A Lightweight Inference Method for Image Classification

John Mark Agosta
johnmark.agosta@gmail.com

Preeti J. Pillai
preetipillai04@gmail.com

Abstract

We demonstrate a two phase classification method, first of individual pixels, then of fixed regions of pixels for scene classification—the task of assigning posteriors that characterize an entire image. This can be realized with a probabilistic graphical model (PGM), without the characteristic segmentation and aggregation tasks characteristic of visual object recognition. Instead the spatial aspects of the reasoning task are determined separately by a segmented partition of the image that is fixed before feature extraction. The partition generates histograms of pixel classifications treated as virtual evidence to the PGM. We implement a sampling method to learn the PGM using virtual evidence. Tests on a provisional dataset show good (+70%) classification accuracy among most all classes.

1 Introduction

Scene recognition is a field of computer understanding for classification of scene types by analysis of a visual image. The techniques employed for scene recognition are well known, relying on methods for image analysis and automated inference. The fundamental process is to assign probabilities over a defined set of categories—the scene characteristics—based on analysis of the current visual state. This paper shows the practicability of a lightweight approach that avoids much of the complexity of object recognition methods, by reducing the problem to a sequence of empirical machine learning tasks.

The problem we have applied this to is classification of scene type by analysis of a video stream from a moving platform, specifically from a car. In this paper we address aspects of spatial reasoning—clearly there is also a temporal reasoning aspect, which is not considered here. In figurative terms the problem may be compared with Google's *Streetview*® application. *Streetview's* purpose is to tell you what your surroundings look like by knowing your location. The scene recognition problem is the opposite: to characterize your location from what your surroundings look like.

In this paper we consider a classification scheme for images where the image is subject to classification in multiple categories. We will consider outdoor roadway scenes, and these classification categories:

1. surroundings, zoning, development (urban, residential, commercial, mountainous, etc.)
2. visibility (e.g., illumination and weather),
3. roadway type,
4. traffic and other transient conditions,
5. roadway driving obstacles.

An image will be assigned one label from each of the set of five categories.

1.1 Uses of Scene Classification

There are numerous uses where the automated classification assigned to a scene can help. The purpose of scene classification is to capture the gist of the current view from its assigned category labels. For example, how would you describe a place from what you see? Certainly this is different from what you would know from just the knowledge of your lat-long coordinates. These are some envisioned uses:

- A scene classification provides context. For example in making a recommendation, the context could be to consider the practicality of the request: For instance, "Do you want to get a latte now? This is not the kind of neighborhood for that."

- Supplement search by the local surroundings. For example, "Find me a winery in a built-up area." "Find me a restaurant in a remote place." "Find a park in a less-travelled residential area."
- Coming up with a score for the current conditions. How is the view from this place? How shaded or sunny is the area? What fraction of the surroundings are natural versus artificial? Taking this one step further, given an individual driver's ratings of preferred locations, suggest other desirable routes to take, possibly out of the way from a "best" route.
- Distributed systems could crowd-source their findings about nearby locations to form a comprehensive picture of an area. For example, "How far does this swarm (road-race, parade) extend?"

1.2 Relevant previous work

One of the earliest formulations of image understanding as a PGM is found in Levitt, Agosta, and Binford (1989) and Agosta (1990). The approach assumed an inference hierarchy from object categories to low-level image features, and proposed aggregation operators that melded top-down (predictive) with bottom-up (diagnostic) reasoning over the hierarchy.

The uses of PGMs in computer vision have expanded into a vast range of applications. Just to mention a couple of examples, L. Fei-Fei, Fergus and P. Perona (2003) developed a Bayesian model for learning new object categories using a "constellation" model with terms for different object parts. In a paper that improved upon this, L. Fei-Fei and P. Perona (2005) proposed a Bayesian hierarchical theme model that automatically recognizes natural scene categories such as forest, mountains, highway, etc. based on a generalization of the original texton model by T. Leung and J. Malik (2001) and, L. Fei-Fei R. VanRullen, C. Koch, and P. Perona (2002). In another application of a Bayesian model, Sidenbladh, Black, and Fleet (2000) develop a generative model for the appearance of human figures. Both of these examples apply model selection methods to what are implicitly PGMs, if not explicitly labeled as such.

Computer vision approaches specifically to scene recognition recognize the need to analyze the image as a whole. Hoiem, Efros, and Hebert (2008) approach the problem by combining results from a set of *intrinsic images*, each a map of the entire image for one aspect of the scene. Oliva and Torralba (2006) develop a set of scene-centered global image features that capture the spatial layout properties of the image. Similar to our approach, their method does not require segmentation or grouping steps.

1.3 How Scene Classification differs from Object Recognition

Scene classification implies a holistic image-level inference task as opposed to the task of recovering the identity, presence, and pose of objects within an image. Central to object recognition is to distinguish the object from background of the rest of the image. Typically this is done by segmenting the image into regions of smoothly varying values separated by abrupt boundaries, using a bottoms-up process. Pixels may be grouped into "super-pixels" whose grouping is further refined into regions that are distinguished as part of the foreground or background. Object recognition then considers the features and relationships among foreground regions to associate them with parts to be assembled into the object, or directly with an entire object to be recovered.

Scene classification as we approach it does not necessarily depend on segmenting the image into regions, or identifying parts of the image. Rather it achieves a computational economy by treating the image as a whole; for example, to assign the image to the class of "indoor," "outdoor," "urban landscape," or "rural landscape," etc. from a set of pre-defined categories. We view classification as assigning a posterior to class labels, where the image may be assigned a value over multiple sets of labels; equivalently, the posterior may be a joint distribution over several scene variables.

Despite the lack of a bottoms-up segmentation step in our approach, our method distinguishes regions of the image by a partition that is prior to analyzing the image contents. This could be a fixed partition, which is appropriate for a camera in a fixed location such as a security camera, or it could depend on inferring the geometry of the location from sources distinct from the image contents, such as indicators of altitude and azimuth of the camera. In our case, the prior presumption is that the camera is on the vehicle, facing forward, looking at a road.

The rest of this paper is organized as follows. Section 2 describes the inference procedure cascade; the specific design and learning of the Bayes network PGM is the subject of Section 3, and the results of the learned model applied to classification of a set of images is presented in Section 4.

2 Lightweight inference with virtual evidence

In treating the image as a whole, our approach to inference for scene classification takes place by a sequence of two classification steps:

- First the image's individual pixels are classified, based on pixel level features. This classifier resolves the pixel into one of n discrete types, representing the kind of surface that generated it. In our examples $n = 8$: sky, foliage, building-structure, road-surface, lane, barrier-sidewalk, vehicle, and pedestrian.

- In the second step, the pre-defined partitions are applied to the image and in each partition the pixel types are histogrammed, to generate a likelihood vector for the partition. These likelihoods are interpreted as virtual evidence[1] for the second level image classifier, the scene classifier, implemented as a PGM. The classifier returns an joint distribution over the scene variables, inferred from the partitions' virtual evidence.

There is labeled data for both steps, to be able to learn a supervised classifier for each. Each training image is marked up into labeled regions using the open source *LabelMe* tool, (Russell, Torralba, K. Murphy and Freeman, 2007) and also labeled by one label from each category of scene characteristics. From the region labelings a dataset of pixels, with color and texture as features, and the region they belong to as labels can be created. In the second step we learn the structure and parameters of a Bayes network—a discrete valued PGM—from the set of training images that have been manually labeled with scene characteristics. Each image has one label assigned for each scene characteristic. The training images are reduced to a set of histograms of the predicted labels for the pixels, one for each partition. The supervised data for an image consists of the histogram distributions and the label set.

Scene recognition output is a summarization of a visual input as an admittedly modest amount of information from a input source orders of magnitude greater–even mores than for the object recognition task. From the order of 10^6 pixel values we infer a probability distribution over a small number of discrete scene classification variables. To obtain computational efficiency, we've devised an approach that summarizes the information content of the image in an early stage of the process that is adequate at later stages for the classification task.

2.1 Inference Cascade

The two phases in the inference cascade can be formalized as follows, starting from the pixel image and resulting in a probability distribution over scene characteristics. Consider an image of pixels p_{ij} over $i \times j$, each pixel described by a vector of features $\mathbf{f}_{ij}$. The features are derived by a set of filters, e.g. for color and texture, centered at coordinate (i, j). A pixel-level classifier is a function from the domain of $\mathbf{f}$ to one of a discrete set of n types, $C : \mathbf{f} \rightarrow \{c^{(1)}, \cdots c^{(n)}\}$. The result is an array of classified image pixels.

A pre-determined segmentation, G_m partitions the pixels in the image into M regions by assigning each pixel to one region, $r_m = \{p_{ij} \,|\, p_{ij} \in G_m\}, m = 1 \ldots M$, to form regions that are contiguous sets of pixels. Each region is described by a histogram of the pixel types it contains: $H_m = \left(|C(\mathbf{f}_{ij}) = c^{(1)}|, \cdots |C(\mathbf{f}_{ij}) = c^{(n)}|\right) s.t. \, \mathbf{f}_{ij} \in G_m$, for which we introduce the notation, $H_m = \left(|c_{ij}^{(1)}|_m, \cdots |c_{ij}^{(n)}|_m\right)$, where $|c^{(i)}|_m$ denotes the count of pixels of type $c^{(i)}$ in region m. The scene classifier is a PGM with virtual evidence nodes corresponding to the M regions of the image. See Figure 3. Each evidence node receives virtual evidence in the form of a lambda message, λ_m, with likelihoods in the ratios given by H_m. The PGM model has a subset of nodes $\mathbf{S} = \{S_1, \cdots S_v\}$, distinct from its evidence nodes, for scene characteristic variables, each with a discrete state space. Scene classification is completely described by $\mathsf{P}(\mathbf{S} \,|\, \lambda_1, \cdots \lambda_M)$, the joint of $\mathbf{S}$ when the λ_m are applied, or by a characterization of the joint by the MAP configuration over $\mathbf{S}$, or just the posterior marginals of $\mathbf{S}$.

2.2 Partitions of Pixel-level Data

As mentioned we avoid segmenting the image based on pixel values by using a fixed partition to group classified pixels. We introduce a significant simplification over conventional object recognition methods by using such a segmentation. This makes sense because we are not interested in identifying things that are in the image, but only in treating the image as a whole. For instance in the example we present here, the assumption is that the system is classifying an outdoor roadway scene, with sky above, road below, and surroundings characteristic of the scene to either side. The partitions approximate this division. The image is partitioned symmetrically into a set of twelve wedges, formed by rays emanating from the image center.

For greater efficiency the same method could be applied over a smoothed, or down-sampled image, so that every pixel need not be touched, only pixels on a regular grid. The result of the classification step is a *discrete class-valued image* array. See Figure 2. Despite the classifier ignoring local dependencies, neighboring pixels tend to be classed similarly, and the class-valued

[1] Sometimes called "soft evidence." We prefer the term virtual evidence, since soft evidence is also used to mean an application of Jeffrey's rule of conditioning that can change the CPTs in the network.

image resembles a cartoon version of the original.

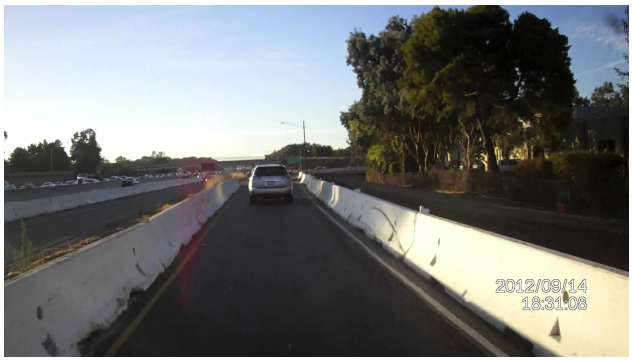

Figure 1: The original image. The barriers bordering the lane are a crucial feature that the system is trained to recognize.

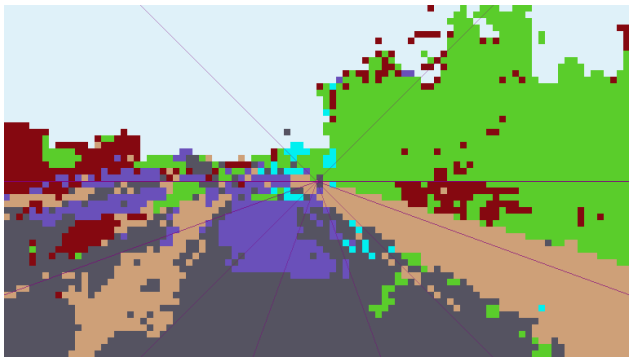

Figure 2: The image array of $C(\mathbf{f}_{ij})$, the pixel classifier, on an image down-sampled to 96 by 54. Rays emanating from the image center show the wedge-shaped regions. Colors are suggestive of the pixel class, e.g. green indicates foliage and beige indicates barriers.

2.2.1 Inferring partition geometry

The point chosen as the image center, where the vertices of the wedges converge approximates the vanishing point of the image. Objects in the roadway scene tend to conform (very) roughly to the wedge outlines so that their contents are more uniform, and hence, likelihoods are more informative. For example, the contents of the image along the horizon will fall within one wedge, and the road surface within another.

2.3 The image as a source of virtual evidence

For each wedge that partitions the image, the evidence applied to the Bayes network from the wedge m is: $\lambda_m \propto |c_{ij}^{(1)}|_m : |c_{ij}^{(2)}|_m : \cdots : |c_{ij}^{(n)}|_m$. One typically thinks of virtual evidence as a consequence of measurements coming from a sensor that garbles the precise value of the quantity of interest—where the actual observed evidence value is obscured by an inaccuracy in the sensor reading. Semantically, one should not think of the virtual image evidence as a garbled sensor variable. Rather it *is* the evidence that describes the region.

3 Bayes network design

Formally a Bayes network is a factorization of a joint probability distribution into local probability models, each corresponding to one node in the network, with directed arcs between the nodes showing the conditioning of one node's probability model on another's (Koller and Friedman, 2010). Inference—for example, classification—operates in the direction against the causal direction of the arc. In short, inference flows from lower level evidence in the network upward to the class nodes at the top of the network where it generates the posterior distributions over the class variables, in this case, the scene characteristics. We learn a fully observable Bayes network with virtual evidence for scene classification.

3.1 How the structure and parameters are defined

The design of the Bayes network model is fluid: It is easily re-learned under different partition inputs, output categories and structural constraints. The ability to easily modify the model to test different kinds of evidence as inputs, or differently defined nodes as outputs is an advantage of this approach. The structure of the model discovers dependencies among the model variables that reveal properties of the domain.

Learning the Bayes network is composed of two aspects; the first, learning the variables' structure, the second, learning the parameters of the variable conditional probability tables. The algorithm used is SMILE's *Bayesian Search* (Druzdzel et al., 1997), a conventional fully observable learning algorithm, with a Bayesian scoring rule used to select the preferred model. Learning structure and parameters occur simultaneously.

The model is structured into two levels, the top level of outputs and the lower level of inputs as shown in Figure 3. This is the canonical structure for classification with a Bayes network, in this case a multi-classifier with multiple output nodes. In the learning procedure this node ordering is imposed as a constraint on the structure, so that conditioning arcs cannot go from the lower level to the upper level.

Further constraints are used to limit in-degree and node ordering. The in-degree of evidence nodes is

limited to two. Node ordering of output nodes follows common sense causal reasoning: for instance, the "Surroundings" variable influences the "Driving Conditions" and not the other way around. The model consequently follows an approximately näive Bayes structure for each scene variable, but with additional arcs that are a consequence of the model selection performed during learning. The resulting network is relatively sparse and hence learning a network of this size, let alone running inference on it can be done interactively.

3.2 Bayes Network Learning Dataset

An interesting challenge in learning this model is that there is no conventional procedure for learning from virtual evidence, such as the histogram data.

3.2.1 Consideration of partition contents as virtual evidence

We considered three ways to approximate learning the Bayes network from samples that include virtual evidence.

1) Convert the dataset into an approximate equivalent observed evidence dataset by generating multiples of each evidence row, in proportion to the likelihood fraction for each state of the virtual evidence. If there are multiple virtual evidence nodes, then to capture dependencies among virtual evidence nodes this could result in a combinatorial explosion of row sets, one multiple for each combination of virtual evidence node states, with multiplicities in proportion to the likelihood of the state combination. This is equivalent in complexity to combining all virtual evidence nodes into one node for sampling.

Similarly one could sample from the combination of all virtual evidence nodes and generate a sample of rows based on the items in the sample. This is a bit like logic sampling the virtual states.

Both these methods make multiple copies of a row in the learning set as a way to emulate a training weight. Instead one could apply a weight to each row in the sampled training set, in proportion to its likelihood.

2) One could also consider a mixture, a "multi-net," of learned deterministic evidence models. The models would have the same structure, so the result would be a mixture of CPTs, weighted (in some way) by the likelihoods. It appears this would also suffer a combinatorial explosion of mixture components, and might be amenable to reducing the set by sampling.

3) Alternatively, one could consider the virtual evidence by a virtual node that gets added as a child to the evidence node, which is then instantiated to send the equivalent lambda msg to its parent. This is the method used in Refaat, Choi and Darwiche (2012). With many cases, there would be a set of virtual nodes added to the network for each case, again generating a possibly unmanageable method. Perhaps there is an incremental learning method that would apply: Build a network with one set of nodes, do one learning step, then replace the nodes with the next set, and repeat a learning step.

4 Results on a sample dataset

In this section we present the evaluation of the Bayes network as a classifier. We argue that the first-stage pixel-level classifier, whose accuracy approaches 90%, is a minor factor in the scene classification results, since the partition-level inputs to the Bayes network average over a large number of pixels, although this premise could be tested.

4.1 Learning from a sampled dataset

The sample dataset to learn the model was a further approximation on alternative 1), where each virtual evidence node was sampled independently to convert the problem into an equivalent one with sampled data. Each histogram was sampled according to its likelihood distribution, to generate a set of conventional evidence samples that approximated the histogram. The result was an expanded dataset that multiplied the number rows by the sample size for each row in the histogram dataset. The resulting dataset description is:

1. Original data set: 122 rows of 12 region histograms of images labeled by 5 scene labels.
2. Each region histogram is sampled 10 times, to generate 1220 rows
3. Final data set of 5 labels and 12 features by 1220 rows

4.2 Inference Results

As mentioned, the second-stage Bayes network classifier infers a joint probability distribution over the set of scene characteristic nodes—the nodes shown in orange in Figure 3. We will evaluate the scene classifier by the accuracy of the predicted marginals, comparing the highest posterior prediction for each scene variable with the true value.[2]

[2]The "dynamic environment" variable is not counted in the evaluation results, since most all labeled data was collected under overcast conditions, making the predicted results almost always correct, and uninteresting.

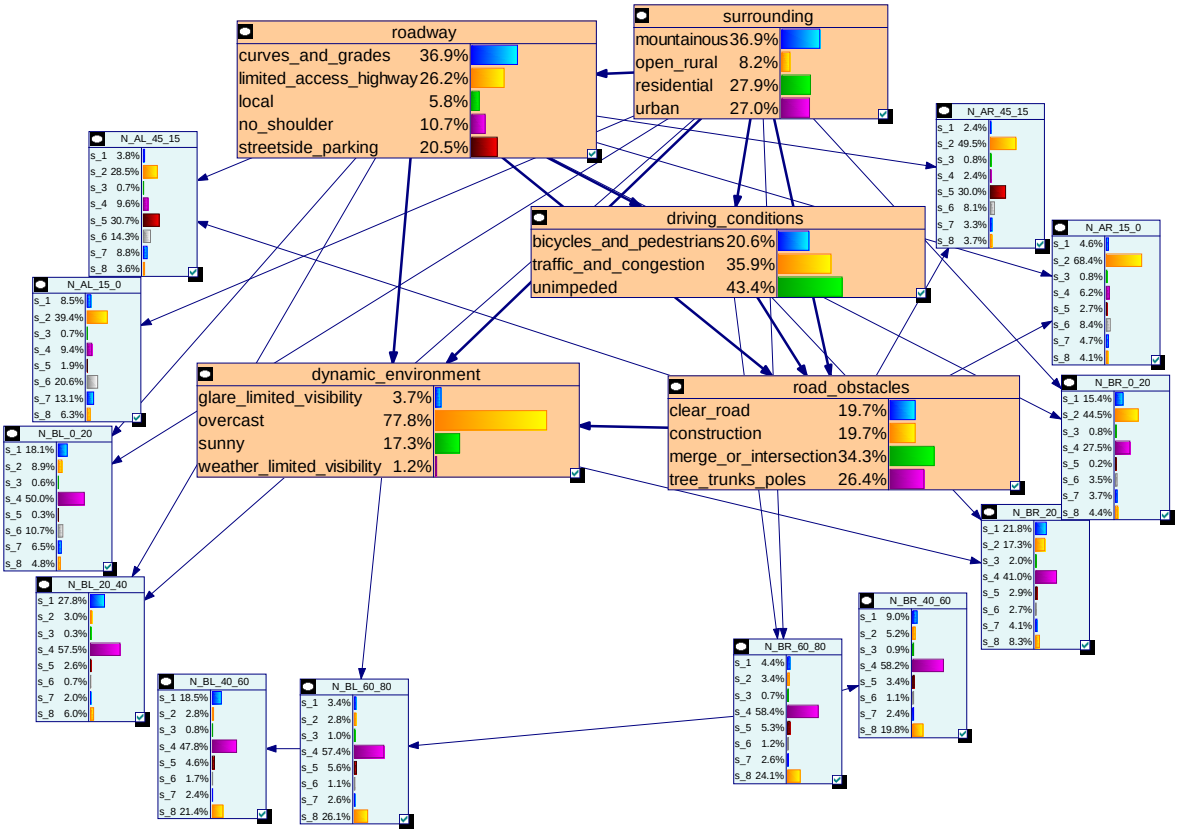

Figure 3: The entire Bayes network used for scene classification. Input nodes, corresponding to the wedges that partition the image are shown in light blue, and output nodes for the scene variables are in orange. The input nodes are arranged roughly in the positions of the corresponding wedges in the image. The input node histograms show the virtual evidence applied from that wedge. The labels used here for virtual evidence states, $s\_1, \ldots s\_8$ correspond to the classifier outputs $c^{(1)}, \ldots c^{(8)}$.

The matrix of counts of the true class by the predicted class is called a confusion matrix. The row sum of the confusion matrix for any class divided into the diagonal (true count) is the fraction of correct cases out of those possible, known as the recall or the coverage for that class. The column sum divided into the diagonal element is the fraction classified with that columns class label that truly belong to that class, which is called the precision. Tables $1-4$ show the recall and precision for each class, for each of the scene variables. As may be expected "Surroundings" that takes in the entire image performs better than "Road obstacles" that requires attention to detail in just the car's lane. This poor performance is even more true with "Bicycles and pedestrians," in Table 3 that appear in small areas of the image. In other classes either precision or recall approach 1.0, except for "Local" roads, where all cases were confused with "Curves and grades," again due to the limited variety in the training set.

Beyond evaluating the accuracy of marginal predictions, we can also make observations about the structure learned for the Bayes network. Arcs in the learned model show which wedge histograms are relevant to which scene variables. These arcs are relatively sparse, in part due to the afore-mentioned design constraint in-degree arc limit of two. The arcs chosen by the structure learning algorithm show a strong association between the location of the partitions, and different scene variables. We see this in the associations where the "Driving conditions" scene variable connects to partitions at the base of the image, and "Surroundings" connects to partitions on the image periphery. The relevance of the two wedges at the bottom of the diagram is limited, since their only incoming arcs are from other wedges, indicating that their evidence is supported entirely by neighboring wedges. We leave them in the model, since in the case of virtual evidence they will still have some information value for classification. Further along these lines, in terms of wedge dependencies, only one arc was learned between wedge histograms, indicating that the evidence contributed

to the scene is conditionally independent in all but this case. The sub-network of scene variables is more connected, indicating strong dependencies among the scene variables. Some of these are to be expected, for instance "Curves and grades" correlates strongly with "Mountainous" surroundings. Some are spurious, as a result of biased selection of the training sample images, (e.g. all divided highway images corresponded to overcast scenes) and have been corrected by adding more samples.

4.3 Discussion and Conclusion

We have demonstrated a novel scene classification algorithm that takes advantage of the presumed geometry of the scene to avoid computationally expensive image processing steps characteristic of object recognition methods, such as pixel segmentation, by a cascade of a pixel level and fixed partition level multi-classifier, for which we learn a Bayes network. As a consequence of the partition-level data we learn the Bayes network with virtual evidence.

The Bayes network classifies the scene in several dependent dimensions corresponding to a set of categories over which a joint posterior of scene characteristics is generated. Here we have only considered the marginals over categories, however it is a valid question whether a MAP interpretation—of the most likely combination of labels—is more appropriate.

The use of virtual evidence also raises questions about whether it is proper to consider the virtual evidence likelihood as a convex combination of "pure" image data. Another interpretation is that the histograms we are using are better"sliced and diced" to generate strong evidence from certain ratios of partition content. For instance a partition that includes a small fraction of evidence of roadway obstacles—think evidence of a small person—may be a larger concern than a partition obviously full of obstacles, and should not be considered a weaker version of the extreme partition contents. These subtleties could be considered as we expand the applicability of the system. In this early work it suffices that given the approximations, useful and accurate results can be achieved at modest computational cost.

Acknowledgements

This work would not have been possible without valuable discussions with Ken Oguchi and Joseph Dugash, and by Naiwala Chandrasiri, Saurabh Barv, Ganesh Yalla, and Yiwen Wan.

References

Agosta, J. M., 1990. The structure of Bayes networks for visual recognition, UAI (1988) North-Holland Publishing Co., pp. 397 - 406.

Druzdzel, M. et. al., 1997. SMILE (Structural Modeling, Inference, and Learning Engine) http://genie.sis.pitt.edu/.

Fei-Fei, L., R. Fergus, P. Perona, 2003. A Bayesian Approach to Unsupervised One-Shot Learning of Object Categories (ICCV 2003), pp. 1134-1141 vol.2.

Fei-Fei L. and P. Perona, 2005. A Bayesian Hierarchical Model for Learning Natural Scene Categories. (CVPR 2005), pp. 524-531.

Fei-Fei, L., R. VanRullen, C. Koch, and P. Perona, 2002. Natural scene categorization in the near absence of attention(PNAS 2002), 99(14):95969601.

Hoiem, D., A. A. Efros, and M. Hebert. 2008. Closing the Loop in Scene Interpretation.2008 IEEE Conference on Computer Vision and Pattern Recognition (June): 18.

Koller, D., and Friedman, N. 2010. Probabilistic Graphical Models: Principles and Techniques. Cambridge, Massachusetts: The MIT Press.

Leung, T. and J. Malik, 2001. Representing and recognizing the visual appearance of materials using three-dimensional textons(IJCV 2001), 43(1):2944.

Levitt, T., J. M. Agosta, T.O. Binford, 1989. Model-based influence diagrams for machine vision, (UAI 1989).

Oliva, A., and A. Torralba. 2006. Building the Gist of a Scene: The Role of Global Image Features in Recognition.Progress in Brain Research 155 (January): 2336.

Oliva, A., A. Torralba, 2001. Modeling the shape of the scene: a holistic representation of the spatial envelope. International Journal of Computer Vision, Vol. 42(3): 145-175.

Refaat, K. S. , A. Choi and A. Darwiche, 2012. New Advances and Theoretical Insights into EDML. (UAI 2012), pp. 705-714 .

Russell, B., A. Torralba, K. Murphy, W. T. Freeman, 2007. "LabelMe: a database and web-based tool for image annotation" International Journal of Computer Vision.

Sidenbladh, H., M. J. Black, and D. J. Fleet, 2000. Stochastic Tracking of 3D Human Figures Using 2D Image Motion. ECCV 2000: 702718.

	Mountainous	Open rural	Residential	Urban
Recall	1.0	0.9	0.794	0.45
Precision	0.642	1.0	0.964	1.0
Accuracy				0.784

Table 1: Surroundings

	Curves and grades	Limited access highway	Local	No shoulder	Streetside parking
Recall	1.0	0.75	0.0	0.85	0.56
Precision	0.61	1.0	NaN	1.0	1.0
Accuracy					0.770

Table 2: Roadways

	Bicycles and pedestrians	Traffic and congestion	Unimpeded
Recall	0.56	0.6	0.98
Precision	0.875	0.93	0.67
Accuracy			0.754

Table 3: Driving Conditions

	Clear road	Construction	Merge intersection	Tree trunks and poles
Recall	0.42	0.96	0.6	1.0
Precision	1.0	0.92	0.86	0.55
Accuracy				0.738

Table 4: Road Obstacles