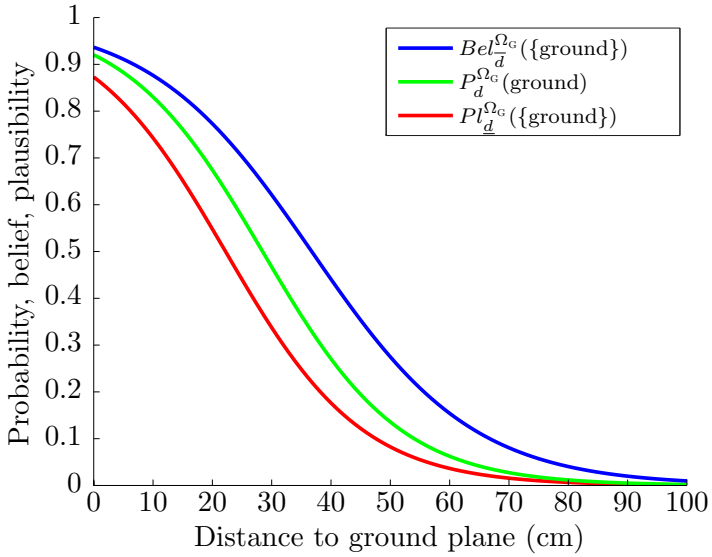
where the parameters $\underline{a}$ and $\underline{b}$ were determined using some training data. In a similar way, the maximum distance $\overline{d}$ was used to build a mass function $m_{\overline{d}}^{\Omega_G}$ that only supports the ground class. A combined mass function $m_{\underline{d},\overline{d}}^{\Omega_G} = m_{\underline{d}}^{\Omega_G} \oplus m_{\overline{d}}^{\Omega_G}$ was then obtained by Dempster's rule. Finally, the mass function was discounted by a factor $\alpha = 1 - k/n$, which results in the vacuous mass function when no disparity was estimated.

Figure 5.6a shows the measures $P_d^{\Omega_G}$, $m_{\underline{d}}^{\Omega_G}$ and $m_{\overline{d}}^{\Omega_G}$ obtained from logistic regression. Figure 5.6b illustrates the degree of ignorance $m_{\underline{d},\overline{d}}^{\Omega_G}(\Omega_G)$ for different values of $\underline{d}$ and $\overline{d}$. We can see that when $\underline{d}$ is small and $\overline{d}$ is large (top right corner), the amount of ignorance is high. In contrast, when $\underline{d}$ is large (bottom right) or $\overline{d}$ is small (top left), the information is more certain. Figure 5.5d displays the degree of ignorance in a typical case.

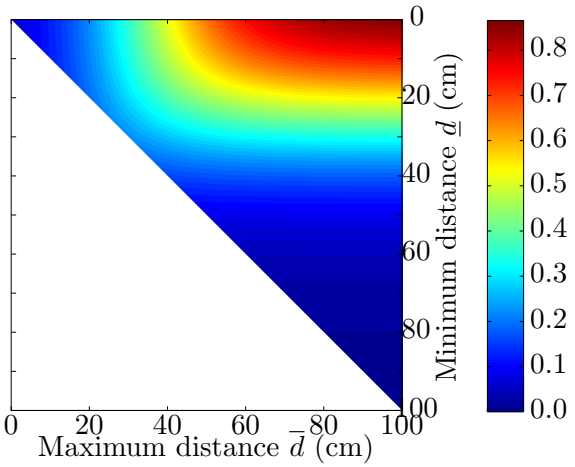
5.2.3 LiDAR-based classification

A LiDAR sensor provides a set of 3D points that are the impacts of laser beams (Figure 5.3b). As with the stereo camera, a segment S hit by some laser beams is perceived as a set of k 3D points. By using the ground plane estimated from the disparity map, the same form of mass function as in the stereo case was used for S. Additionally, the space between the projections on the ground plane of the laser impacts and the LiDAR's origin is considered to be obstacle free.

The data from the LiDAR sensor are illustrated in Figure 5.7a. The red dots represent the impacts returned by the LiDAR. The segments hit by these impacts are modeled and classified in the same way as in the stereo case. The green dots correspond to the projections of the impacts on the ground plane estimated by the stereo module. The green lines represent the laser rays from the green dots to the LiDAR's origin. The segments crossed by at least one green line are assimilated to the "ground" class. A categorical mass function $m_L^{\Omega_G}(\{\text{ground}\}) = 1$ is assigned to these segments. In the probabilistic case, the probability $P_L^{\Omega_G}(\text{ground}) = 1$ is used. Furthermore, a discounting factor or reliability measure $P_R(r=0) = \alpha = k/n$ is considered for the segments hit or crossed by at least one laser beam. The quantity n is defined as the maximum number of beams that could have hit or crossed the



(a)



(b)

Figure 5.6: (a) Probability, belief and plausibility of the ground class with respect to the distance to the ground plane. The belief and plausibility are defined as $Bel_{\overline{d}}^{\Omega_G}(\{\text{ground}\}) = m_{\overline{d}}^{\Omega_G}(\{\text{ground}\})$ and $Pl_{\underline{d}}^{\Omega_G}(\{\text{ground}\}) = 1 - m_{\underline{d}}^{\Omega_G}(\{\cancel{\text{ground}}\})$. (b) Amount of ignorance $m_{\underline{d},\overline{d}}^{\Omega_G}(\Omega_G)$ given the minimum and maximum distances of a segment to the ground plane.

(a) Impacts and beams from the LiDAR sensor

(b) Mass functions from the LiDAR module

Figure 5.7: (a) Red dots represents the impacts returned by the LiDAR. Green dots correspond to the projections of the impacts on the ground plane. The green lines represent the laser rays from the green dots to the LiDAR's origin. (b) The red, green and blue (RGB) colors represent the mass assigned to {~~ground~~}, {ground} and Ω_G respectively.

segment. Finally, the segments between the red and green dots, represented by the blue lines, are ambiguous and are modeled by a vacuous mass function or uniform probability distribution. It is also the case for all the segments that are neither hit nor crossed by some laser beams. The result obtained from the LiDAR module is displayed in Figure 5.7b.

5.2.4 Surface layout from monocular images

Geometric structures in the scene can also be estimated directly from a single image. We used the method proposed by Hoiem et al. [57], whose code and pre-trained models are publicly available[1]. They used a set of multiple features including location, color, texture and perspective cues such as line intersections or vanishing points. Boosted decision trees were used to learn a multi-class classifier. The logistic regression version of Adaboost was used in order to get well-calibrated probabilities as output.

Hoiem et al. [57] considered three classes: "support", "vertical" and

[1] http://www.cs.uiuc.edu/~dhoiem

"sky". In our case, the "support" class corresponds to the ground. Hoiem et al. further decomposed the "vertical" class into five subclasses: "left", "center", "right", "porous" and "solid". These five subclasses are, however, of limited meaning in our case, so they were not considered. Additionally, the over-segmentation algorithm from Felzenszwald and Huttenlocher [48], which was originally used, was replaced by the SLIC over-segmentation [1].

As the output is a probability distribution, it can be directly used for probabilistic fusion. It can also be considered as a Bayesian mass function in an evidential context. However, the use of a Bayesian mass function would constrain the results of the combination to be Bayesian. To avoid such situations, the inverse pignistic transformation is used (see Section 2.1.3). Finally, the accuracy of the algorithm of Hoiem *et al.* [57] on our training data is used as a discounting factor. Figure 5.8a displays results obtained with this module.

5.2.5 Texture-based classification

The textural appearance of a segment is an important cue about its class. We used the Walsh-Hadamard transform to encode the texture, as proposed by Wojek and Schiele [127]. For each segment, the Walsh-Hadamard coefficients were computed over 8×8 and 16×16 pixel patches centered at the centroid of the segment. The three color channels were processed individually in the L*a*b* color space resulting in a feature vector of dimension 960.

This texture information was then used to build a vegetation detection module. A linear binary classifier was trained from a L1-regularized logistic regression as implemented in the Liblinear library [45]. This library was designed to efficiently learn linear classifiers from very large datasets. The probabilistic classifier output was handled as described in Sec. 5.2.4. No discounting or reliability estimation was used for this module as the classifier was directly trained on the KITTI dataset and can be assumed to be well-calibrated. Figure 5.8b displays results obtained with this vegetation detector.

5.2.6 Temporal propagation

Given two consecutive images at times t and $t-1$, the optical flow (Figure 5.3c) can be used to propagate the information. We used the OpenCV implementation of the TV-L1 formulation as proposed by Zach et al. [135]. To each segment S_t at time t was associated a previous segment S_{t-1} at time $t-1$, defined as the segment pointed by the mean flow of the pixels in S_t. The mass function or probability associated to S_{t-1} was simply propagated to S_t. A discounting factor corresponding to the ratio of pixels in S_t whose flow actually points to S_{t-1} was then used as reliability measure. This temporal propagation can be used with any frame of discernment. The prop-

agation of the results from the stereo-based ground detector is illustrated in Figure 5.8c and Figure 5.8d.

5.3 Experimental results

The KITTI dataset [51] was used to validate our approach, considering the stereo color camera and Velodyne 64-beam LiDAR. However, only one layer of the Velodyne LiDAR was used in order to simulate a single layer LiDAR, commonly employed in mobile robotics. A total of 110 images were manually annotated, 70 for training and 40 for testing. Details about the annotated frames are given in Table 5.1. Figure 1.3 (page 10) shows some annotations examples.

The training data were used to learn the probabilities and mass functions for the stereo, LiDAR and texture-based modules. They were also used to get the discounting factor of the monocular surface layout estimation. No training was needed for the pixel-based and temporal propagation modules. Each classification context introduced in Section 5.2 was considered as an individual module. Table 5.2 summarizes the frames of discernment of these different modules.

5.3.1 Ground detection

The first task was to evaluate ground detection. Modules 2, 3, 4 and 7 were first considered. Table 5.3 shows the results of the ground detection task. Some detection examples are shown in Figure 5.9. By considering the stereo module alone, about 10% of the segments were ignored due to lack of disparity estimation. The blue regions in the images in Figure 5.9b are the segments with high uncertainty. Typically, the disparities could not be estimated in some textureless regions such as the sky or on white building facades.

After adding the LiDAR module, the recall rate of the ground class was increased by more than 5%. For example, we can see in Figure 5.9c-ii that the bottom right part of the ground was detected by the LiDAR module but not by the stereo one. The LiDAR module also slightly increased the recall rate of the non-ground class ($\approx +0.2\%$). In Figure 5.9c-iii, we can see in the left part some laser impacts corresponding to some non-ground segments that were not detected by the stereo module. Finally, we can also observe a slight increase of the misclassification rate for the non-ground class ($\approx +0.2\%$). In the KITTI platform setup, the Velodyne LiDAR was installed on top of the car. This may explain that, in some particular cases, some small objects below the laser beam may be misdetected. In Figure 5.9c-i, we can see that the pole in the foreground was missed by the LiDAR sensor and thus classified as ground. Such minor issues may be dealt with by considering additional LiDAR layers.

(a) Results from the monocular surface layout estimation

(b) Vegetation detection using texture information

(c) Stereo-based ground detection at time $t-1$

(d) Temporal propagation of the stereo module

Figure 5.8: Classification from different modules. For (a), the RGB colors represent the probability of the vertical, ground and sky classes, respectively. For (b), the RGB colors represent the mass assigned to $\{\sout{\text{vegetation}}\}$, $\{\text{vegetation}\}$ and Ω_{V}, respectively. For (c) and (d), the RGB colors represent the mass assigned to $\{\sout{\text{ground}}\}$, $\{\text{ground}\}$ and Ω_{G} respectively.

Table 5.1: Annotated frames from the KITTI dataset. The highlighted rows correspond to the data used for testing.

Category	**Date**	**Seq.**	**Annotated frames**
Campus	2011-09-28	016	13, 144
Campus	2011-09-28	021	153
Campus	2011-09-28	038	29
City	2011-09-26	001	59, 107
City	2011-09-26	002	16, 56
City	2011-09-26	005	16, 56, 104, 153
City	2011-09-26	009	13, 58, 158, 265, 360, 370, 380, 390, 400, 412, 417
City	2011-09-26	011	10, 30, 50, 75, 100, 126, 150, 175, 190, 200
City	2011-09-26	013	14, 100, 143
City	2011-09-26	014	157, 200, 209
City	2011-09-26	017	32
City	2011-09-26	048	0, 21
City	2011-09-26	051	67, 86
City	2011-09-26	056	80, 158, 201
City	2011-09-26	057	41, 112
City	2011-09-26	059	26
City	2011-09-26	060	7
City	2011-09-26	084	248
City	2011-09-26	091	12, 85
City	2011-09-26	093	30, 303, 404
City	2011-09-26	095	126
City	2011-09-26	096	0, 92, 362
City	2011-09-26	104	16, 43, 239, 285
City	2011-09-26	106	1
City	2011-09-26	113	0
City	2011-09-26	117	103, 230, 384, 461, 594
City	2011-09-28	002	40, 50, 60, 70, 93, 317
City	2011-09-29	026	0
City	2011-09-29	071	11, 103, 318, 665, 906, 940
Residential	2011-09-26	019	329, 371
Residential	2011-09-26	020	0
Residential	2011-09-30	018	80, 192, 277, 329, 357, 496, 600, 650, 700, 750, 800, 850
Road	2011-09-26	015	167, 184, 220, 280
Road	2011-09-26	027	56
Road	2011-09-26	028	184, 231

Table 5.2: Frames of discernment of the different modules.

	Module	Frame of discernment
#1	Pixel	$\Omega_S = \{\text{sky}, \sout{\text{sky}}\}$
#2	Pixel	$\Omega_G = \{\text{ground}, \sout{\text{ground}}\}$
#3	Stereo	$\Omega_G = \{\text{ground}, \sout{\text{ground}}\}$
#4	LiDAR	$\Omega_G = \{\text{ground}, \sout{\text{ground}}\}$
#5	Surface	$\Lambda = \{\text{ground}, \text{vertical}, \text{sky}\}$
#6	Texture	$\Omega_V = \{\text{vegetation}, \sout{\text{vegetation}}\}$
#7	Optical flow	multiple

The third considered module was the pixel-based one, for which all the segments above the horizon line upper bound were assigned to the non-ground class. As this module did not provide any information about the ground class, the results for this class remained unchanged. However, an increase of more than 5% was observed for the recall rate of the non-ground class. In particular, additional information was provided by this module in the parts of the sky and the buildings that were not classified by the stereo or LiDAR module (see Fig 5.9d).

Finally, the temporal propagation increased the recall rate or both the ground and non-ground classes by about 2%. Less than 2% of the segments were left without decision. In this ground detection module case, all the modules were correctly defined in their initial frames of discernment. The use of upper and lower bounds for the distance to the ground plane in the evidential method yielded slightly better results than the probabilistic model. But overall, the results from the probabilistic and evidential approaches were very similar.

5.3.2 Addition of the sky class

The sky class was added to the system with the monocular surface layout estimation module. The pixel-based module applied to the sky class was also included. The classification results are detailed in Table 5.4 and some examples are shown in Figure 5.10.

As explained in Section 5.2.1, the outputs of the ground detection modules had to be transformed onto the new frame of discernment. In the probabilistic case, several effects could be noted. First, all the probabilities assigned to the non-ground class were divided by two and distributed to the vertical and sky classes. This resulted in over-confidence about the ground class. We can see from Table 5.4 that, after combining the monocular module with the ground detection ones, the recall rate of the ground class was increased by more than 10%. However, this came at the expense of a higher error rate ($\approx +5\%$) and a lower recall rate of the non-ground class ($\approx -5\%$).

We can see in Figure 5.10c that many non-ground regions were misclassified as ground.

A large increase of recall for the sky class ($\approx +10\%$) was also observed. This resulted from the combination of the two pixel based modules. The probability distribution resulting from their combination (5.8) always assigned more confidence to the sky class when a segment was not under the horizon line lower bound. We can see in Figure 5.10c-ii that a large part of the buildings was classified as sky.

The over-confidence in both the ground and sky classes led to a large decrease of the vertical class recall rate ($\approx -10\%$). This also led to a very low error rate for the vertical class. We can see that the percentage of ground and sky segments being misclassified as vertical structures became both very low. For the ground class, this can be justified by the combination with an additional ground detector. However, for the sky, the decrease of the error rate was actually artificial. The ground detection modules did not provide any information about the sky and the pixel-based module only corrected some misclassifications occurring at the lower part of the image. In the upper part of the image, the originally misclassified sky segments were corrected only because the probability of the vertical class was artificial decreased. Overall, the accuracy was still increased ($+1.7\%$) but the error distribution became completely different.

In the evidential case, the recall rates of all three classes were increased and their error rates were decreased. Moreover, the performance of the combined system remained coherent with respect to the performances of the individual modules. We can see that the percentage of sky segments being misclassified as vertical structures only decreased slightly (-2.2%). Overall, the accuracy was increased by about 4%.

5.3.3 Addition of the vegetation class

Finally, the vegetation detector was added. The results are shown in Table 5.5. The probabilistic combination led again to over-confidence in the sky class as the probabilities on the ground and vertical classes were both distributed to two finer classes. We obtained a 99.6% recall rate of the sky class but with a very large error rate of 12.1%. Again, the lower error rate of the obstacle class in the probabilistic case compared to the evidential one ($\approx -10\%$) is artificial and was induced by this over-confidence in the sky class. Moreover, the probability originally assigned to the vegetation class is distributed among two classes while the probability of the non-vegetation class is distributed among three classes. This explains the very low recall rate of the obstacle class. Again, the evidential fusion was more robust to refinements and led to better overall accuracy. In particular, as the vegetation module did not provide any information about the ground and vertical classes, the recall rate of these two classes remained unchanged in the eviden-

tial case. On the contrary, in the probabilistic case, the recall rates changed for both the ground and vertical classes. Some examples of classification are shown in Figure 5.11.

5.4 Discussion and conclusion

The computation of the over-segmentation, disparity and optical flow can be done in parallel. Real time implementations of these tasks are described in the literature [99, 126], but they have not be applied in this work. All the modules can also process data independently and in parallel, except the LiDAR module, which actually needs the ground plane estimation from the stereo module. For most of the modules, very simple methods were used, resulting in low computation time. The cost of mass function combination is linear in both the number of modules and the number of considered singletons, as only the plausibilities of singletons are computed.

One drawback of our approach is that we may not reach the best performance attainable given all the information at hand. All the modules are considered independently and only use a part of the available information. A global learning, as well as an optimized combination rule [98], could yield better results. It is, however, the price to pay if we want the system to be flexible enough to allow for the inclusion of new modules and new classes without having to retrain the whole system every time. Moreover, the complexity of a global approach would grow with the increasing number of modules and classes. The modular structure of the system also makes it more robust to the failure of a sensor, such as the LiDAR.

We have introduced an original framework for multimodal information fusion based on over-segmented images and Dempster-Shafer theory. This framework is very flexible as it makes it possible to include new classes, new sensors or new object detection algorithms without having to retrain the whole system. The information combination approach lends itself to parallel implementation and can cope with sensor failures. Future work will consider additional classes such as pedestrian. New sources of information such as GPS or digital maps will also be considered to detect moving objects. Finally, syntactic-based approaches such the one proposed in [11] will be further studied in order to merge segments belonging to the same object and allow for a deeper understanding of the scene.

Table 5.3: Classification results of the ground detection modules 2, 3, 4 and 7. The lines correspond to the decisions made by the system and the column to the actual classes. The figures represent the recall rates in percentage.

		Stereo		Stereo+LiDAR		Stereo+LiDAR +Pixel		Stereo+LiDAR +Pixel+Flow	
		ground	~~ground~~	ground	~~ground~~	ground	~~ground~~	ground	~~ground~~
Prob	ground	87.4	4.1	93.1	4.3	93.1	4.3	95.2	4.5
	~~ground~~	4.2	85.2	4.2	85.4	4.2	91.3	3.9	93.8
	ignore	8.4	10.7	2.7	10.3	2.7	4.4	0.9	1.7
Belief	ground	87.6	3.9	93.5	4.2	93.5	4.2	95.4	4.2
	~~ground~~	4.0	85.4	3.9	85.5	3.9	91.4	3.7	94.1
	ignore	8.4	10.7	2.6	10.3	2.6	4.4	0.9	1.7

Table 5.4: Classification results of the combination of the surface layout estimation module with the ground detection ones. The figures represent the recall and error rates in percentage. The numbers in brackets correspond to the overall accuracy.

	Surface layout (90.5%)				Probabilistic fusion (92.2%)				Evidential fusion (94.3%)			
	ground	vert.	sky	error	ground	vert.	sky	error	ground	vert.	sky	error
ground	85.0	4.0	0.0	4.5	98.5	10.8	0.0	9.9	94.4	3.7	0.0	3.8
vert.	15.0	95.0	12.8	22.6	1.5	86.8	2.6	4.5	5.6	95.3	10.6	14.5
sky	0.0	1.0	87.2	1.1	0.0	2.5	97.4	2.5	0.0	1.0	89.4	1.1
		96.6				91.0				96.9		
		~~ground~~				~~ground~~				~~ground~~		

Table 5.5: Results of the combination of all the modules. The figures represent the recall and error rates in percentage. The numbers in brackets correspond to the overall accuracy.

	Probabilistic fusion (79.0%)						Evidential fusion (81.4%)					
	grass	road	tree	obst.	sky	error	grass	road	tree	obst.	sky	error
ground	86.4	3.7	4.0	5.3	0.0	13.1	73.3	1.7	1.4	1.7	0.0	6.1
road	7.0	95.0	0.4	7.6	0.0	13.6	11.2	94.5	0.4	4.2	0.0	14.3
tree	6.2	0.5	80.2	27.0	0.0	29.6	12.7	0.6	75.3	21.3	0.0	31.5
obst.	0.4	0.8	14.6	47.2	0.4	25.6	2.8	3.2	22.8	70.7	10.6	35.8
sky	0.0	0.0	0.8	12.9	99.6	12.1	0.0	0.0	0.0	2.0	89.4	2.2
	97.8		85.2				94.4		95.3			
	ground		vertical				ground		vertical			

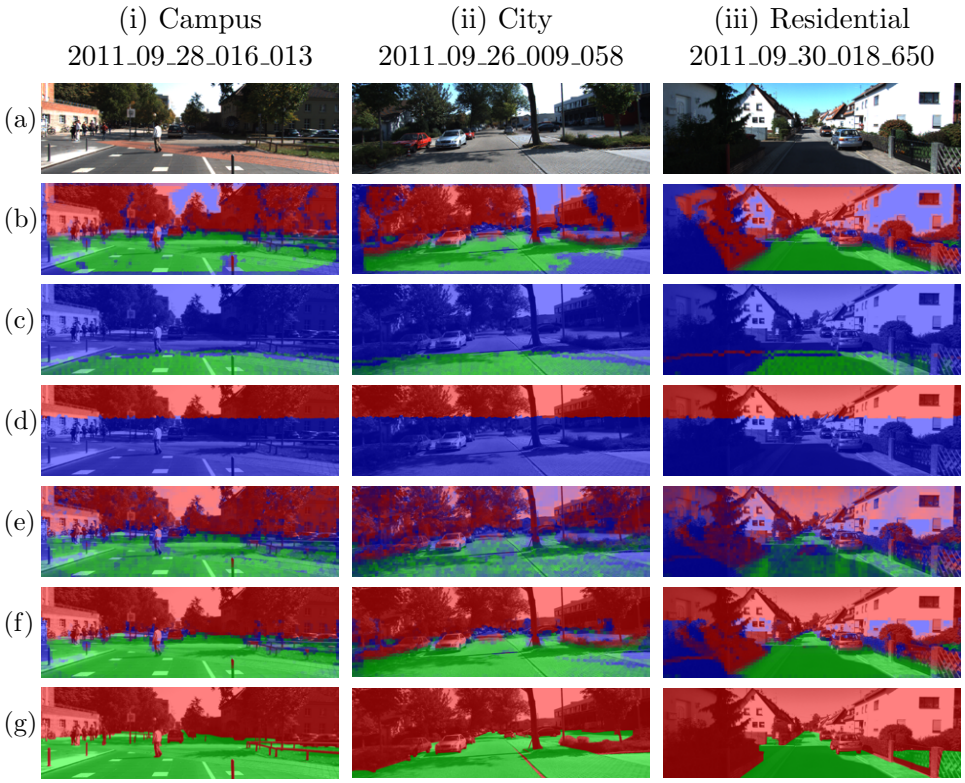


Figure 5.9: Classification from the ground detection modules. The RGB colors represent the mass assigned to {~~ground~~}, {ground} and Ω_{G}, respectively. (a) Raw images. (b) Stereo-based module. (c) LiDAR module. (d) Pixel-based module. (e) Temporal propagation of the combined mass function from the previous frame. (f) Combined mass functions. (g) Ground truth images.

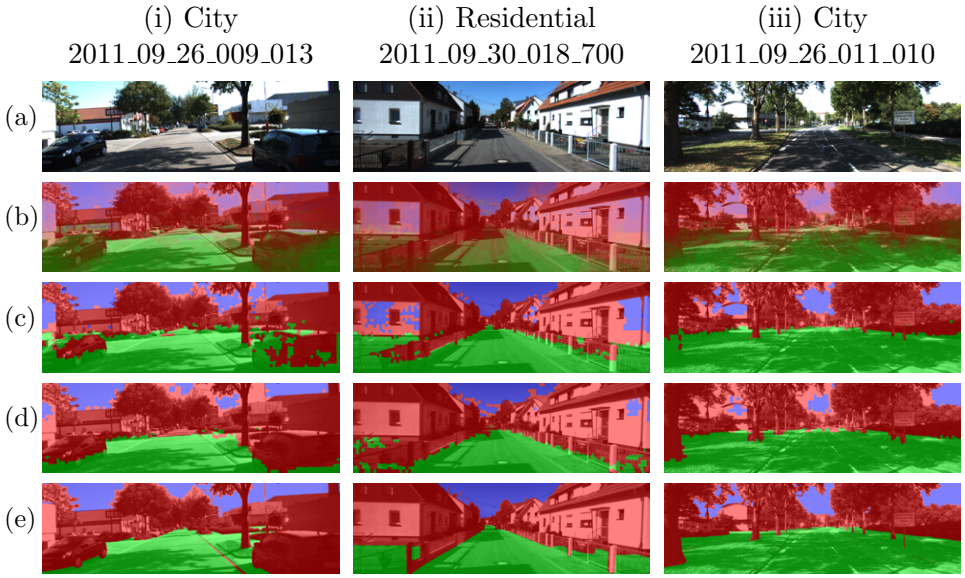


Figure 5.10: Classification from different modules. The color code for (c), (d) and (e) is defined as follows: ground = green, vertical = red, sky = blue. (a) Raw image. (b) Output of the monocular surface layout estimation module, the RGB colors represent the probabilities assigned to the ground, vertical and sky classes, respectively. (c) Decisions resulting from the probabilistic combination with the ground detection modules. (d) Decision results from the evidential combination. (e) Ground truth images.

(a) Probabilistic (b) Evidential (c) Ground truth

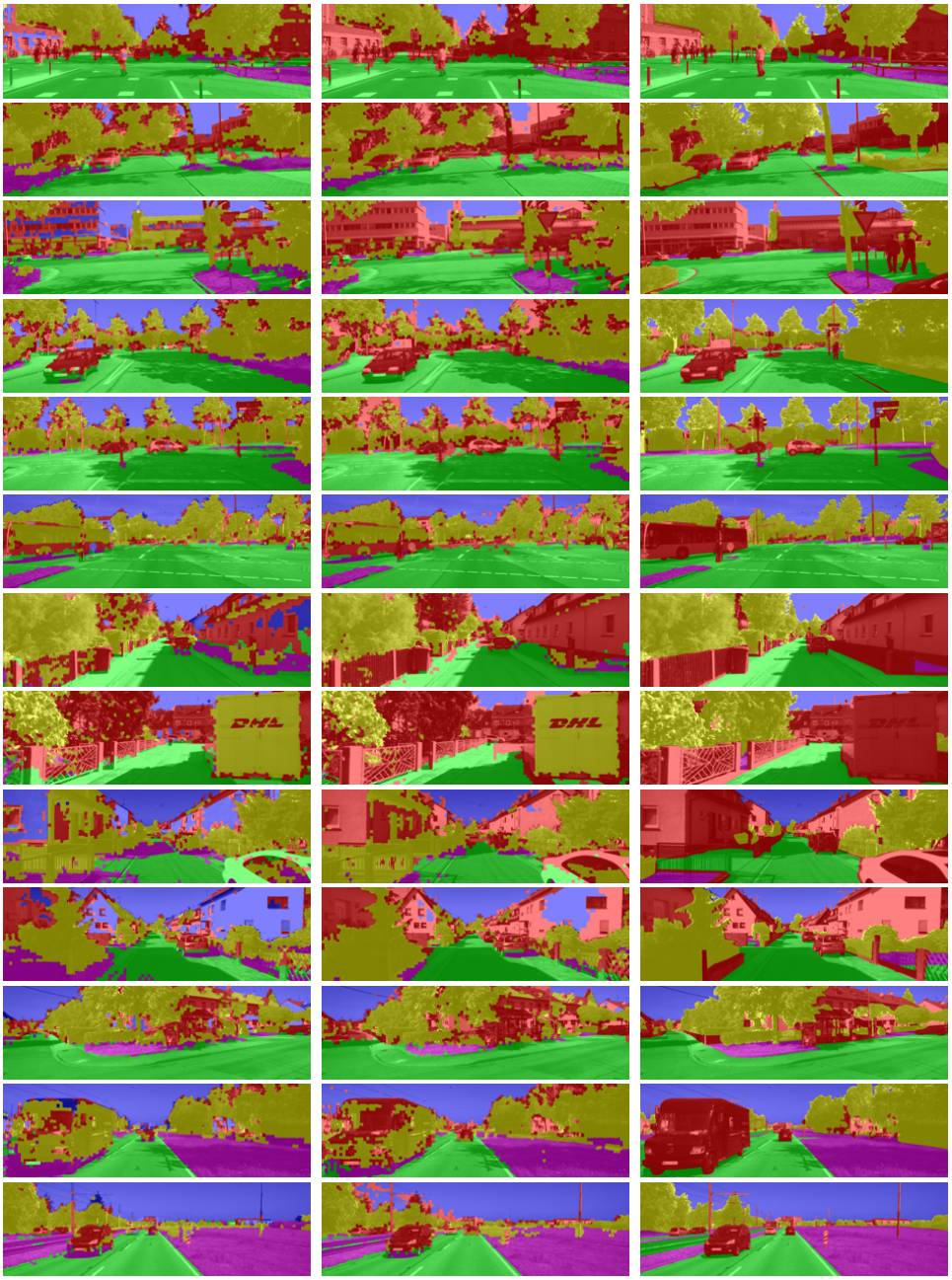


Figure 5.11: Classification results considering all the modules. The color code is defined as follows: grass = magenta, road = green, tree = yellow, obstacle = red, sky = blue.

Conclusion and future work

In this thesis, we addressed scene understanding through an information fusion point of view. Scene understanding was considered as an image understanding problem at two levels: object detection and semantic segmentation. The main contribution of this thesis was to propose an evidential extension of probabilistic calibration methods and to apply it to object detection and semantic segmentation. The evidential calibration methods developed in this thesis can be used to transform the output of any classifier into a belief function. Belief functions can represent many kinds of imperfect information that are not properly handled by classical probabilities. Belief functions can better represent the uncertainty due to small training dataset. Also, contrary to probability distributions, belief functions can be made to support a unique hypothesis without being certain. The theory of belief functions also offers a high flexibility in terms of class definition. Refinement of frames of discernment is properly handled by belief functions contrary to probability. All these properties made the theory of belief functions especially well suited for the combination of object detectors and semantic segmentation methods.

Many issues addressed in this thesis are, however, still open problems. Several aspects are currently under study and many perspectives will be studied in future work.

Belief functions The evidential calibration methods proposed in this thesis are extensions of probabilistic approaches. The mean-squared error, also called Brier score [13], is a conventional measure used to compare an estimated probability distribution to a reference one, i.e., ground truth. With Dempster-Shafer theory, there is no consensus over the definition of a well calibrated belief function. In particular, the distance between a belief function and a ground truth can be formulated in many different ways [58].

Most of the distance measures proposed in the literature use some structural properties by encoding the relations between focal elements. In particular, the cardinality of focal elements are often taken into account. This last point implies that those distance measures can not be consistent over refinement. For most of the distance measures d proposed in [58], we can show that it is always possible to find two mass functions m_1^Ω and m_2^Ω, and a reference one m_*^Ω, such that $d(m_1^\Omega, m_*^\Omega) < d(m_2^\Omega, m_*^\Omega)$ but $d(m_1^\Theta, m_*^\Theta) > d(m_2^\Theta, m_*^\Theta)$

for a particular refinement Θ of Ω. Only, the simple Euclidean distance: $d(m_1, m_2)^2 = \sum_{A \subseteq \Omega}(m_1(A) - m_2(B))^2$, remains consistent over refinement. However, it lacks other important properties. In our particular case, as the frame of discernment needs to be refined every time new classes are defined, the consistency with respect to refinement is an important issue. One way to address this issue is to compute distances over a coarsening that will have only singleton as atomic focal element. This idea will be investigated in future work.

Multi-class calibration was addressed from a binary decomposition point of view. However, not all multi-class classifiers use such an approach. A more general formulation would be to calibrate a vector of scores, which would represent the confidence degree in each class, into a multi-class belief function. This aspect is still an open issue.

Object detection We showed that the combination of multiple detectors can greatly increase the performance. The method we proposed to combine object detectors can also be used to better measure the contribution of a new algorithm. In Figure 5.12, we show the performance evolution of pedestrian detectors over time. We can see from the blue curve, which represents the log-average miss rate, that newer detectors are not always better than existing ones. However, the red curve, which represents the performance of the combination of the detectors, has clearly a decreasing shape. In the bottom graph of Figure 5.12, we show the difference in performance of each detector compared to the previously best existing method (blue bar) and the gain obtained from the combination of the detectors (red bar). We can see that even though many methods perform worse than existing methods, their contribution in the combination leads to a positive gain. It is the other way round for a few methods. For example, we can see that the 'MultiFtr' (#25) algorithm has a 5% lower miss rate compared to existing methods but its combination with them does not lead to any gain. This can be easily understood as the 'MultiFtr' algorithm only aggregates existing features. Conversely, methods that introduce new features or information, such as 'HOG' (#2), 'LatSvm-V2' (#5), 'MultiFtr+Motion' (#27), 'MultiResC' (#6) or 'ACF+SDt' (#24), have the highest contribution. The novelty of a detector can thus be measured by its contribution when combined with existing methods.

Our fusion framework is not limited to the combination of pedestrian detectors and can be easily extended to other classes of objects. Yan *et al.* [133] used a car detector to help pedestrian detectors. Modeling the interactions between different detected objects will be considered in future work. The tracking of detected objects is also an important aspect to address. In particular, the prediction at time t of the positions of objects detected at time $t-1$ can be seen as a detection module itself.

Finally, one may argue that it is unreasonable to run more than 30 independent detectors. Selecting only a subset of classifiers would be more acceptable in practice. Classifier selection is yet another challenging task [64]. Moreover, depending on the context (scale, aspect ratio, occlusion), the optimal subset of classifiers may vary. This issue remains an open problem and will be investigated in future work.

Semantic segmentation We showed that our evidential fusion framework is flexible enough to easily include new classes of objects, new sensors and new detection modules. In [12], we additionally used the Automatic Labeling Environment (ALE) software [68] to have a total of 14 classes. Figure 5.13b shows a classification result obtained from ALE. The combination with other algorithms [46, 115, 116] will also be considered in future work. In order to take into account the interaction between neighboring segments, a global optimization step is often considered in semantic segmentation. For example, conditional random fields (CRF) were used in [68]. An evidential extension of CRF could be considered in our case. However, one limitation of such approaches is that it is not possible to segment out individual object. The particular structural properties of some classes are not taken into account. The computation of a complete *and*/*or* parse graph using visual grammars is one way to get a deeper understanding of an image. We extended stochastic grammars to evidential grammars in [11]. Figure 5.13c shows a preliminary result obtained from an evidential grammar based semantic segmentation.

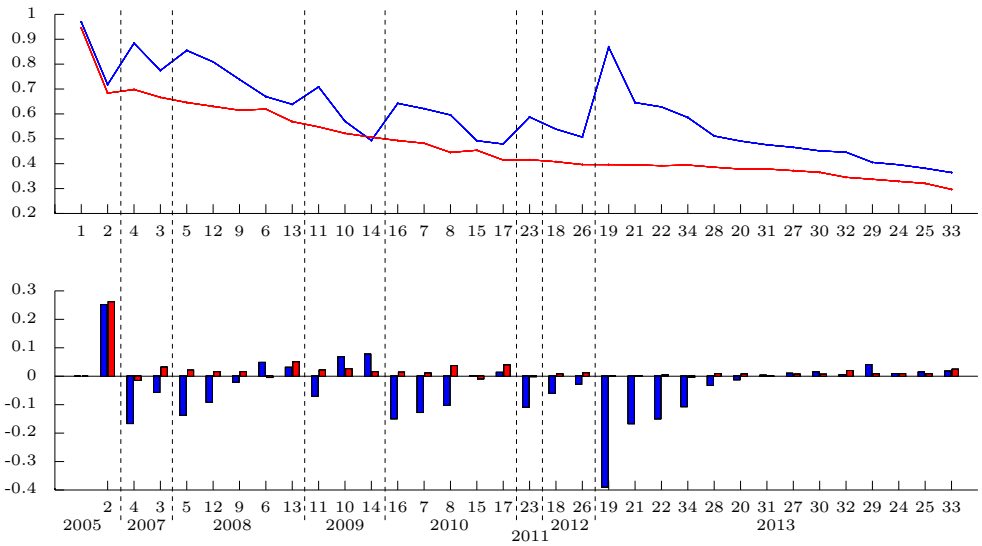
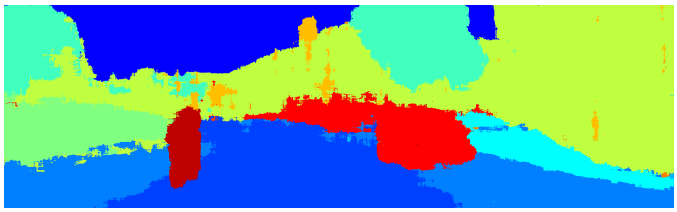
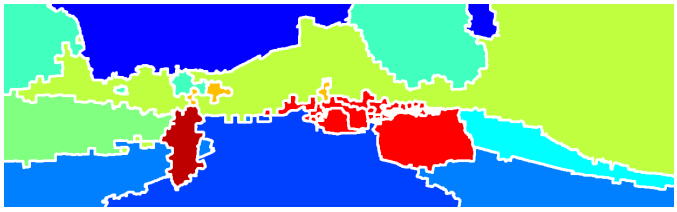


Figure 5.12: Performance evolution of pedestrian detectors over time. The numbers on the x-axis corresponds to the index number of the pedestrian detectors (see Table 4.1). The methods are sorted by their year of publication. All the methods published in a same year are sorted by increasing order of performance, i.e., by decreasing order of log-average miss rate. (Top graph) The blue curve represents the performance of the detectors in terms of log-average miss rate. The red curve represents the performance of the iterative combination of the detectors by Dempster's rule. The y-axis corresponds to the log-average miss rate. (Bottom graph) The blue bars represent the absolute difference in performance compared to the previously best existing method. The red bars represent the absolute performance gain obtained from the combination of detectors. The y-axis corresponds to absolute difference in terms of log-average miss rate.

(a) Raw image



(b) ALE classification result



(c) Semantic segmentation using evidential grammar

Figure 5.13: Semantic segmentation using ALE and an evidential grammar based global optimization.

Appendix

Proof of Proposition 1.1

Proposition 1.1 *Let $P_{\mathbf{x_1}}$ and $P_{\mathbf{x_2}}$ be two binary probability distributions. Their combination follows the following ordering:*

$$H(P_{\mathbf{x_1}} \boxtimes P_{\mathbf{x_2}}) \leq H(P_{\mathbf{x_1}} \wedge P_{\mathbf{x_2}}) \leq H(P_{\mathbf{x_1}} \boxplus P_{\mathbf{x_2}}) \leq H(P_{\mathbf{x_1}} \vee P_{\mathbf{x_2}}).$$

We define $P_{\mathbf{x_1}}$ and $P_{\mathbf{x_2}}$ as

$$\begin{aligned} P_{\mathbf{x_1}}(Y=1) &= p, \qquad P_{\mathbf{x_1}}(Y=0) = 1-p, \\ P_{\mathbf{x_2}}(Y=1) &= q, \qquad P_{\mathbf{x_2}}(Y=0) = 1-q, \end{aligned}$$

where $p, q \in [0,1]$. As $P_{\mathbf{x_1}}$ and $P_{\mathbf{x_2}}$ play symmetric role, we can assume, without loss of generality, that $p \leq q$.

To prove Proposition 1.1, we can first prove the following two lemmas.
Lemma 1 *Let P a binary probability distribution defined as $P(Y=1) = p$. The entropy function $H(P) = -p\ln p - (1-p)\ln(1-p)$ (see Definition 1.2) is increasing for $p \in [0, 1/2]$ and decreasing for $p \in [1/2, 1]$.*

Proof.

$$\begin{aligned} \frac{dH(P)}{dp}(p) \geq 0 \quad &\Leftrightarrow \quad -\ln p - 1 + \ln(1-p) + 1 \geq 0 \\ &\Leftrightarrow \quad \ln\left(\frac{1}{p} - 1\right) \geq 0 \\ &\Leftrightarrow \quad p \leq 1/2. \end{aligned}$$

□

Lemma 2 *Let $*$ be any of the four combination rules: $\boxtimes$, $\boxplus$, $\wedge$ and $\vee$. The following property holds:*

$$(P_{\mathbf{x_1}} * P_{\mathbf{x_2}})(Y=1) \leq \frac{1}{2} \quad \Leftrightarrow \quad p+q \leq 1.$$

Proof.

- $(P_{\mathbf{x_1}} \boxtimes P_{\mathbf{x_2}})(Y=1) \leq \frac{1}{2} \Leftrightarrow \frac{pq}{pq+(1-p)(1-q)} \leq \frac{1}{2}$
  $\Leftrightarrow pq \leq \frac{1-p-q+2pq}{2}$
  $\Leftrightarrow p+q \leq 1.$

- $(P_{\mathbf{x_1}} \boxplus P_{\mathbf{x_2}})(Y=1) \leq \frac{1}{2} \Leftrightarrow \frac{p+q}{p+q+(1-p)+(1-q)} \leq \frac{1}{2}$
  $\Leftrightarrow p+q \leq 1.$

- $(P_{\mathbf{x_1}} \wedge P_{\mathbf{x_2}})(Y=1) \leq \frac{1}{2} \Leftrightarrow \frac{p \wedge q}{(p \wedge q)+((1-p) \wedge (1-q))} \leq \frac{1}{2}$
  $\Leftrightarrow p \leq \frac{p+1-q}{2}$
  $\Leftrightarrow p+q \leq 1.$

- $(P_{\mathbf{x_1}} \vee P_{\mathbf{x_2}})(Y=1) \leq \frac{1}{2} \Leftrightarrow \frac{p \vee q}{(p \vee q)+((1-p) \vee (1-q))} \leq \frac{1}{2}$
  $\Leftrightarrow q \leq \frac{q+1-p}{2}$
  $\Leftrightarrow p+q \leq 1.$

□

Proof of Proposition 1.1. We first consider the case where $p+q \leq 1$. Using Lemma 1 and Lemma 2, proving Proposition 1.1 becomes equivalent to proving the following ordering:

$$(P_{\mathbf{x_1}} \boxtimes P_{\mathbf{x_2}})(Y=1) \leq (P_{\mathbf{x_1}} \wedge P_{\mathbf{x_2}})(Y=1) \leq (P_{\mathbf{x_1}} \boxplus P_{\mathbf{x_2}})(Y=1) \leq (P_{\mathbf{x_1}} \vee P_{\mathbf{x_2}})(Y=1).$$

- $(P_{\mathbf{x_1}} \boxtimes P_{\mathbf{x_2}})(Y=1) \leq (P_{\mathbf{x_1}} \wedge P_{\mathbf{x_2}})(Y=1)$
  $\Leftrightarrow \frac{pq}{pq+(1-p)(1-q)} \leq \frac{p \wedge q}{(p \wedge q)+((1-p) \wedge (1-q))}$
  $\Leftrightarrow \frac{pq}{pq+(1-p)(1-q)} \leq \frac{p}{p+1-q}$
  $\Leftrightarrow q(p+1-q) \leq pq+(1-p)(1-q)$
  $\Leftrightarrow p+q \leq 1$, which is assumed to be true.

- $(P_{\mathbf{x_1}} \wedge P_{\mathbf{x_2}})(Y=1) \leq (P_{\mathbf{x_1}} \boxplus P_{\mathbf{x_2}})(Y=1)$
  $\Leftrightarrow \quad \frac{p \wedge q}{(p \wedge q) + ((1-p) \wedge (1-q))} \leq \frac{p+q}{p+q+(1-p)+(1-q)}$
  $\Leftrightarrow \quad \frac{p}{p+1-q} \leq \frac{p+q}{2}$
  $\Leftrightarrow \quad 2p \leq (p+q)(p+1-q)$
  $\Leftrightarrow \quad 2p \leq p^2 + p - pq + qp + q - q^2$
  $\Leftrightarrow \quad 0 \leq p^2 - q^2 + q - p$
  $\Leftrightarrow \quad 0 \leq (q-p)(1-(p+q)),$
  which is true as we have assumed that $p+q \leq 1$ and $p \leq q$.

- $(P_{\mathbf{x_1}} \boxplus P_{\mathbf{x_2}})(Y=1) \leq (P_{\mathbf{x_1}} \vee P_{\mathbf{x_2}})(Y=1)$
  $\Leftrightarrow \quad \frac{p+q}{p+q+(1-p)+(1-q)} \leq \frac{p \vee q}{(p \vee q) + ((1-p) \vee (1-q))}$
  $\Leftrightarrow \quad \frac{p+q}{2} \leq \frac{q}{q+1-p}$
  $\Leftrightarrow \quad (p+q)(q+1-p) \leq 2q$
  $\Leftrightarrow \quad pq + p - p^2 + q^2 + q - qp \leq 2q$
  $\Leftrightarrow \quad q^2 - p^2 + p - q \leq 0$
  $\Leftrightarrow \quad (p-q)(1-(p+q)) \leq 0,$
  which is true as we have assumed that $p+q \leq 1$ and $p \leq q$.

In the other case where $p+q \geq 1$. Using Lemma 1 and Lemma 2, proving Proposition 1.1 becomes equivalent to proving the following ordering:

$$(P_{\mathbf{x_1}} \boxtimes P_{\mathbf{x_2}})(Y=1) \geq (P_{\mathbf{x_1}} \wedge P_{\mathbf{x_2}})(Y=1) \geq (P_{\mathbf{x_1}} \boxplus P_{\mathbf{x_2}})(Y=1) \geq (P_{\mathbf{x_1}} \vee P_{\mathbf{x_2}})(Y=1).$$

It can be proved in the same way as done previously.

□

Bibliography

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, Nov 2012.

[2] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, Alaska, June 2008.

[3] K. Arras, O. Mozos, and W. Burgard. Using boosted features for the detection of people in 2D range data. In *IEEE International Conference on Robotics and Automation*, pages 3402–3407, Roma, Italy, April 2007.

[4] M. Ayer, H. Brunk, G. Ewing, W. Reid, and E. Silverman. An empirical distribution function for sampling with incomplete information. *Annals of Mathematical Statistics*, 5:641–647, 1955.

[5] H. Badino, U. Franke, and R. Mester. Free space computation using stochastic occupancy grids and dynamic programming. In *Proceedings of the International Conference on Computer Vision Workshop on Dynamical Vision*, Rio de Janeiro, Brazil, 2007.

[6] A. Bar-Hillel, D. Levi, E. Krupka, and C. Goldberg. Part-based feature synthesis for human detection. In *Proceedings of the European Conference on Computer Vision*, pages 4.5–4.11, Crete, Greece, 2010.

[7] J. A. Barnett. Calculating Dempster-Shafer plausibility. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):599–602, 1991.

[8] A. Bella, C. Ferri, J. Hernández-Orallo, and M. J. Ramírez-Quintana. On the effect of calibration in classifier combination. *Applied Intelligence*, 38(4):566–585, 2013.

[9] R. Benenson, M. Mathias, T. Tuytelaars, and L. Van Gool. Seeking the strongest rigid detector. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3666–3673, Portland, Oregon, June 2013.

[10] M. Bertozzi, A. Broggi, A. Fascioli, T. Graf, and M. Meinecke. Pedestrian detection for driver assistance using multiresolution infrared vision. *IEEE Transactions on Vehicular Technology*, 53(6):1666–1678, Nov 2004.

[11] J.-B. Bordes, F. Davoine, Ph. Xu, and T. Denœux. Evidential grammars for image interpretation - Application to multimodal traffic scene understanding. In *Z. Qin and V. N. Huyn (Eds), Integrated Uncertainty in Knowledge Modelling and Decision Making*, pages 65–78, Beijing, China, July 2013.

[12] J.-B. Bordes, Ph. Xu, F. Davoine, H. Zhao, and T. Denœux. Information fusion and evidential grammars for object class segmentation. In *Fifth IROS Workshop on Planning, Perception and Navigation for Intelligent Vehicles*, pages 165–170, Tokyo, Japan, November 2013.

[13] G. W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

[14] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27.1–27.27, 2011.

[15] G. Chen, Y. Ding, J. Xiao, and T. Han. Detection evolution with multi-order contextual co-occurrence. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1798–1805, Portland, Oregon, June 2013.

[16] C. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26:404–413, 1934.

[17] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, May 2002.

[18] F. Cuzzolin. Lp consonant approximations of belief functions. *IEEE Transactions on Fuzzy Systems*, 22(2):420–436, April 2014.

[19] N. Dalal. *Finding people in images and videos.* PhD thesis, Institut National Polytechnique de Grenoble, France, 2006.

[20] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 886–893, San Diego, California, June 2005.

[21] A. Dempster. The Dempster-Shafer calculus for statisticians. *International Journal of Approximate Reasoning*, 48(2):365–377, 2008.

[22] A. P. Dempster. New methods for reasoning towards posterior distributions based on sample data. *Annals of Mathematical Statistics*, 37(2):355–374, 1966.

[23] T. Denœux. A k-nearest neighbor classification rule based on Dempster-Shafer theory. *IEEE Transactions on Systems, Man and Cybernetics*, 25(5):804–813, 1995.

[24] T. Denœux. Analysis of evidence-theoretic decision rules for pattern classification. *Pattern Recognition*, 30(7):1095–1107, 1997.

[25] T. Denœux. A neural network classifier based on dempster-shafer theory. *IEEE Transactions on Systems, Man and Cybernetics*, 30(2):131–150, 2000.

[26] T. Denœux. Conjunctive and disjunctive combination of belief functions induced by non distinct bodies of evidence. *Artificial Intelligence*, 172:234–264, 2008.

[27] T. Denœux. Likelihood-based belief function: justification and some extensions to low-quality data. *International Journal of Approximate Reasoning*, 55(7):1535–1547, 2014.

[28] T. Denœux, N. El Zoghby, V. Cherfaoui, and A. Jouglet. Optimal object association in the dempster-shafer framework. *IEEE Transactions on Cybernetics*, 2014. (accepted for publication).

[29] P. Dollár, R. Appel, and W. Kienzle. Crosstalk cascades for frame-rate pedestrian detection. In *Proceedings of the European Conference on Computer Vision*, pages 645–659, Florence, Italy, 2012.

[30] P. Dollár, S. Belongie, and P. Perona. The fastest pedestrian detector in the west. In *Proceedings of the British Machine Vision Conference*, pages 68.1–68.11, Aberystwyth, Wales, 2010.

[31] P. Dollár, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *Proceedings of the British Machine Vision Conference*, pages 91.1–91.11, London, England, 2009.

[32] P. Dollar, Z. Tu, H. Tao, and S. Belongie. Feature mining for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, Minnesota, June 2007.

[33] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):743–761, April 2012.

[34] D. Dubois and H. Prade. A set-theoretic view of belief functions: logical operations and approximations by fuzzy sets. *International Journal of General Systems*, 12(3):193–226, 1986.

[35] D. Dubois and H. Prade. Consonant approximations of belief functions. *International Journal of Approximate Reasoning*, 4(5–6):419–449, 1990.

[36] D. Dubois, H. Prade, and Ph. Smets. A definition of subjective possibility. *International Journal of Approximate Reasoning*, 48(2):352–364, 2008.

[37] R. P. W. Duin. The combining classifier: to train or not to train? In *Proceedings of the International Conference on Pattern Recognition*, volume 2, pages 765–770, Quebec, Canada, 2002.

[38] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part i. *IEEE Robotics Automation Magazine*, 13(2):99–110, June 2006.

[39] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, June 1989.

[40] M. Enzweiler and D. M. Gavrila. Monocular pedestrian detection: Survey and experiments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2179–2195, Dec 2009.

[41] E. Erdem, S. Dubuisson, and I. Bloch. Visual tracking by fusing multiple cues with context-sensitive reliabilities. *Pattern Recognition*, 45(5):1948–1959, 2012.

[42] A. Ess, B. Leibe, K. Schindler, and L. Van Gool. A mobile vision system for robust multi-person tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, Alaska, June 2008.

[43] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL visual object classes challenge: A retrospective. *International Journal of Computer Vision*, pages 1–39, 2014.

[44] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, 2012.

[45] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[46] C. Farabet. *Towards Real-Time Image Understanding with Convolutional Networks*. PhD thesis, Université Paris-Est, France, 2013.

[47] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[48] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[49] D. Forsyth, J. Malik, M. Fleck, H. Greenspan, T. Leung, S. Belongie, C. Carson, and C. Bregler. Finding pictures of objects in large collections of images. In J. Ponce, A. Zisserman, and M. Hebert, editors, *Object Representation in Computer Vision II*, volume 1144 of *Lecture Notes in Computer Science*, pages 335–360. Springer Berlin Heidelberg, 1996.

[50] D. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *International Journal of Computer Vision*, 73(1):41–59, 2007.

[51] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[52] A. Geiger, M. Roser, and R. Urtasun. Efficient large-scale stereo matching. In *Proceedings of the Asian Conference on Computer Vision*, pages 25–38, Queenstown, New Zealand, 2010.

[53] A. Geiger, J. Ziegler, and C. Stiller. Stereoscan: Dense 3D reconstruction in real-time. In *IEEE Intelligent Vehicles Symposium*, pages 963–968, Baden-Baden, Germany, June 2011.

[54] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf. Survey on pedestrian detection for advanced driver assistance systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1239–1258, 2010.

[55] E. Haenni. Are alternatives to Dempster's rule of combination real alternatives? *Information Fusion*, 3:237–239, 2002. Comments on "About the belief function combination and the conflict management problem".

[56] T. Hastie and R. Tibshirani. Generalized additive models. *Statistical Science*, 1(3):297–318, 1986.

[57] D. Hoiem, A. Efros, and M. Hebert. Recovering surface layout from an image. *International Journal of Computer Vision*, 75(1):151–172, 2007.

[58] A. Jousselme and P. Maupin. Distances in evidence theory: Comprehensive survey and generalizations. *International Journal of Approximate Reasoning*, 53(2):118–145, 2012.

[59] O. Kanjanatarakul, S. Sriboonchitta, and T. Denœux. Forecasting using belief functions: an application to marketing econometrics. *International Journal of Approximate Reasoning*, 55(5):1113–1128, 2014.

[60] J. M. Keynes. *A Treatise on Probability*. Macmillan and Company, Limited, 1921.

[61] B. Khaleghi, A. Khamis, F. Karray, and S. Razavi. Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, 14(1):28–44, 2013.

[62] S. Kim, S. young Oh, J. Kang, Y. Ryu, K. Kim, S.-C. Park, and K. Park. Front and rear vehicle detection and tracking in the day and night times using vision and sonar sensor fusion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2173–2178, Edmonton, Alberta, Canada, Aug 2005.

[63] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, March 1998.

[64] L. I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.

[65] C.-H. Kuo and R. Nevatia. Robust multi-view car detection using unsupervised sub-categorization. In *Workshop on Applications of Computer Vision (WACV)*, pages 1–8, Snowbird, Utah, Dec 2009.

[66] M. Kurdej, J. Moras, V. Cherfaoui, and P. Bonnifait. Controlling remanence in evidential grids using geodata for dynamic scene perception. *International Journal of Approximate Reasoning*, 55(1):355–375, January 2014. Available online 31 March 2013.

[67] R. Labayrade, D. Aubert, and J. P. Tarel. Real time obstacle detection in stereovision on non flat road geometry through "v-disparity" representation. In *IEEE Intelligent Vehicle Symposium*, volume 2, pages 646–651, Versailles, France, June 2002.

[68] L. Ladický. *Global structured models towards scene understanding*. PhD thesis, Oxford Brookes University, 2011.

[69] L. Ladický, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. Torr. Joint optimization for object class segmentation and dense stereo reconstruction. *International Journal of Computer Vision*, 100(2):122–133, 2012.

[70] D. Larlus, J. Verbeek, and F. Jurie. Category level object segmentation by combining bag-of-words models with dirichlet processes and random fields. *International Journal of Computer Vision*, 88(2):238–253, 2010.

[71] E. Lefèvre, O. Colot, and P. Vannoorenberghe. Belief function combination and conflict management. *Information Fusion*, 3(2):149–162, 2002.

[72] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool. Dynamic 3D scene analysis from a moving vehicle. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, USA, June 2007.

[73] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *Proceedings of ECCV Workshop on Statistical Learning in Computer Vision*, pages 1–16, Prague, Czech Republic, May 2004.

[74] D. Levi, S. Silberstein, and A. Bar-Hillel. Fast multiple-part based object detection using kd-ferns. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 947–954, Portland, Oregon, June 2013.

[75] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi. Turbopixels: Fast superpixels using geometric flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2290–2297, Dec 2009.

[76] Z. Lin and L. Davis. A pose-invariant descriptor for human detection and segmentation. In *Proceedings of the European Conference on Computer Vision*, Marseille, France, 2008.

[77] S. Maji, A. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, Alaska, June 2008.

[78] MobileEye. http://www.mobileye.com/.

[79] A. P. Moore, S. J. D. Prince, J. Warrell, U. Mohammed, and G. Jones. Scene shape priors for superpixel segmentation. In *IEEE International*

Conference on Computer Vision, pages 771–778, Kyoto, Japan, Sept 2009.

[80] J. Moras. *Grilles de perception évidentielles pour la navigation robotique en milieu urbain.* PhD thesis, Univerité de Technologie de Compiègne, Compiègne, France, Jan 2013.

[81] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. *PAMI*, 27(11):1832–1837, 2005.

[82] D. Munoz, J. A. Bagnell, and M. Hebert. Co-inference machines for multi-modal scene analysis. In *European Conference on Computer Vision*, pages 668–681, Florence, Italy, 2012.

[83] W. Nam, B. Han, and J. H. Han. Improving object localization using macrofeature layout selection. In *ICCV Workshop on Visual Surveillance*, Barcelona, Spain, 2011.

[84] L. E. Navarro-Serment, C. Mertz, and M. Hebert. Pedestrian detection and tracking using three-dimensional ladar data. In A. Howard, K. Iagnemma, and A. Kelly, editors, *Field and Service Robotics*, volume 62 of *Springer Tracts in Advanced Robotics*, pages 103–112. Springer Berlin Heidelberg, 2010.

[85] H. T. Nguyen. *An Introduction to Random Sets.* Chapman and Hall/CRC Press, Boca Raton, Florida, 2006.

[86] A. Niculescu-Mizil and R. Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the International Conference on Machine Learning*, pages 625–632, Bonn, Germany, 2005.

[87] W. Ouyang and X. Wang. A discriminative deep model for pedestrian detection with occlusion handling. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3258–3265, Providence, Rhode Island, June 2012.

[88] W. Ouyang and X. Wang. Single-pedestrian detection aided by multi-pedestrian detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3198–3205, Portland, Oregon, June 2013.

[89] W. Ouyang, X. Zeng, and X. Wang. Modeling mutual visibility relationship with a deep model in pedestrian detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3222–3229, Portland, Oregon, June 2013.

[90] S. Paisitkriangkrai, C. Shen, and A. van den Hengel. Efficient pedestrian detection by directly optimize the partial area under the roc curve. In *IEEE International Conference on Computer Vision*, pages 1057–1064, Sydney, Australia, Dec 2013.

[91] D. Park, D. Ramanan, and C. Fowlkes. Multiresolution models for object detection. In *Proceedings of the European Conference on Computer Vision*, pages 241–254, Crete, Greece, 2010.

[92] D. Park, C. Zitnick, D. Ramanan, and P. Dollár. Exploring weak stabilization for motion feature extraction. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2882–2889, Portland, Oregon, June 2013.

[93] S. J. Park, T. Y. Kim, S. M. Kang, and K. H. Koo. A novel signal processing technique for vehicle detection radar. In *IEEE MTT-S International Microwave Symposium Digest*, volume 1, pages 607–610, Philadelphia, Pennsylvania, June 2003.

[94] P. Perez, J. Vermaak, and A. Blake. Data fusion for visual tracking with particles. *Proceedings of the IEEE*, 92(3):495–513, Mar 2004.

[95] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In A. J. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large-Margin Classifiers*, pages 61–74. MIT Press, 1999.

[96] B. Quost, T. Denœux, and M. Masson. One-against-all combination in the framework of belief functions. In *Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knownledge-Based Systems*, volume 1, pages 356–363, Paris, France, 2006.

[97] B. Quost, T. Denœux, and M.-H. Masson. Pairwise classifier combination using belief functions. *Pattern Recognition Letters*, 28(5):644–653, 2007.

[98] B. Quost, M.-H. Masson, and T. Denœux. Classifier fusion in the Dempster-Shafer framework using optimized t-norm based combination rules. *International Journal of Approximate Reasoning*, 52(3):353–374, 2011.

[99] C. Y. Ren and I. Reid. gSLIC: a real-time implementation of SLIC superpixel segmentation. Technical report, University of Oxford, Department of Engineering Science, June 2011.

[100] S. A. Rodríguez Flórez, V. Frémont, Ph. Bonnifait, and V. Cherfaoui. Multi-modal object detection and localization for high integrity driving assistance. *Machine Vision and Applications*, 25(3):583–598, 2014.

[101] P. Sabzmeydani and G. Mori. Detecting pedestrians by learning shapelet features. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, Minnesota, June 2007.

[102] W. R. Schwartz, A. Kembhavi, D. Harwood, and L. S. Davis. Human detection using partial least squares analysis. In *IEEE International Conference on Computer Vision*, pages 24–31, Kyoto, Japan, Sept 2009.

[103] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. L. Cun. Pedestrian detection with unsupervised multi-stage feature learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3626–3633, Portland, Oregon, June 2013.

[104] G. L. S. Shackle. *Decision order and time in human affairs*. Cambridge University Press, Second edition, 1969.

[105] G. Shafer. *A mathematical theory of evidence*. Princeton University Press, Princeton, New Jersey, 1976.

[106] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In A. Leonardis, H. Bischof, and A. Pinz, editors, *Computer Vision – ECCV 2006*, volume 3951 of *Lecture Notes in Computer Science*, pages 1–15. Springer Berlin Heidelberg, 2006.

[107] P. Smets. The combination of evidence in the transferable belief model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):447–458, May 1990.

[108] Ph. Smets. Belief functions: the disjunctive rule of combination and the generalized Bayesian theorem. *International Journal of Approximate Reasoning*, 9(1):1–35, 1993.

[109] Ph. Smets and R. Kennes. The transferable belief model. *Artificial Intelligence*, 66(2):191–234, 1994.

[110] F. Suard, A. Rakotomamonjy, A. Bensrhair, and A. Broggi. Pedestrian detection using infrared images and histograms of oriented gradients. In *IEEE Intelligent Vehicles Symposium*, pages 206–212, Tokyo, Japan, 2006.

[111] Z. Sun, G. Bebis, and R. Miller. On-road vehicle detection: a review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(5):694–711, May 2006.

[112] N. Sutton-Charani, S. Destercke, and T. Denœux. Classification trees based on belief functions. In T. Denoeux and M.-H. Masson, editors, *Belief Functions: Theory and Applications*, volume 164 of *Advances in Intelligent and Soft Computing*, pages 77–84. Springer Berlin Heidelberg, 2012.

[113] D. M. J. Tax, M. van Breukelen, R. P. W. Duin, and J. Kittler. Combining multiple classifiers by averaging or by multiplying? *Pattern Recognition*, 33(9):1475–1485, 2000.

[114] A. Tchamova and J. Dezert. On the behavior of dempster's rule of combination and the foundations of dempster-shafer theory. In *Proceedings of the 6th IEEE International Conference on Intelligent Systems*, pages 108–113, Sofia, Bulgaria, 2012.

[115] J. Tighe and S. Lazebnik. Superparsing: Scalable nonparametric image parsing with superpixels. *International Journal of Computer Vision*, 101(2):329–349, Jan. 2013.

[116] J. Tighe, M. Niethammer, and S. Lazebnik. Scene parsing with object instances and occlusion ordering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3748–3755, Columbus, Ohio, June 2014.

[117] S. Tulyakov, S. Jaeger, V. Govindaraju, and D. Doermann. Review of classifier combination methods. In S. Marinai and H. Fujisawa, editors, *Machine Learning in Document Analysis and Recognition*, volume 90 of *Studies in Computational Intelligence*, pages 361–386. Springer Berlin Heidelberg, 2008.

[118] P. A. Viola and M. J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2012.

[119] S. Walk, N. Majer, K. Schindler, and B. Schiele. New features and insights for pedestrian detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1030–1037, San Francisco, California, June 2010.

[120] P. Walley. *Statistical reasoning with imprecise probabilities*, volume 42 of *Monographs on Statistics and Applied Probability*. Chapman and Hall, 1990.

[121] C.-C. Wang, C. Thorpe, and A. Suppe. Ladar-based detection and tracking of moving objects from a ground vehicle at high speeds. In *IEEE Intelligent Vehicles Symposium*, pages 416–421, Columbus, Ohio, June 2003.

[122] C.-C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte. Simultaneous localization, mapping and moving object tracking. *The International Journal of Robotics Research*, 26(9):889–916, 2007.

[123] X. Wang, T. Han, and S. Yan. An HOG-LBP human detector with partial occlusion handling. In *IEEE International Conference on Computer Vision*, pages 32–39, 2009.

[124] A. Wedel, H. Badino, C. Rabe, H. Loose, U. Franke, and D. Cremers. B-spline modeling of road surfaces with an application to free-space estimation. *IEEE Transactions on Intelligent Transportation Systems*, 10(4):572–583, December 2009.

[125] A. Wedel and D. Cremers. *Stereo Scene Flow for 3D Motion Analysis*. Springer, 2011.

[126] M. Werlberger. *Convex Approaches for High Performance Video Processing*. PhD thesis, Institute for Computer Graphics and Vision, Graz University of Technology, Graz, Austria, June 2012.

[127] C. Wojek and B. Schiele. A performance evaluation of single and multi-feature people detection. In *DAGM Symposium Pattern Recognition*, pages 82–91, Munich, Germany, 2008.

[128] C. Wojek, S. Walk, and B. Schiele. Multi-cue onboard pedestrian detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 794–801, Miami, Florida, USA, June 2009.

[129] T.-F. Wu, C.-J. Lin, and R. C. Weng. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5:975–1005, 2004.

[130] X. Xiong, D. Munoz, J. A. Bagnell, and M. Hebert. 3-d scene analysis via sequenced predictions over points and regions. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2609–2616, Shanghai, China, May 2011.

[131] L. Xu, A. Krzyzak, and C. Y. Suan. Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Transactions on Systems, Man and Cybernetics*, 22(3):418–435, May 1992.

[132] R. R. Yager. On the Dempster-shafer framework and new combination rules. *Information Sciences*, 41:93–137, 1987.

[133] J. Yan, X. Zhang, Z. Lei, S. Liao, and S. Z. Li. Robust multi-resolution pedestrian detection in traffic scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3033–3040, Portland, Oregon, 2013.

[134] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Computing Surveys*, 38(4), Dec. 2006.

[135] C. Zach, T. Pock, and H. Bischof. A duality based approach for real-time TV-L1 optical flow. In F. Hamprecht, C. Schnörr, and B. Jähne, editors, *Pattern Recognition*, volume 4713 of *Lecture Notes in Computer Science*, pages 214–223. Springer Berlin Heidelberg, 2007.

[136] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

[137] L. A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 1(1):3–28, 1978.

[138] B. Zadrozny and C. Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *International Conference on Machine Learning*, pages 609–616, Williamstown, Maryland, 2001.

[139] B. Zadrozny and C. Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 694–699, New York, USA, 2002. ACM.

[140] H. Zhao, J. Sha, Y. Zhao, J. Xi, J. Cui, H. Zha, and R. Shibasaki. Detection and tracking of moving objects at intersections using a network of laser scanners. *Intelligent Transportation Systems, IEEE Transactions on*, 13(2):655–670, June 2012.