Evidential grammars: A compositional approach for scene understanding. Application to multimodal street data

Jean-Baptiste Bordes, Franck Davoine, Philippe Xu, Thierry Denoeux

Evidential Grammars: A Compositional Approach For Scene Understanding. Application To Multimodal Street Data

Jean-Baptiste Bordes[a,1,*], Franck Davoine[b], Philippe Xu[b], Thierry Denœux[b]

[a]*Ecole Polytechnique, Université Paris-Saclay, France.*
[b]*Sorbonne universités, Université de technologie de Compiègne, CNRS, Heudiasyc, Centre de recherche Royallieu, CS 60 319, 60 203 Compiègne cedex, France.*

Abstract

Automatic scene understanding from multimodal data is a key task in the design of fully autonomous vehicles. The theory of belief functions has proved effective for fusing information from several sensors at the superpixel level. Here, we propose a novel framework, called evidential grammars, which extends stochastic grammars by replacing probabilities by belief functions. This framework allows us to fuse local information with prior and contextual information, also modeled as belief functions. The use of belief functions in a compositional model is shown to allow for better representation of the uncertainty on the priors and for greater flexibility of the model. The relevance of our approach is demonstrated on multi-modal traffic scene data from the KITTI benchmark suite.

Keywords: Machine learning, Computer vision, Belief functions, Dempster-Shafer theory

1. Introduction

The ability of the human visual system to rapidly analyze a complex visual scene is remarkable. In the recent years, several problems related to the understanding of the visual content of an image have been tackled. While significant progress has been made in the categorization and localization of isolated objects in recent years, automatic understanding of real-world scenes is still considered as highly challenging. There is a consensus in the computer vision community that the so-called *semantic gap* [1] (between a raw image and the underlying visual class) is too wide to be crossed in one single step of inference. Gradually transforming raw pixels into higher level of representations through deep learning methods [2] gives very promising results. Several recent methods [3, 4, 5] enhance state-of-the-art semantic image segmentation on datasets like PASCAL VOC [6] or ImageNet [7] using deep convolutional networks.

*Corresponding author
Email address: jean-baptiste.bordes@polytechnique.edu (Jean-Baptiste Bordes)
[1]This work was done when J.-B. Bordes was with Heudiasyc, CNRS.

Other classical structured model that take into account dependencies or constraints between output variables can tackle the issue of image understanding. They generally proceed by decomposing the scene into objects, then into parts and subparts, down to the pixels. Since such models require expert knowledge, they have been initially applied to specific categories of scenes. However, in [8], the authors present a more ambitious use of *and/or* graph-based visual grammars to model the decomposition of wide varieties of scenes into objects, parts-of-objects and visual words called primitives. This compositional model contains a large amount of human-provided visual knowledge and is augmented with a set of probabilities estimated thanks to an annotated set of training images.

As a contribution to the development of efficient Advanced Driver Assistance Systems (ADAS), we aim in this paper at labeling the visual classes that are present in the neighborhood of an intelligent vehicle containing several types of sensors (LIDAR, cameras, etc.). Indeed, since traffic scenes are highly cluttered, have varying lighting conditions and contain many occlusions, autonomous vehicles need to fuse several sources of information to get a correct understanding of their complex environment.

Xu et al. [9] demonstrated that belief functions (or mass functions) [10] outmatch other state-of-the-art methods for the task of multimodal information fusion on driving scenes. Mass functions are basically extensions of Bayesian probabilities that carry an explicit representation of ignorance and allow efficient fusion of independent sources of information by reasoning on sets of elements rather than on single elements. To illustrate the practical importance of reasoning on subsets, let us take the example of a pedestrian detector processing an image. Assuming this pedestrian detector cannot discriminate female from male pedestrians, the corresponding belief on the pedestrian class should not be distributed among subclasses. A uniform probability distribution would be misleading as it implies that there are as many male pedestrians as female ones, which for some reason may not be true. This better representation of ignorance is very useful to process multimodal data. For instance, a LIDAR is efficient at detecting an obstacle, but it is of little help for discriminating the type of this obstacle.

In [9], Xu et al. perform an oversegmentation of the image (coming from a camera set on the intelligent vehicles) and independent classification modules are supposed to process sensor information. For each segment of the image, mass functions are used to represent the output of each module and are fused using Dempster's combination rule. This method is robust as sensor failure merely results in some mass functions to be vacuous. The system can therefore continue to operate in degraded mode. The method is also flexible: new classes can be added to the discernment space with almost no impact on the system as the evidence can be simply transferred to the refined classes. These advantages are of the highest importance for a real world perception system.

The output of the previous method is an oversegmented image. The mass function that describes the class of each segment is computed from multimodal but purely local information. Indeed, this mass is obtained through the analysis

of the pixels contained in the particular segment. This evidence can be refined through fusion with prior and contextual information, which can be seen as clues to enhance weak detection or prevent false detection. For instance, considering again the example of an obstacle detected by a LIDAR, shape or neighboring information can help discriminate the type of the obstacle. Consequently, we propose a compositional model augmented with mass functions called *evidential grammars*. This model can deal with input data described by mass functions, and fuse them with priors also represented by mass functions. This model thus benefits from the previously mentioned advantages of mass functions over traditional stochastic visual grammars: it allows for better representation of the uncertainty of the priors as compared to probabilities, and it provides more flexibility to add new priors or to refine the discernment space.

1.1. Related Work

Many graphical stochastic models have been proposed in recent years to refine a local labeling and ensure its spatial consistency. In [11], Conditional Random Fields (CRFs) are introduced at several levels of a stochastic hierarchical model in order to integrate features extracted at different scales. Sudderth et al. [12] propose a sophisticated model containing parts that are shared by a set of object categories: for each category, the distribution and layout of these parts are characterized by parameters estimated from an annotated database. Hierarchical models can also incorporate coarser features. Choi et al. [13] achieve impressive results in scene interpretation by using a tree-based model that incorporates global image features, dependencies between object categories as well as outputs of local detectors into a single probabilistic model. In [14], the authors consider a set of *rules* for ranking alternative parsing hypotheses of the scene using the spatial layout of the objects. The relative importance of these rules is estimated from the training data using a structural SVM.

Using rules to decompose a scene into objects and objects into parts is the core of visual grammars. They have been used recently to perform scene parsing [15]. In [16], a novel object detection framework is introduced with a grammar formalism making it possible to decompose an object into parts and subparts, while taking into account placement and appearance. A scoring derivation is then defined and maximized to find the best parsing. However, this model can only detect a single object. In [17], this grammar framework is augmented with a probability distribution and used to localize faces in images. A belief propagation scheme combines bottom-up and top-down contextual information to aggregate evidence for robust understanding of a visual scene. In [18], a grammar-like generative model is introduced to provide a globally consistent interpretation of an image. This model is based on a hierarchy of reusable parts and compositional grouping rules.

In [19], the authors perform automatic image parsing using an *And/Or* graph-based visual grammar. The *Or* nodes correspond to the "kind-of" relationship and provide a semantic organization of the visual classes into more general or more specific concepts. The *And* nodes correspond to the "composed-of" type of relation, which lists the possible components of a visual object. The

graph is augmented with energy-based probabilities to rank the alternative decompositions, so that it is possible to represent deformable objects.

However, three major difficulties clearly stand out to process multimodal traffic scenes:

1. First, none of these methods can deal with input represented by mass functions. The uncertainty of the low-level features and of the intermediate classes are usually represented using probabilities, whereas there is strong evidence that mass functions are essential to process multimodal data [9].
2. The second difficulty concerns the training of the model: capturing compositional visual knowledge using stochastic models requires the introduction of a large number of parameters. These parameters need to be estimated, since we need to know the likelihood of every two parts to be in every possible spatial layout (hinges, borders, surrounds, etc.). The maximum likelihood estimator then consists in counting the occurrence frequency of every pair of objects in every particular layout [8]. The parameters are consequently estimated with a very unbalanced amount of training samples, especially those corresponding to different levels of hierarchy. Prior information should be taken into account.
3. Finally, the flexibility of the model is an important requirement for a fully operational system: the set of classes should evolve without impacting the whole framework. New priors should be added easily.

1.2. Contributions

The core of the method presented in this paper consists in augmenting a visual grammar with belief functions rather than probabilities. Instead of a stochastic grammar, the resulting framework is called *evidential grammar*; its theoretical basis has been explained in [20]. The present paper introduces two essential contributions for applying evidential grammars to real world data: a training method to estimate the parameters of the belief functions of an evidential grammar from a set of occurrence frequencies, and an optimization method to determine the optimal parse graph for a given image with fast belief computation. By exploiting the strengths of this theory, we show that evidential grammars provide an answer to the three major difficulties of stochastic compositional models.

Handling input data represented by mass functions. Dempster-Shafer mass functions extend probabilities by assigning probability masses to sets, called *focal elements*. As will be detailed in Section 2, the combination of two mass functions transfers belief masses to the intersections of the focal elements. As a result, combining an input represented by a mass function with priors represented by probabilities results a probability mass function, since all the belief is transferred to singletons. Consequently, the priors must be expressed as mass functions in order to take full advantage of the representation of imprecision captured by mass functions at the segment level.

Taking into account prior uncertainty. By using results on parameter estimation with belief functions, the ignorance on the value of the parameters can be explicitly captured and taken into account using mass functions. When all parameters are estimated with a high number of training samples, this approach is equivalent to the Bayesian approach. However, in the case of compositional models, we have previously shown that the various parameters are very likely to be estimated with an unbalanced amount of samples. Our approach provides an elegant solution to this problem by formulating the problem of scene interpretation as the fusion of sources of information, which we assume to be independent. Essentially, our approach considers the whole set of spatial relationships in the scene, the elicited or estimated composition rules, and the extracted low-level features as independent pieces of information, which are finally fused in order to reach the most consistent configuration.

Flexibility of the model. Treating scene understanding as a fusion process also leads to much greater flexibility and modularity as compared to traditional stochastic compositional models. Indeed, grammar rules are seen as independent pieces of information that can easily be added or removed. Moreover, the fact that mass functions are defined on sets make it possible to refine or coarsen the considered classes with minor impact on the model. In particular, the parameters do not require to be re-estimated. This is particularly important when dealing with visual systems that have to be upgraded regularly.

Our method takes place after a first step of scene understanding, described in [9] and referred to as "local fusion" (Figure 1). During this stage, several classifiers and detectors independently process the output of a multi-sensor system including a stereo camera and a LIDAR. The left image of the stereo camera is initially oversegmented and the supposed class of each segment is coded by a belief function. This belief function is estimated by transforming the output of each module into a belief function, and by fusing these belief functions using Dempster's rule.

The output mass function of a segment is thus built only from the information pertaining to that segment; it is used as the input of our method. Our method aims at combining the beliefs of neighboring segments using prior knowledge in order to boost them as well as infer more complex objects. As the prior is expressed using mass functions, this second stage of image understanding is considered as a "global" fusion process.

The paper is organized as follows. Section 2 defines mass functions and introduces the most common operations that are required to handle them. In Section 3, the evidential grammar framework will be recalled with a particular emphasis on the differences with the corresponding stochastic grammar framework, and the training of the parameters of the model will be detailed. In Section 4, the algorithm to find the optimal optimization will be presented. Finally, in Section 5, we will demonstrate the viability of our approach using traffic scene images from the KITTI benchmark suite [21].

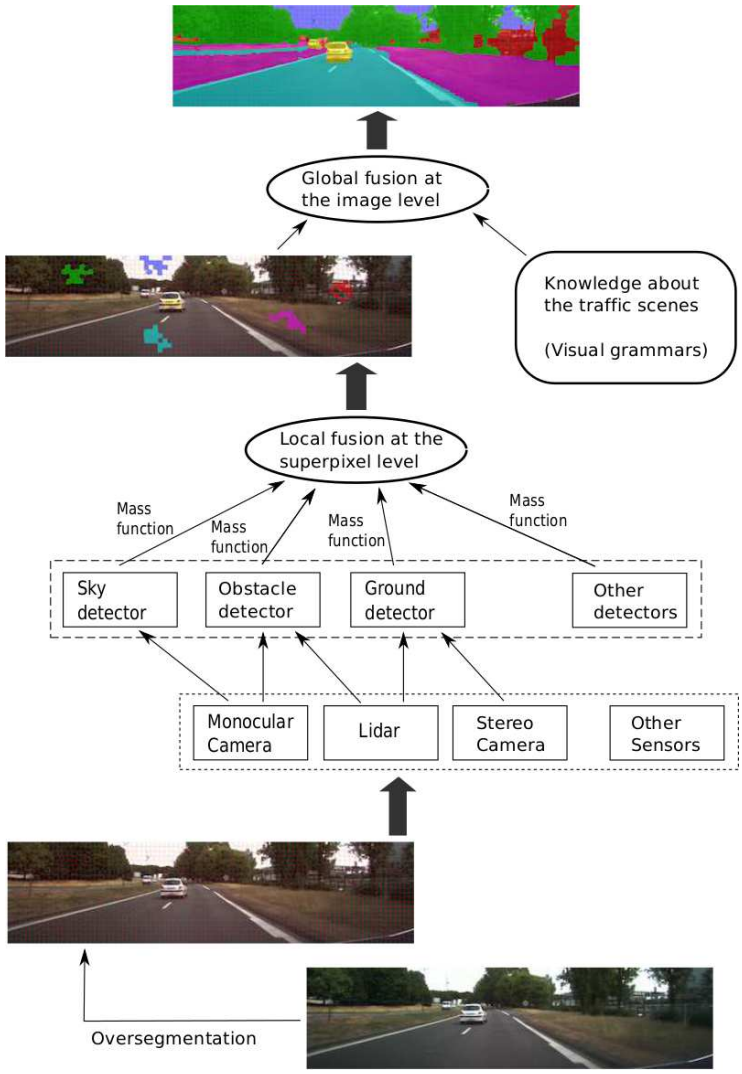


Fig 1: Illustration of the local and global fusion steps. Local fusion is performed at the segment (superpixel) level, combining outputs from different detectors. In a second stage, the masses assigned to each individual segment are combined with compositional rules for consolidation and inference of more complex objects.

2. Theory of belief functions

The Dempster-Shafer theory of belief function [10] generalizes Probability theory by allowing probability masses to be assigned to sets. Formally, let Ω be the set of all the considered classes supposed to be mutually exclusive. A mass function is a function $m : 2^\Omega \rightarrow [0, 1]$ verifying the following normalization property:

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

A particular case is total ignorance, which can be represented by the *vacuous* mass function: $m(\Omega) = 1$. A variable X whose distribution is defined by a mass function is called an *evidential variable*. Let us focus again on the example of a pedestrian detector used previously, and let X be an evidential variable defined on the discernment frame $\Omega = \{woman, man\}$. Its mass function m_X may encode prior information on the ratio of male and female pedestrians at a given location. Complete ignorance would lead to defining m_X as a vacuous mass function. However, if some statistical information shows that the proportion of women in this context is, say, between 30% and 70%, a more committed mass could be used: $m_X(\{woman\}) = 0.3$, $m_X(\{man\}) = 0.3$, $m_X(\Omega) = 0.4$.

2.1. Information fusion using belief functions

Given two mass functions m_1 and m_2 corresponding to two independent sources of information, the conjunctive fusion rule provides a new mass function $m_{1\cap2}$ that combines the information of the previous masses:

$$\forall A \in 2^\Omega,\ m_{1\cap2}(A) = \sum_{B\cap C=A, B\subseteq\Omega, C\subseteq\Omega} m_1(B)m_2(C). \tag{2}$$

This combination rule is commutative and associative. The vacuous mass function is the neutral element and the mass function verifying $m(\emptyset) = 1$ is the absorbing element.

Let us notice that this combination rule generalizes set-intersection, since the mass is transferred to the intersection of the focal elements. Conflict appears when two pieces of information are inconsistent, which is explicitly represented by the mass on the empty set: $m(\emptyset) > 0$ means that the fused pieces of evidence are conflicting. Conflict may have two causes: either a divergence of point of view between experts, or a an incorrect model. In our approach, conflict will be an essential tool to quantify the consistency between a scene model and sensor data.

To prevent the propagation of conflict during the fusion process, Dempster proposed to add a normalization step to the conjunctive fusion to transfer the conflict to the focal elements. The resulting operator, called Dempster's rule of combination is considered as the reference operator to combine belief functions:

$$\forall A \in 2^\Omega,\ A \neq \emptyset, m_{1\oplus2}(A) = \frac{m_{1\cap2}(A)}{1 - m_{1\cap2}(\emptyset)}, (\text{ if } m_{1\cap2}(\emptyset) < 1), \tag{3}$$

and

$$m_{1\oplus 2}(\emptyset) = 0. \tag{4}$$

If there is no conflict between m_1 and m_2, the output mass function is the same as that computed by the conjunctive fusion rule.

Let us emphasize that the fundamental assumption underlying Dempster's rule is the independence and reliability of the sources of evidence. Other fusion operators have been proposed to deal with dependent sources [22]; they can be useful for specific modeling requirements.

2.2. Multivariable reasoning with belief functions

For our present problem, it is necessary to handle belief masses defined on a product space with several evidential variables. In this section, the usual operations required for this purpose are introduced. Two evidential variables X and Y defined, respectively, on the discernment spaces Ω_X and Ω_Y, are considered.

Marginalization. This operation consists in transferring to Ω_X the belief of the joint mass m_{XY} defined on $\Omega_{XY} = \Omega_X \times \Omega_Y$. The resulting mass function is denoted $m_{XY\downarrow X}$:

$$\forall B \subset \Omega_X,\ m_{XY\downarrow X}(B) = \sum_{A\subset\Omega_{XY}\ /\ A\downarrow\Omega_X = B} m_{XY}(A), \tag{5}$$

where $A \downarrow \Omega_X$ stands for the projection of A on the set Ω_X. It can be easily verified that this operation generalizes Bayesian marginalization.

Vacuous Extension. This is the inverse operation of marginalization: it extends a mass function m_X defined on Ω_X to the product space Ω_{XY}. However, there exists actually several mass functions, the marginalization of which provides m_X. To select one of them, the theory of Belief Functions uses the *least commitment principle*, which states that the least informative mass should be chosen. The resulting mass function, denoted by $m_{X\uparrow XY}$, is thus defined by transferring the mass of every focal element of m_X to its cylindrical extension:

$$\forall B \subset \Omega_X,\ m_{X\uparrow XY}(A) = m_X(B),\ A = B \times \Omega_Y. \tag{6}$$

Conditioning. Given a mass function m_{XY} and the assumption that $X \in B, B \subset \Omega_X$, the conditioning operation aims at quantifying the belief about Y. For this purpose, let m_X^B be the mass function defined by $m_X^B(B) = 1$. The vacuous extension is first applied to m_X^B, then combined with m_{XY} and the resulting mass is finally marginalized on Ω_Y. More formally:

$$m_{Y|X\in B} = (m_{X\uparrow XY}^B \cap m_{XY})_{XY\downarrow Y}. \tag{7}$$

If m_{XY} is a Bayesian mass function and p_{XY} the corresponding joint probability distribution, it can be easily verified that the previous operation produces a Bayesian mass function corresponding to $p_{Y|X}$.

Deconditioning. This is the inverse of the marginalization operation; it consists in extending a conditional mass function $m_{Y|X\in B}$ to the product space. The least informative function providing $m_{Y|X\in B}$ when conditioned relatively to B has to be chosen. The resulting mass m_{XY} is thus defined for each subset $A \subset \Omega_Y$ as

$$m_{XY}(C) = \begin{cases} m_{Y|X\in B}(A),\ C = (B \times A) \cup (\overline{B} \times \Omega_Y), \\ 0 \quad \text{otherwise .} \end{cases} \tag{8}$$

Equation (8) can be interpreted as follows: $m_{Y|X\in B}$ carries information on Y when $X \in B$, and the least informative mass is the one which carries complete ignorance when $X \notin B$.

2.3. Decision using mass functions

The plausibility function pl measures the level of plausibility one has for a given evidential variable X. It is computed from mass function m as

$$\forall A \subset \Omega_X,\ pl(A) = \sum_{B\subset\Omega_X | B\cap A\neq\emptyset} m(B). \tag{9}$$

Plausibility is a higher bound of a mass function since $pl(A)$ represents the amount of evidence not contradicting the hypothesis $X \in A$. It is standard to transform masses into plausibility functions in the final step before classification [23].

2.4. Evidential Networks

Evidential networks [24, 25] generalize Bayesian networks by representing graphically the dependencies between mass functions with an hypergraph. An evidential network is composed of a set of evidential variables and a set of edges, each one of them connecting possibly several variables. Variables that are connected by an edge in the hypergraph can have their joint belief function expressed independently from the other evidential variables.

Formally, let U be the set of evidential variables, and $O = \{U_1, \ldots, U_r\}$ a collection of subsets of evidential variables of U. For every set U_i, a joint mass function m_{U_i} is defined on the discernment space of U_i. If $M = \{m_{U_1}, \ldots, m_{U_r}\}$ denotes the set of all these mass functions, the 3-tuple (U, O, M) defines an evidential network.

For instance, let us consider the task of estimating the joint mass function of U, which stands for the set of evidential variables $\{X, X_1, X_2, X_{11}, X_{12}\}$. The computation of m_U requires to combine five mass functions, which might be costly since the number of operations to compute Dempster's rule increases exponentially with the size of the product space. To reduce this complexity, one can use the evidential network represented in Figure 2, which stands for an assumption of independence of the joint belief functions for $U_1 = \{X, X_1, X_2\}$ and $U_2 = \{X_1, X_{11}, X_{12}\}$. This makes it possible to compute m_{U_1} and m_{U_2} first, then the desired mass function $m_u = m_{U_1\uparrow U} \oplus m_{U_2\uparrow U}$. Using this method,

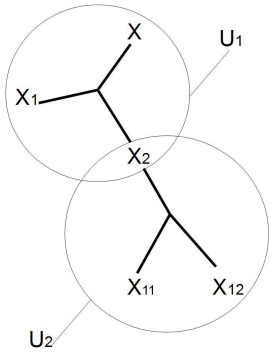


Fig 2: Example of evidential network.

only the final combination is computed on the product space of U, since the computations of m_{U_1} and m_{U_2} are performed on smaller product spaces, which leads to more efficient computation.

3. Evidential Grammars and Image Interpretation

In textual or visual data, some low-level signal elements co-occur frequently. The main idea of grammar-based methods is to group such co-occurring units to form higher-order elements. For instance, *wheels* or *headlights* are likely to be reusable elements in a database of driving scenes since they will be shared by different types of vehicles. Grammars are thus an interesting and powerful framework to perform hierarchical analysis of images.

3.1. Visual grammars

Definition. The modern formalization of grammars can be attributed to Chomsky [26]. A formal grammar G is defined as a 4-tuple $\{S, V_N, V_T, \Gamma\}$ where S is the starting symbol, V_N a finite set of non-terminal symbols, V_T a finite set of terminal symbols, and Γ is a set of production (or derivation) rules rewriting a set of symbols (containing at least one non-terminal symbol) into another set of symbols. Grammars were initially tools to verify the validity of a sequence. They define a generative process which starts with the S symbol and finishes when the resulting sequence, denoted w, is only composed of terminal symbols. However, when switching from textual data to visual data, the natural left-to-right ordering of textual data does not exist, and visual grammars are consequently augmented with a set of spatial relationships denoted Ξ. Many different spatial relations can indeed occur at all levels of vision, like butting, surrounding, adjacent etc.

The whole set of symbols $V_N \cup V_T$ is structured by the production rules Γ. This structure is often represented graphically using nodes as classes and edges

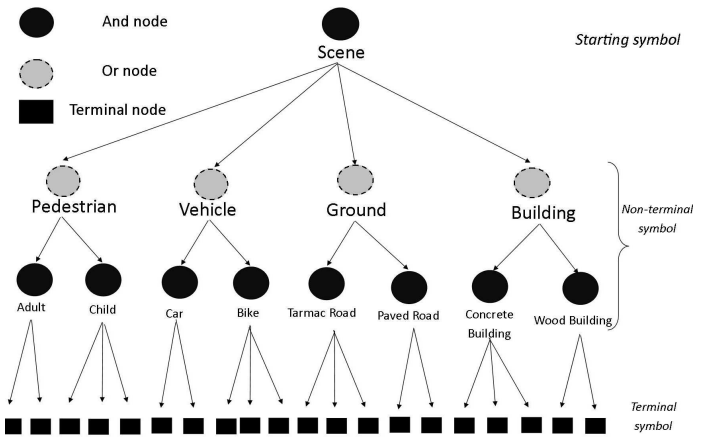


Fig 3: Example of a simple *And-Or* graph for a street scene database

as production rules, as illustrated using a simple example in Figure 3. Zhu et al. [8] showed that *And/Or* graphs are generic structures that can represent almost any type of visual scene. An *And/Or* graph embodies a recursive structure containing two sets of nodes: a set of *And* nodes V^{And} and a set of *Or* nodes V^{Or}:

- An *And* node defines the compositional nature of a node and corresponds to the rule: $A \rightarrow A_1 A_2 \dots A_{n(A)}$, where the number of sub-components of A is denoted by $n(A)$. Some spatial relationships are imposed to enforce spatial consistency in the image decomposition. In the example of Figure 3, a scene is decomposed into vehicles, pedestrians, ground and building. Buildings, vehicles and pedestrians have to stand on the ground, while pedestrians and buildings can occlude each others.

- An "Or" node defines a set of alternative subclasses and corresponds to the rule: $B \rightarrow B_1$ or $B_2 \dots$ or $B_{n(B)}$, where $n(B)$ stands for the number of sub-categories of B. In the example of Figure 3, a pedestrian is modeled as being either an adult or a child.

After the visual words in the whole image have been computed, they are connected in a planar graph of adjacency w. The sequence of rules generating w is called a *parse tree* and will be denoted in this paper $pt(w)$. It can be augmented to a parse graph $pg(w) = (pt(w), E)$, where E stands for the set of relationships between the *And* nodes.

A parse graph can be considered as a hierarchic interpretation of an image such that every pixel is explained by an object category and its parts with spatial relationships between them (cf Figure 4). The set of all the possible sequences of terminals that can be derived from G is called its *language* $L(G)$. The compositional power of grammars comes from the fact that the following inequality

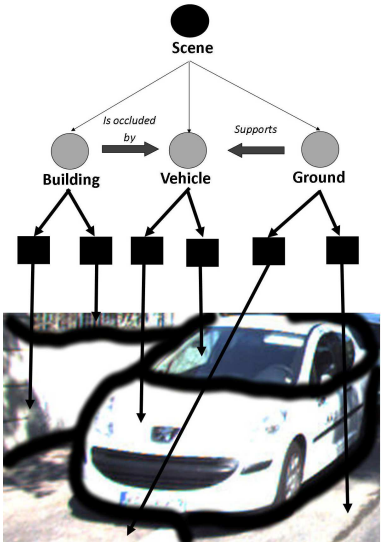


Fig 4: Illustration of a parse graph: sequence of production rules from the starting symbol S to pixels augmented with spatial relationships between child nodes.

is often true in practice: $\mid L(G) \mid >> \mid V_T \mid + \mid V_N \mid$. Indeed, grammars focus on structural compositions rather than listing all the possible configurations.

Stochastic visual grammars. To deal with real-world data, visual grammars include a large number of rules in order to be flexible enough to parse images containing a lot of irregular patterns. Therefore, many valid parse graphs can be derived in practice for a given image. To rank these alternative interpretations, visual grammars are traditionally augmented with a set of probabilities $\mathbb{P}$. For an and/or graph model, $\mathbb{P}$ contains two types of parameters:

1. The probabilities of the spatial pairwise relations between the children of an *And*-node: let A be a non-terminal symbol with the following production rules: $A \rightarrow A_1 A_2 \dots A_{n(A)}$. A probability parameter is assigned to every pair of symbols A_i and A_j to be linked by a spatial relationships e_k conditionally on the father symbol A, denoted $P(e_k \mid A_i, A_j, A)$. For instance, the relationships between a building and a vehicle could be characterized by assessing that the probability for a vehicle to be occluded by a building is 0.1, while the probability for a vehicle to occlude a building is 0.9, all other relationships having zero probability.
2. The probabilities of the alternative productions for the Or-nodes: let B be a non-terminal element with the following production rules: $B \rightarrow B_1 \mid \cdots \mid B_{n(B)}$. Each possible rewriting has a probability $P(\gamma_i) = P(B \rightarrow B_i)$ such that $\sum_{i=1}^{n(B)} P(B \rightarrow B_i) = 1$. The possible rewriting of a vehicle can

be, for instance, ranked as follows: $P(vehicle \rightarrow car) = 0.8$, $P(vehicle \rightarrow bike) = 0.2$.

To estimate the parameters, the maximum likelihood estimator is used. On an annotated database, it consists in the computation of relative frequencies of occurrences of every relation and alternative rewriting:

$$P(B \rightarrow B_i) = \frac{\#(B \rightarrow B_i)}{\sum_{j=1}^{n(B)} \#(B \rightarrow B_j)}, \tag{10}$$

where $\#(B \rightarrow B_i)$ is the number of times that the node B is derived in B_i in the training set. Similarly,

$$P(e_k \mid A_i, A_j, A) = \frac{\#(e_k(A_i, A_j, A))}{\sum_{l=1}^{n(e)} \#(e_l(A_i, A_j, A))}, \tag{11}$$

where $\#(e_l(A_i, A_j, A))$ is the number of times the nodes A_i and A_j are linked with the relation e_l and are children of A, and $n(e)$ is the number of possible configurations between A_i and A_j.

3.2. Evidential grammars

Evidential grammars augment visual grammars with a set of mass functions $\mathbb{M}$ instead of a set of probabilities [20]. An evidential grammar is thus defined by a 6-tuple $\{V_N, V_T, \Gamma, S, \mathbb{M}, \Xi\}$ with every production rule associated to a conditional mass function. Since a mass function is the generalization of a Bayesian probability distribution, evidential grammars generalize stochastic grammars. Indeed, if all the mass functions included in $\mathbb{M}$ are defined only on singletons, the evidential grammar boils down to a stochastic grammar. An advantage of the representation of rules by conditional mass functions is greater flexibility. In particular, refining a class needs no reestimation of the parameters since the belief can be transferred to the union of the refined classes. Moreover, rules can be easily updated by combining estimates and expert assessments by Dempster's rule.

3.2.1. Image model using evidential grammars

As a preliminary step to image interpretation, the image is oversegmented in a partition of small regions containing approximately 250 pixels called segments.

Parse trees are used to decompose a scene down to the segments and we constrain a parse tree to embody a recursive *And/Or* structure. Set V_N is thus split in two subsets V^{Or} and V^{And}. Each *And*-node is linked to a set of *Or*-nodes using a composition relationship and every *Or*-node is linked to an *And*-node using a specification link. Each node covers an area of the image and is augmented with an evidential variable that describes the class contained inside the region. To simplify the vocabulary, we will refer indifferently to the node, the region it covers, or its class variable. An evidential variable is also introduced to describe the shape of any node except the terminal ones. Evidential variables

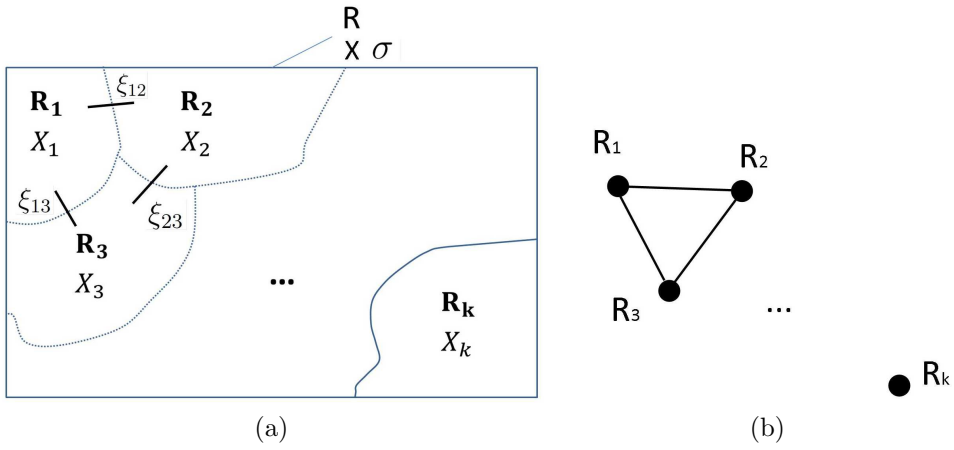


Fig 5: (a) Decomposition of a region into subregions and the corresponding spatial relationships between adjacent regions (b) Representation by a planar graph.

are also used to model the spatial relationships between every pair of adjacent nodes related to the same parent node.

Three domains are defined for the evidential variables: one for the values of the object classes, one for the spatial relationships, and one for the region shapes. Moreover, the edges of the parse tree define the structure of an evidential network structuring the variables. Let U be the set of evidential variables describing the hierarchy (classes of the regions, relationships, shapes), and $O = \{U_1, \ldots, U_r\}$ a set of subsets of variables in U linked in the parse tree by a specification or a composition relationship. Every set U_i is associated to a joint mass function m_{U_i}. If $M = \{m_{U_1}, \ldots, m_{U_r}\}$, the 3-tuple (U, O, M) is the evidential network built from a parse graph.

3.2.2. Production rules

The production rules in Γ are formalized as conditional mass functions providing prior information on the joint mass functions m_{U_i} of the evidential network. Different priors are considered depending on the type of the parent node.

Specification link. Let R be a region whose class is described by two evidential variables X and Y. Every class $\omega \in V^{Or}$ is characterized by the specification mass $m_{Y|X=\omega}$. This conditional mass function is then deconditioned to get a mass m^{ω} in a two-dimensional product space. For instance, the rewriting of a pedestrian could be:

$$m_{Y|X=pedestrian}(\{adult\}) = 0.3,$$
$$m_{Y|X=pedestrian}(\{child\}) = 0.1,$$
$$m_{Y|X=pedestrian}(\{adult, child\}) = 0.6.$$

Composition link. Let R be a region of an image and X an evidential variable describing the class contained in this region. Region R is partitioned into k regions R_1, R_2, ..., R_k and the class contained in each region R_i is defined by the evidential variable Y_i. The evidential variables describing the spatial relationship between every pair of adjacent regions are denoted by ξ_i. The set of regions and their spatial relationships can be represented by a planar graph as illustrated in Figure 5. Another variable for the shape description is denoted by σ.

A class $\omega \in V^{And}$ is characterized by three types of conditional mass functions:

1. A compositional mass function $m_{Y_1,\ldots,Y_k|X=\omega}$, describing the classes of k subcomponents of an object of class ω. This conditional mass function is then deconditioned to get a mass m^{comp} in a $k+1$ dimensional product space. For instance, the decomposition of a simple traffic scene can be written with a categorical mass function:

$$m_{Y_1,Y_2,Y_3|X=Scene}(\{pedestrian\} \times \{ground\} \times \{vehicle\}) = 1.$$

2. A spatial mass function $m_{\xi|Y_i=\omega_i,Y_j=\omega_j,X=\omega}$ describing the possible relationships between two adjacent subcomponents of respective classes ω_i and ω_j of a given object of class ω. This conditional mass function is then deconditioned to get a mass m^{spat} in a 4-dimensional product set. For instance, the spatial relationship between a vehicle and the ground in a scene could be written as

$$m_{\xi|Y_1=vehicle,Y_j=ground,X=scene}(\{over\}) = 1.$$

3. A shape mass function $m_{\sigma|X=\omega}$ describing the shape appearance of region R. This conditional mass function is then deconditioned to get a mass m^{shape} in a two-dimensional product set. For instance, the compactness of a vehicle could be modeled as

$$m_{\sigma|X=vehicle}([0.6, 0.9]) = 1.$$

3.3. Training of evidential grammars

3.3.1. Dempster's model

Let us consider a Bernoulli trial with success probability $p \in [0, 1]$ and N_1 the number of successful samples on N experiments. The maximum likelihood estimate of p is $\widehat{p} = N_1/N$. Dempster's model [27] is quite similar to the Laplace estimator; it defines the belief one can have on the chance of success from the

N experiments using the following mass function,

$$m_D(\{1\}) = \frac{N_1}{N+1},$$
$$m_D(\{0\}) = \frac{N - N_1}{N+1},$$
$$m_D(\{0,1\}) = \frac{1}{N+1}.$$

The belief on $\{0,1\}$ accounts for the uncertainty of the estimator and tends to 0 when the number N of training samples increases; the ratio $m_D(\{1\})/\hat{p}$ thus converges asymptotically to 1. In the particular case where $N = 0$, no training sample is available and there is a belief 1 on $\{0,1\}$. Mass function m_D is then vacuous, which reflects complete ignorance on the value of p.

If a set of $n > 2$ elements is considered, Dempster's model can be applied in a one against all settings to get n estimators that are then fused using Dempster's rule. This method provides better result than probabilistic fusion, especially when the estimators have been trained with an unbalanced number of samples [28].

3.3.2. Estimation of evidential grammar parameters

The mass function m^{comp} is supposed to be provided by experts during the definition of the non-terminal nodes. For the three other types of production rules, these mass functions are estimated with Dempster's model from a set of parse graphs provided with the training dataset.

1. Specification mass function: For every node ω in V^{Or}, the specification mass function $m_{Y|X=\omega}$ is estimated using Dempster's model and then deconditioned to obtain a mass m^{ω}. The final specification mass function m^{Or} can then be computed using Dempster's rule:

$$m^{Or} = \bigoplus_{\omega \in V^{or}} m^{\omega}. \tag{12}$$

2. Spatial mass function: For every 3-tuple $(\omega_i, \omega_j, \omega) \in V^{Or} \times V^{Or} \times V^{And}$, the spatial mass function $m_{\xi|Y_i=\omega_i, Y_j=\omega_j, X=\omega}$ is estimated using the Dempster model and then deconditioned to obtain a mass $m^{\omega_i,\omega_j,\omega}$. The final specification mass function m^{spat} can then be computed using Dempster's rule:

$$m^{spat} = \bigoplus_{(\omega_i,\omega_j,\omega) \in V_{Or} \times V_{Or} \times V_{And}} m^{\omega_i,\omega_j,\omega}. \tag{13}$$

3. Shape mass function: the shape variable is supposed to be defined on a discrete collection; as for the specification mass function, a conditional mass function $m_{\sigma|X=A}$ is estimated for every And-node A. These mass functions are deconditioned and combined to get the final mass m^{shape}.

Example: The principle for parameter estimation is similar for these three kinds of rules. Consequently, we only give instances of spatial mass function

estimation for the example of Section 3.1. The first conditional mass function is written using Dempster's model applied to the Bernoulli trial "pedestrian occludes a building" against all the other spatial relationships. As this trial contains three success and one failure, the mass function is defined as

$$\begin{aligned} m^1_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\{occludes\}) &= 0.6, \\ m^1_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\Xi \setminus \{occludes\}) &= 0.2, \\ m^1_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\Xi) &= 0.2. \end{aligned}$$

The second conditional mass can be built from the Bernoulli trial "pedestrian adjacent to a building", which includes one success and three failures:

$$\begin{aligned} m^2_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\{adjacent\}) &= 0.2, \\ m^2_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\Xi \setminus \{adjacent\}) &= 0.6, \\ m^2_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\Xi) &= 0.2. \end{aligned}$$

All the information incoming from the data for the relationship between these two entities is contained in these two mass functions. The resulting conditional mass function associated to the knowledge is obtained by combining them with Dempster's rule. We finally obtain

$$\begin{aligned} m_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\{occludes\}) &= 0.545, \\ m_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\{adjacent\}) &= 0.093, \\ m_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\Xi \setminus \{occludes\}) &= 0.045, \\ m_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\Xi \setminus \{adjacent\}) &= 0.136, \\ m_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\Xi \setminus \{adjacent, occluded\}) &= 0.136, \\ m_{\xi|X_1=pedestrian,X_2=building,X=Scene}(\Xi) &= 0.045. \end{aligned}$$

Because of the small number of training samples, the conditional mass function above is highly uncertain. When the sample size increases, the mass function converges asymptotically to the probabilistic estimator.

4. Scene interpretation using evidential grammars

A parse graph represents an interpretation of an image that can be derived from a grammar model. Many possible parse graphs are possible in practice for a given image and the goal of the inference task is to find the optimal parse graph according to some relevance criterion. In [20], the criterion proposed is the minimization of the conflict of the root node, since this particular node aggregates the overall conflict of the evidential network. Indeed, it was shown that conflict appears in the evidential network when the structure of the parse tree is not consistent with the observed image primitives and the grammar rules.

The belief masses of the leaf nodes are assumed to be known from sensor information. To evaluate a given parse tree, belief has to be propagated from

the leaves to the root in a bottom-up process following the grammar rules.

4.1. Belief Propagation

Let $\{U, O, M\}$ be an evidential network. By definition, every element $U_i \in O$ contains a parent node and its children. The proposed bottom-up inference process consists in recursively evaluating the mass function of the parent node by fusing the masses of the children with the prior mass of the grammar rules.

4.1.1. Bottom-up propagation

Inference of an Or-node. Let $U_i = \{X, Y\}$ be a set of variables containing an Or-node X and its child node Y. Mass function m_Y is assumed to be known and m_X has to be inferred. The vacuous extension is first applied to m_Y to obtain a joint mass m_{XY}, which is then combined to the specification mass,

$$m_{U_i} = (m^{Or} \cap m_{Y\uparrow U_i}). \tag{14}$$

Mass function m_X is finally evaluated by marginalization of the joint mass: $m_X = m_{U_i\downarrow X}$.

Inference of an And-node. Let us consider a set of variables

$$U_i = \{X, X_1, \ldots, X_l, \xi_1, \ldots, \xi_r, \sigma\}.$$

Variables X_j, ξ_k and σ are assumed to be known, respectively, through the masses m_{X_j}, m_{ξ_k}, and m_σ and the parent node X has to be inferred. Let $t_1(j)$ and $t_2(j)$ stand for the indices of the two adjacent regions linked by the spatial relationship ξ_j. We denote by m_j^{spat} the spatial mass function defined on the discernment frame $\{\xi_j, X_{t_1(j)}, X_{t_2(j)}, X\}$. The mass function of the variables and the grammar rules are extended to U_i and then marginalized:

$$m_{U_i} = \cap_{j=1}^{r} m_{\xi_j\uparrow U_i} \cap_{k=1}^{r} m_k^{spat} \cap_{p=1}^{l} m_{X_p\uparrow U_i} \cap m_{\uparrow U_i}^{comp} \cap m_{\uparrow U_i}^{shape}. \tag{15}$$

Mass function m_X is finally computed by marginalizing the joint mass fnuction: $m_X = m_{U_i\downarrow X}$.

4.1.2. Top-down propagation

After performing the bottom-up propagation, any joint mass m_{U_i} can be marginalized on a particular variable $X_j \in U_i$ to obtain the belief on its class, which only takes into account the information lying inside the region described by the discernment frame U_i. However, the context of the whole image can be exploited by first fusing the mass functions m_{U_i} on the discernment frame U before marginalization:

$$m_{X_j} = (\cup_{i=1}^{n} m_{U_i\uparrow U})_{\downarrow X_j}.$$

The same process is also applied to compute the evaluation criterion $m_{X_s}(\emptyset)$. Once the optimal parse tree has been computed, the maximum plausibility

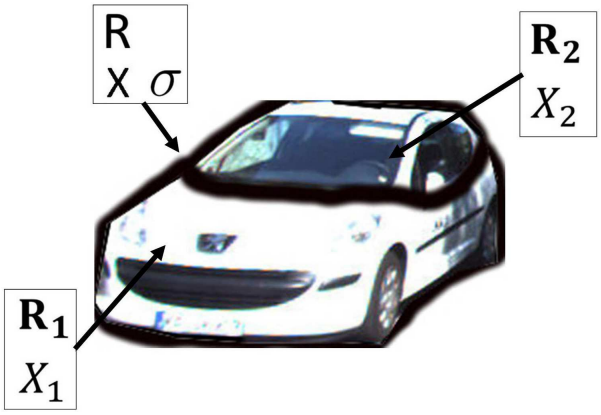


Fig 6: A region R, with class and shape described by evidential variables X and σ. Region R is partitioned into two regions R_1 and R_2 whose classes are described, respectively, by X_1 and X_2.

criterion is used to make a decision on the class lying in every region of the parse tree.

4.2. An example of belief propagation

We illustrate the belief propagation process on the simple example shown in Figure 6. The goal is to infer knowledge on the value of the Y variable describing the class of the region R, and to propagate it back on the value of X_1 and X_2. The whole set of variables is $U_1 = \{\xi, X_1, X_2, X, \sigma\}$.

Let us define the rules that will be used for that purpose. For the sake of clarity of the notations, only two classes will be considered: *pedestrian* (denoted by P) and *vehicle* (denoted by V). These classes are formed by grouping a set of *parts-of pedestrians* (denoted by POP) and *parts-of vehicle* (denoted by POV), respectively. The latter information is written using the compositional mass function:

$$m^{comp}_{X_1,X_2,X}(POP \times POP \times P \cup POV \times POV \times V) = 1.$$

Let us state that for the parts to become a whole single object, the corresponding segments have to be adjacent:

$$m^{spat}_{\xi,X_1,X_2,X}(\{adjacent\} \times POP \times POP \times P \cup \{adjacent\} \times POV \times POV \times V) = 1.$$

Let us assume finally that a confidence interval on the value of compactness has been evaluated from a training dataset:

$$m^{shape}_{X,\sigma}(P \times [0.3, 0.5] \cup V \times [0.7, 1]) = 1.$$

Then, the values of X_1, X_2 and ξ are used as input to the belief propagation.

First, the spatial relationship ξ is assumed to be adjacent: $m_\xi(\{adjacent\}) = 1$. Second, the values of the classes lying inside R_1 and R_2 are assumed to come from the output of a weak detection prior to this step (fusion of vehicle and pedestrian detectors for example):

$$\begin{aligned} m_{X_1}(\{POP\}) &= 0.4, \\ m_{X_1}(\{\Omega\}) &= 0.6, \\ m_{X_2}(\{POP, POV\}) &= 0.5, \\ m_{X_2}(\{\Omega\}) &= 0.5, \\ m_\sigma(0.8) &= 1. \end{aligned}$$

4.2.1. Bottom-up propagation

The vacuous extension is performed on U_1 to the seven previous mass functions:

$$\begin{aligned} m^{comp}_{X_1,X_2,X\uparrow U_1}(&\Omega \times POP \times POP \times P \times [0,1] \cup \\ &\Omega \times POV \times POV \times V \times [0,1]) = 1, \end{aligned}$$

$$\begin{aligned} m^{spat}_{\xi,X_1,X_2,X\uparrow U_1}(&\{adjacent\} \times POP \times POP \times P \times [0,1] \cup \\ &\{adjacent\} \times POV \times POV \times V \times [0,1]) = 1, \end{aligned}$$

$$\begin{aligned} &m^{shape}_{\sigma,X\uparrow U_1}(\Omega \times \Omega \times \Omega \times P \times [0.3, 0.5] \cup \Omega \times \Omega \times \Omega \times V \times [0.7, 1]) = 1, \\ &m_{\xi\uparrow U_1}(\{adjacent\} \times \Omega \times \Omega \times \Omega \times [0,1]) = 1, \\ &m_{X_1\uparrow U_1}(\Omega \times \{POP\} \times \Omega \times \Omega \times [0,1]) = 0.4, \\ &m_{X_1\uparrow U_1}(\Omega \times \Omega \times \Omega \times \Omega \times [0,1]) = 0.6, \\ &m_{X_2\uparrow U_1}(\Omega \times \Omega \times \{POP, POV\} \times \Omega \times [0,1]) = 0.5, \\ &m_{X_2\uparrow U_1}(\Omega \times \Omega \times \Omega \times \Omega \times [0,1]) = 0.5, \\ &m_{\sigma\uparrow U_1}(\Omega \times \Omega \times \{P, V\} \times \Omega \times 0.8) = 1. \end{aligned}$$

By combining all these mass functions with the conjunctive rule, we obtain

$$m_{U_1}(\emptyset) = 0.4,\ m_{U_1}(\{adjacent\} \times POV \times POV \times V \times 0.8) = 0.6.$$

Marginalizing on X the previous joint mass function yields

$$m_X(\emptyset) = 0,4,\ m_X(V) = 0.6.$$

This result means that, according to the rule and the input information, the class lying in R is a vehicle. The mass 0.4 on the empty set is due to conflict between the input information and the rules, which results from a false detection of pedestrian in R_1.

4.2.2. Top-down propagation

To propagate the knowledge obtained by the fusion of the rules and the input mass functions back to the segments, the mass m_{U_1} is first computed as in the previous step but using Dempster's rule (to remove the mass on the empty set):

$$m_{U_1}(\{adjacent\} \times POV \times POV \times V \times 0.8) = 1. \tag{16}$$

This mass function is then marginalized on X_1 and X_2 to get

$$m_{X_1}(POV) = 1,\ m_{X_2}(POV) = 1.$$

These mass functions are very different from the original ones since a lot of contextual information has been exploited during the bottom-up and top-down propagation.

4.3. Optimization process using MCMC

The goal of the optimization process is to find the parse graph pg minimizing the conflict of the root node $C(pg)$. Let Σ be the set of all the valid parse graphs whose leaf nodes are the terminal symbols of a test image. This set has an enormous number of local maxima and we propose to explore Σ with a reversible jump Markov Chain Monte Carlo (MCMC) algorithm [29]. For this purpose, the solution space is modeled by a Markov chain structure: every valid hierarchy defines a state of the chain and transitions are defined by considering three dynamics:

1. Merging: two nodes are merged in one single node, the children of which are the union of the children of the initial nodes.
2. Splitting: the children of a node (containing more than one child node) are split in two nodes.
3. Boundary competition: this type of modification between two adjacent nodes changes the attribution of their children located besides their common boundary.

The initial state is defined by creating an object for each superpixel: all the terminal nodes are linked directly to the root node. The Metropolis-Hastings (MH) algorithm is then used with the following two probability distribution: the stationary distribution π defined by the Boltzman energy distribution $\pi(H) = \frac{\exp(-C(pg))}{Z(\Sigma)}$, where $Z(\Sigma)$ is the partition function, and the transition probabilities. For the latter, similar transition probabilities as in [30] are chosen for the three dynamics mentioned above, as the Data Driven MCMC scheme yielded successful results for low-level segmentation as well as for grammar-based image parsing. For faster convergence, the probabilities to choose each of the three dynamics have been modified to encourage the algorithm to choose fusion with higher probability when the number of regions is high.

The exploration is finally performed using simulated annealing. At every step of the algorithm, a neighboring state is chosen and the conflict of the root node of this state is computed. At every iteration of the MCMC algorithm, the

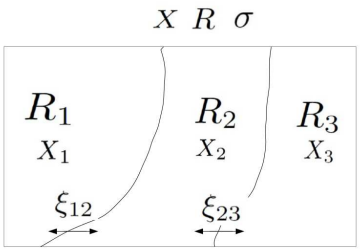


Fig 7: Example for fast computation of the root conflict. R stands for a region that is decomposed into three sub-regions R_1, R_2 and R_3. Evidential variables X and σ describe respectively the class and shape of region R. Evidential variables X_1, X_2, X_3 describe the classes that lie respectively in R_1, R_2 and R_3, while ξ_{12} and ξ_{23} contain the belief one has on the nature of the relationship between respectively regions R_1 and R_2, and between regions R_2 and R_3.

conflict of the root node has to be estimated. For this purpose, it is necessary to evaluate $(m_U)_{\downarrow X_S}$. This operation is the bottleneck of the algorithm and is crucial for the practical use of evidential grammars.

4.4. Computational complexity analysis

Let N be the total number of nodes in the hierarchy. The belief on each region is described by an evidential variable as well as all the pairwise relationships between adjacent regions and the shape of every non-terminal node (except the root node). Since all the regions are assumed to be connex, representing every region by a node and an adjacency relationship by an edge linking two nodes leads to a planar graph. Using Euler's formula for planar graphs, the number of edges is at most 3 times the number of nodes. Consequently, since two evidential variables are introduced for every node (one for the shape, one for its class) and one for every spatial relationship, the number of evidential variables in the network is bounded by $5N$.

For a domain of size n (which is considered here to be the number of object classes), combining two mass functions with Dempster's rule has a worst-case complexity $O(2^{2n})$. Computing Dempster's rule on a product space increases exponentially the complexity, and the double exponentiality of the direct combination-marginalisation proposed in Equation (15) makes this process intractable.

However, the *fusion algorithm* [31] allows the exact computation of the marginal on a subset of variables without evaluating the joint mass function on the whole discernment frame. It is a much more efficient alternative to the straightforward computation mentioned above. The fundamental operation is to delete iteratively the variables from the network. Finding the optimal elimination sequence is a NP-hard problem, but several methods have been proposed to find a good elimination sequence [32].

Let us consider the example illustrated in Figure 7 with discernment space $U^1 = \{X, X_1, X_2, X_3, \xi_{12}, \xi_{23}, \sigma\}$. Instead of performing the bottom-up process of inference of the mass function m_X presented in Section 4.1.1 by combining all the mass functions on the discernment space U^1, it is possible to reduce the computational load by performing the combination on local domains followed by elimination of variables. First, spatial relation ξ_{12} can be eliminated by combining the two only pieces of evidence for this variable, which are $m_{\xi_{12}\uparrow\{\xi_{12},X_1,X_2,X\}}$ and the prior $m^{spat}_{\xi_{12},X_1,X_2,X}$, and then marginalizing the obtained mass function on $\{X_1, X_2, X\}$. We will denote this intermediate mass function as $m^{inter}_{X_1,X_2,X}$. The same reasoning can be applied to eliminate the spatial relation ξ_{23}. The class variables corresponding to the regions with minimal number of neighbors are then eliminated iteratively. Variable X_1 can thus be eliminated by first combining $m_{X_1\uparrow\{X_1,X_2,X\}}$ with the two other pieces of evidence describing X_1, namely, $m^{inter}_{X_1,X_2,X}$ and the prior $m^{comp}_{X_1,X_2,X}$, and then marginalizing the obtained mass function on $\{X_2, X\}$. The previous combination has been performed on a discernment space of three variables. Eliminating variable X_2 first would have been less efficient, since the region R_2 has two neighboring regions and the required combination has to be performed on $\{X_1, X_2, X_3, X\}$. After the class variables have all been eliminated, the remaining mass function $m_{X,\sigma}$ is combined with the shape prior $m^{shape}_{X,\Sigma}$ and marginalized to obtain the overall class m_X.

More generally, the inference scheme consists in first eliminating the spatial mass functions, then eliminating iteratively the class mass functions corresponding to the regions with minimal number of neighbors, and finally the shape mass functions. The bottleneck operation is the elimination of the class mass functions since the combination operation is performed on a discernment space of size $2 +$ number of neighbours. A theorem from planar graph theory states that it is possible to find a region with less than five neighbors [33]. The overall computational complexity for the exact inference is thus lower than $4N2^{7n}$.

Despite being considerably reduced, the complexity remains too high for exact inference. However, Bauer [34] showed that for some applications, efficient decision can be performed with belief functions by removing focal elements and redistributing the corresponding masses to a reduced number of focal elements. We actually observe that, in practice, the handled mass functions contain only a small number of focal sets containing a significant belief mass. We can thus constrain the mass functions to a fix amount f of focal sets using the *D1 algorithm* presented in [34], which concentrates the mass on the focal elements that are strongly supported by the original evidence. We verified that, for 16 object classes and $f = 15$, the impact on the performances was negligible while the computational load was considerably reduced. The upper bound on the complexity is thus reduced to $4Nf^7$. In practice, the complexity is much lower: when a local modification (split, merge, competition) is applied to a node, only some of the information has to be updated because only the newly created nodes and their parents are impacted.

5. Experiments

5.1. Architecture of the system

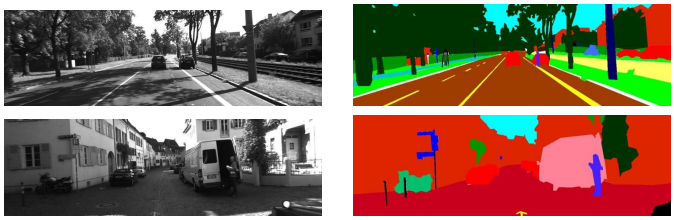
The system we consider is composed of several sensors observing an urban scene, including stereo cameras used to produce an oversegmented image and the disparity map. The average size of the segments is chosen to be large enough so that relevant features can be computed from their pixels, and small enough so that a same segment does not contain several classes. The sensors provide data to a set of classifiers running totally or partially in parallel. For each segment, belief functions are provided by each sensor and are fused using Dempster's rule. The pieces of information provided by the classifiers are supposed to be independent. This assumption makes sense in a multimodal framework, since different sensors provide different types of information. Xu et al. demonstrated in [9] that this fusion approach has significant advantages over other fusion methods. Indeed, reasoning on a subset allows us to fuse the information of classifiers reasoning in different discernment frames (for instance, a pedestrian detector and a ground detector) and to refine the classes without any impact on the system. Moreover, it provides a better representation of the uncertainty of the different classifiers relatively to their performance. The belief functions associated to each segment of the image will be used as the input data of our evidential grammar.

5.2. Database

To our knowledge, no multimodal database is annotated at the pixel level. We annotated manually 300 images of size 1242×375 from the KITTI benchmark suite (Figure 8) using 14 labels. This database contains four different types of street scenes: campus, residential area, city, road. The data were collected from a Velodyne LIDAR, a GPS localization system, and two high-resolution color and grayscale cameras. The left grayscale image was chosen for manual annotation; 140 images were annotated at pixel level, and 160 at superpixel level. One hundred and sixty images were also annotated with object identification for the training of the production rules.

5.3. Benchmark test

The efficiency of evidential grammars was compared with that of two other methods. First, we studied the performance improvement induced by the use of evidential framework by comparing our approach with traditional stochastic grammars. Second, we performed comparison with a state-of-the-art Markov Random Field-based framework, like the one implemented in the Automatic Labelling Environment (ALE) [35]. Cross-validation was performed by splitting the database into three parts while keeping a similar ratio of the different types of scenes.



(a) (b)

Fig 8: Ground truth. Top: Residential area, Bottom: City area. Data available at https://www.hds.utc.fr/∼xuphilip/dokuwiki/en/data

.

5.3.1. Grammar model for street scenes

Street scenes were modeled using an *And-Or* graph composed of four levels of hierarchy. At the top lies the root node S (Scene), which is decomposed into 14 *And*-nodes corresponding to the object-level classes that will be used for the method evaluation. They were decomposed into a set of *Or*-nodes representing more specific categories or different points of view (car from the side, car from the front, etc.). These *Or*-nodes were then decomposed into the terminal symbols of the grammar; they describe the class lying in a segment of the image. Three spatial relationships, based on 3D distance, topological relation and orientation information, were considered to characterize pairwise object-level relationship. For segment-level nodes, only the 3D distance was considered since the average size of segments (100 pixels) makes more sophisticated relationships too little informative. The shape of the object-level nodes was also characterized using size and compactness parameters. However, these parameters were only estimated for the *things* type of objects. For *stuff* types of objects, the shape mass function was defined as a vacuous mass function. A total of 264 mass functions related to the grammar information were estimated during the training stage.

5.3.2. Implementation and parameter settings

In a first step, the left grayscale image was oversegmented into 9000 segments using the SLIC algorithm [36]. The disparity map was then computed from the two grayscale images and used to evaluate the distance between segments. A mass function was used to describe the class lying in each segment. For this purpose, the following modules were used to provide mass functions for each segment:

- Ground detection module based on plane detection in the disparity map [9];
- Ground detection module based on LIDAR information and the disparity map [9];
- Ground and sky detection module proposed by [37].

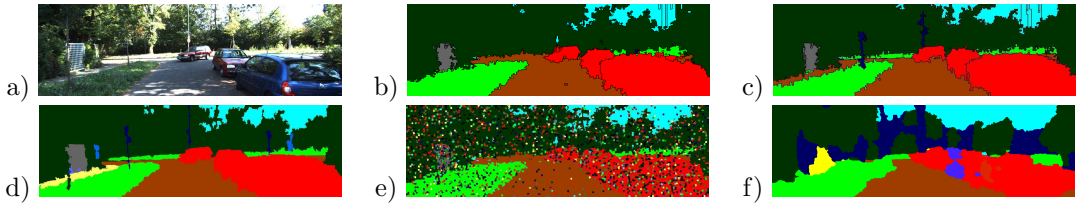


Fig 9: (a) Raw image, (b) Stochastic grammar annotation, (c) Evidential grammar annotation, (d) Ground truth, (e) Local annotation, (f) ALE annotation.

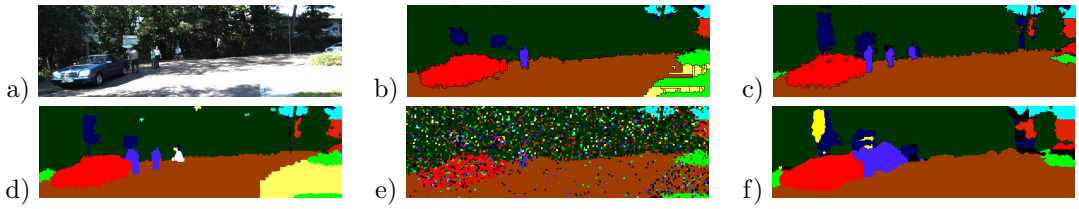


Fig 10: (a) Raw image, (b) Stochastic grammar annotation, (c) Evidential grammar annotation, (d) Ground truth, (e) Local annotation, (f) ALE annotation.

Three independent multiclass detection modules were also added, based on Local Binary Patterns (LBP) [38] and Gabor and Scale Invariant Features (CSIFT) [39]. These features were first extracted densely in the database and then quantized. For every segment of the image an histogram was then computed and a SVM were trained in a one-against-all setting. The SVM score was converted into a belief function using the method described in [40].

As 14 classes were considered at each level of the hierarchy, an array of size 2^{14} would have been necessary to store the whole information of a single mass. To avoid excessive memory usage, only the belief masses assigned to the focal elements were coded using a list structure. All the nodes of the image were also implemented using a single common C++ class. Objects and segments were realized as instances of this class.

5.4. Results

From Table I, we can see that ALE globally has higher annotation accuracy on *stuff* type of classes. This result is due to the fact that ALE integrates image position information, which improves the annotation accuracy of *stuff* classes, while spatial rules are less informative for classes with no particular spatial extent. However, the image position highly depends on the database, which makes the ALE framework less flexible. For object classes, grammars globally show higher performance.

This result was expected as visual grammars allow object-level description and make use of depth information. Moreover, ALE performs a segmentation

Table 1: Performance of four different methods on KITTI benchmark suite using Pixel-wise percentage accuracy

method	Building	Tree	Sky	Car/Bus	Bike	Traffic sign	Road
ALE	82,1	80,2	96,5	75,1	30,7	38,8	92,2
Local fusion	74,2	68	92,4	66,9	22,3	27,8	95,1
Evid. Gram.	78,7	74,1	94,6	78,3	35,7	45,4	95,6
Stoch. Gram.	78,5	73,9	94,7	75,4	33,9	40	95,4

method	Pedestrian	Fence	Sidewalk	Bicyclist	Railways	Grass	Plants
ALE	31,6	48,4	76,9	27,1	91,3	88,6	65
Local fusion	23,4	39,7	60,8	21,3	85,7	81,4	62,7
Evid. Gram.	35,3	45,8	62,1	28,2	83,1	82,3	64,1
Stoch. Gram.	29,6	46	60,3	23,1	84,4	81,1	64,2

of test images based on the class of the pixels but does not identify the number of object instances. We observe that grammars generally provide more accurate object boundaries than ALE when the depth information is reliable (see Figure 10). However, depth and shape information do not always allow to correctly separate distant objects.

We also see that evidential grammars perform slightly better detection of *thing* classes than do stochastic grammars. The difference is more important for classes with low occurrence in the database like bikes and pedestrians, which evidential grammars detect better. It can be observed in Figures 9 and 10 that weak detection of pedestrians, bikes and poles tend to be pruned by ALE and stochastic grammars. We interpret this effect as the result of careful prior information encoded in the evidential grammar rules, regarding classes with higher occurrence in the database. This observation validates our approach, which consists in taking into account uncertainty in the grammar rules.

However, this slightly better detection accuracy comes with a heavier computational load. After feature extraction, it took ALE 24 seconds on average to annotate one image of our database, against 645 and 1057 seconds for, respectively, stochastic and evidential grammars on an Intel i7-3720QM 2.60 GHz CPU.

6. Conclusion

In this work, we have proposed a compositional framework for scene understanding based on the theory of belief functions. This approach has two strong advantages. First, it provides very flexible knowledge representation, as it allows us to evolve the model by adding compositional rules or refining the set of classes without requiring an additional training step. This is particularly useful with many kinds of embedded systems. Second, we have established the importance of taking into account uncertainty in the production rules used to model the scene composition. Indeed, when the relationships between pairs of objects are estimated with very unbalanced data, belief functions model the uncertainty in

the estimation of the parameters. Evidential grammars thus provide an elegant and efficient solution to the problem of overfitting in compositional models.

The experiments performed on the KITTI benchmark suite prove the efficiency of the evidential grammars framework. However, the system presented in this article is still far from an online system that could be embedded in autonomous vehicles. The main reason is the computational load of this approach, which is higher than that of the state of the art. However, we believe that the running time could be considerably reduced by further optimization in the inference algorithm.

More information can be embedded as well in the visual model: models depending on the scale of objects, more object classes, more types of relationships, and another layer of reusable parts-of-objects (*wheels*, *head*, etc.). Moreover, only deterministic relationships have been considered in this work, but taking into account uncertainty in that aspect can be fruitful. We also plan to integrate object detectors in our framework, which would imply changing the inference process in order to remain consistent with the superpixel independence assumption.

7. Acknowledgements

This work was carried out and funded in the framework of the Labex MS2T, supported by the French Government through the program "Investments for the future" managed by the French National Research Agency (Reference ANR-11-IDEX-0004-02). It was supported by the ANR-NSFC Sino-French PRETIV project (Reference ANR-11-IS03-0001).

References

[1] J. S. Hare, P. H. Lewis, P. G. B. Enser, J. C. Sandom, Mind the gap: another look at the problem of the semantic gap in image retrieval, in: E. Y. Chang, A. Hanjalic, N. Sebe (Eds.), Multimedia Content Analysis, Management, and Retrieval, Vol. 6073, SPIE, San Jose, CA, United States, 2006, pp. 607309–1.

[2] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016, `http://www.deeplearningbook.org`.

[3] E. Shelhamer, J. Long, T. Darrell, Fully convolutional networks for semantic segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (4) (2017) 640–651.

[4] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, Las Vegas, Nevada, USA, 2016, pp. 770–778.

[5] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, Honolulu, Hawaii, USA, 2017.

[6] M. Everingham, S. M. A. Eslami, L. V. Gool, C. K. I. Williams, J. Winn, A. Zisserman, The PASCAL visual object classes challenge: A retrospective, International Journal of Computer Vision 111 (1) (2015) 98–136.

[7] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, Imagenet large scale visual recognition challenge, International Journal of Computer Vision 115 (3) (2015) 211–252.

[8] S.-C. Zhu, D. Mumford, A stochastic grammar of images, Foundations and Trends in Computer Graphics and Vision 2 (4) (2006) 259–362.

[9] P. Xu, F. Davoine, J.-B. Bordes, H. Zhao, T. Denœux, Multimodal information fusion for urban scene understanding, Machine Vision and Applications, Springer 27 (3) (2016) 331–349.

[10] G. Shafer, A mathematical theory of evidence, Princeton University Press, 1976.

[11] L. Ladický, P. Sturgess, K. Alahari, C. Russell, P. H. S. Torr, What, where and how many? combining object detectors and CRFs, in: K. Daniilidis, P. Maragos, N. Paragios (Eds.), European Conference on Computer Vision, Vol. 6314, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2010, pp. 424–437.

[12] E. B. Sudderth, A. Torralba, W. T. Freeman, A. S. Willsky, Learning hierarchical models of scenes, objects, and parts, in: Tenth IEEE International Conference on Computer Vision, Beijing, China, 2005, pp. 1331–1338.

[13] M. J. Choi, A. Torralba, A. S. Willsky, A tree-based context model for object recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (2) (2012) 240–252.

[14] J. Xiao, J. Hays, B. C. Russell, G. Patterson, K. A. Ehinger, A. Torralba, A. Oliva, Basic level scene understanding: categories, attributes and structures, Frontiers in Psychology 4 (2013) 506.

[15] J. Porway, K. Wang, B. Z. Yao, S.-C. Zhu, A hierarchical and contextual model for aerial image parsing, International Journal of Computer Vision 88 (2) (2010) 254–283.

[16] P. F. Felzenszwalb, Object detection grammars, in: IEEE International Conference on Computer Vision Workshops, Barcelona, Spain, 2011, p. 691.

[17] J. Chua, P. F. Felzenszwalb, Scene grammars, factor graphs, and belief propagation, CoRR (2016), http://arxiv.org/abs/1606.01307.

[18] Y. Jin, S. Geman, Context and hierarchy in a probabilistic image model, in: Computer Vision and Pattern Recognition, IEEE Computer Society, New York, NY, United States, 2006, pp. 2145–2152.

[19] Y. Chen, L. L. Zhu, C. Lin, A. Yuille, H. Zhang, Rapid inference on a novel AND/OR graph for object detection, segmentation and parsing, in: J. Platt, D. Koller, Y. Singer, S. Roweis (Eds.), Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver and Whistler, B.C., Canada, 2007, pp. 289–296.

[20] J.-B. Bordes, F. Davoine, P. Xu, T. Denoeux, Evidential grammars for image interpretation - application to multimodal traffic scene understanding.,

in: Integrated Uncertainty in Knowledge Modelling and Decision Making (IUKM 2013), Z. Qin and V. N. Huyn (Eds), LNAI 8032, Beijing, China, 2013, pp. 65–78.

[21] A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? the kitti vision benchmark suite, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, Providence, RI, USA, 2012, pp. 3354–3361.

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

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

[24] P. P. Shenoy, A valuation-based language for expert systems, International Journal of Approximate Reasoning 3 (1989) 383–411.

[25] G. Shafer, P. P. Shenoy, K. Mellouli, Propagating belief functions in qualitative Markov trees, International Journal of Approximate Reasoning 1 (1987) 349–400.

[26] N. Chomsky, Syntactic Structures, Mouton: The Hague, 1957.

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

[28] P. Xu, F. Davoine, H. Zha, T. Denoeux., Evidential calibration of binary SVM classifiers, International Journal of Approximate Reasoning 72 (2016) 55–70.

[29] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, E. Teller, Equation of state calculations by fast computing machines, The Journal of Chemical Physics 21 (6) (1953) 1087–1092.

[30] Z. Tu, S.-C. Zhu, Image segmentation by data-driven markov chain monte carlo., IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (5) (2002) 657–673.

[31] R. Haenni, Ordered valuation algebras: a generic framework for approximating inference, International Journal of Approximate Reasoning 37 (2004) 1–41.

[32] R. G. Almond, Graphical belief modeling., Chapman and Hall, 1995.

[33] D. B. West, Introduction to graph theory, Prentice Hall Inc., 2001.

[34] M. Bauer, Approximation algorithms and decision making in the dempster-shafer theory of evidence - an empirical study, International Journal of Approximate Reasoning 17 (1996) 217–237.

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

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

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

[38] L. Nanni, A. Lumini, S. Brahnam, Survey on LBP based texture descriptors for image classification, Expert Systems with Applications 39 (3) (2012) 3634–3641.

[39] A. Abdel-Hakim, A. Farag, CSIFT: A SIFT descriptor with color invariant characteristics, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, New York, NY, USA, 2006, pp. 1978–1983.

[40] D. Dubois, H. Prade, P. Smets, A definition of subjective possibility, International Journal of Approximate Reasoning 48 (2008) 352–364.