# Characterization of 3-D Volumetric Probabilistic Scenes for Object Recognition

Maria I. Restrepo, *Member, IEEE*, Brandon A. Mayer, *Student Member, IEEE*, Ali O. Ulusoy, *Graduate Student Member, IEEE*, and Joseph L. Mundy

***Abstract*—This paper presents a new volumetric representation for categorizing objects in large-scale 3-D scenes reconstructed from image sequences. This work uses a probabilistic volumetric model (PVM) that combines the ideas of background modeling and volumetric multi-view reconstruction to handle the uncertainty inherent in the problem of reconstructing 3-D structures from 2-D images. The advantages of probabilistic modeling have been demonstrated by recent application of the PVM representation to video image registration, change detection and classification of changes based on PVM context. The applications just mentioned, operate on 2-D projections of the PVM. This paper presents the first work to characterize and use the local 3-D information in the scenes. Two approaches to local feature description are proposed and compared: 1) features derived from a PCA analysis of model neighborhoods; and 2) features derived from the coefficients of a 3-D Taylor series expansion within each neighborhood. The resulting description is used in a bag-of-features approach to classify buildings, houses, cars, planes, and parking lots learned from aerial imagery collected over Providence, RI. It is shown that both feature descriptions explain the data with similar accuracy and their effectiveness for dense-feature categorization is compared for the different classes. Finally, 3-D extensions of the Harris corner detector and a Hessian-based detector are used to detect salient features. Both types of salient features are evaluated through object categorization experiments, where only features with maximal response are retained. For most saliency criteria tested, features based on the determinant of the Hessian achieved higher classification accuracy than Harris-based features.**

***Index Terms*—3-D data processing, 3-D object recognition, machine vision, Bayesian learning.**

## I. Introduction

AUTOMATED description of real-world 3-D scenes is an important field of research for many urban and surveillance applications, including city planning, virtual tourism, autonomous navigation, and object localization, detection, and tracking. Much work has been done to solve the object recognition problem in 2-D images, and great performance advances have been achieved with the development of consistent image descriptors, e.g., SIFT [1] and HOG [2], non-parametric machine learning techniques, e.g., SVM, and the availability of public databases and competitions such as the PASCAL challenge. However, the appearance inconsistencies in aerial imagery of urban scenes caused by occlusions, non-Lambertian properties of materials, sensor noise, shadows, transient objects, and others, pose great challenges to 2-D recognition systems, where consistent viewpoint invariant features do not exist. Three-dimensional models of objects offer the advantage of using the full dimensionality of an object's shape and appearance information and avoid the ambiguities due to projection.

The availability of large-scale point cloud data collected with LIDAR sensors has led to recent efforts to detect and classify objects in large-scale urban 3-D models. Examples are the works of Golovinskiy *et al.* [3], Frome *et al.* [4], Korah *et al.* [5], and Patterson *et al.* [6]. The work in this paper addresses the same application but based on a novel data representation. Specifically, this paper presents algorithms to represent and classify objects in large-scale probabilistic volumetric scenes that are learned from aerial imagery. Objects are represented as bag of "volumetric words" that are learned from the appearance and occupancy information in local neighborhoods of probabilistic volumetric models (PVM). The PVM allows for dense 3-D reconstruction of a scene's appearance and geometry, and handles occlusions and ambiguities through probabilistic online updating. By combining the ideas of multi-view geometry and background modeling it is possible to learn occupancy information that systems based on hard thresholds would not recover due to appearance ambiguities. This advantage will be further demonstrated in the experimental section. Furthermore, one could imagine combining different kinds of available information, e.g., color, IR, LIDAR, etc., in a rigorous Bayesian framework to produce detailed 3-D urban models. All these reasons make the problem of object representation in probabilistic volumetric models a relevant and promising field of research.

The 3-D modeling framework used in this work, has been applied to video image registration [7], change detection [8], and classification of changes as vehicles in 2-D [9], [10]. In these applications, the probabilistic volumetric representation predicts occlusion and appearance variability, providing accurate detection of deviations from normal appearance in new images, i.e., change detection. To the authors' knowledge, the work presented in this paper is the first to characterize the 3-D information stored in the PVM, and furthermore, to base scene classification on a volumetric probabilistic model. To do so, voxels in an object are assigned a description of their neighborhood using principal component analysis or Taylor series

Manuscript received August 01, 2011; revised January 18, 2012, April 16, 2012; accepted May 07, 2012. Date of publication May 30, 2012; date of current version August 10, 2012. This work was supported by NGA NURI Grant HM1582-08-1-0015.. B. A. Mayer and A. O. Ulusoy have equal contribution. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Aydin Alatan.

The authors are with the School of Engineering, Brown University, Providence, RI 02912 USA (e-mail: maria_restrepo@brown.edu; brandon_mayer@brown.edu; ali_ulusoy@brown.edu; mundy@lems.brown.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTSP.2012.2201693

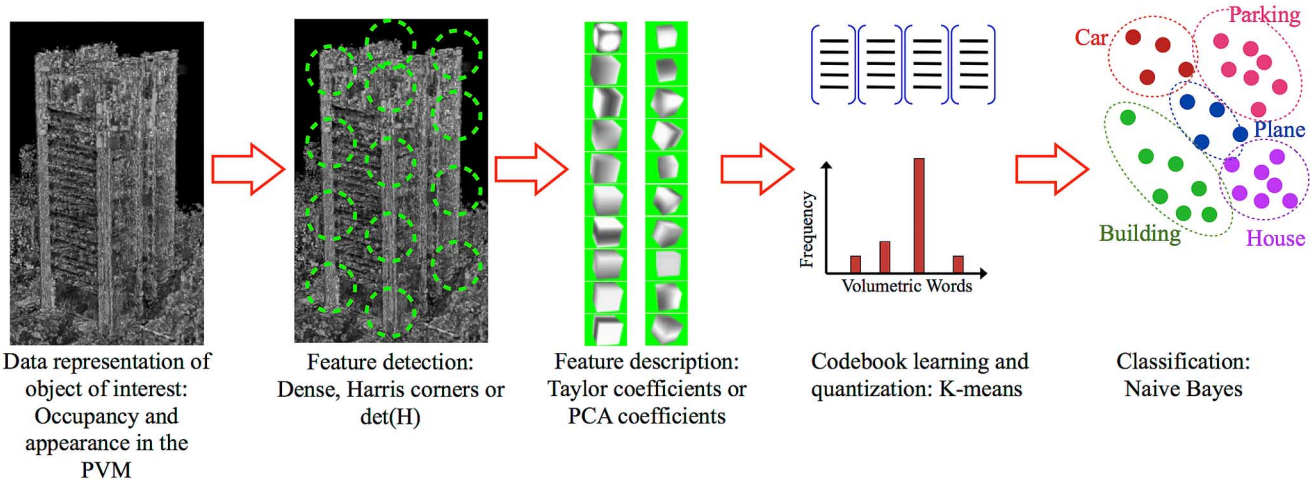



Fig. 1. Bag of volumetric features approach used to categorize objects from volumetric scenes.

approximation of the surface and appearance attributes. During the learning phase, descriptors from different objects of different categories are used to learn a common "volumetric vocabulary." This paper presents results for descriptors that are sampled in a dense manner or located at features that maximize the saliency response of the 3-D Harris detector, or the determinant of the Hessian. Finally, descriptors are assigned to the most similar vocabulary entry and quantized to learn distributions for different object categories. A Bayesian classifier is used during the testing phase to assign to each object the most probable class label. The workflow just described is illustrated in Fig. 1. The work in this paper aims to demonstrate, through simple feature description and recognition approaches, the potential of the PVM representation for object recognition. The use of features based on local neighborhoods in the PVM for recognition have not been explored thus far. This paper characterizes distribution of salient information in the PVM data and quantifies the effectiveness of local features in object categorization tasks.

The proposed framework was first introduced in a recent work by the authors [11], where the effectiveness of the proposed features for dense-feature classification is evaluated. This paper extends the analysis of these local volumetric features for 3-D object recognition. Extensions of the 3-D Harris corner detector and the determinant of the Hessian are used in the PVM to localize salient features. In addition, this paper presents comparisons between the proposed probabilistic volume PVM, and a state-of-the-art 3-D point cloud reconstruction algorithm [12]. Finally, the experimental evaluation reported here is more comprehensive than that in [11].

## II. Related Work and Discussion of Major Design Decisions

### A. 3-D Versus 2-D Modeling

While this paper is focused on the application of a 3-D representation to object classification, it is important to note that there exists a large body of recognition work that is based on descriptions that are derived from 2-D images of a scene. These approaches provide some inspiration for the type of features and recognition algorithms that might be extended to a 3-D representation. Most of the work on image-based recognition in realistic scenes is performed using appearance-based techniques. State-of-the-art systems use deformable part models [13] to handle shape variations in single image views and to account for the random presence/absence of parts caused by occlusion and variations in viewpoint and illumination. Multi-view models have also been proposed [14], where shape models are based on 2-D descriptors observed in multiple views, and single-view codebooks are learned and interconnected on the space of observer viewpoints. However, it can be argued that these multiview algorithms just reveal a subset of the rich set of relationships and constraints that arise from a full 3-D description.

Instead of basing recognition on single or multiple 2-D images, this work aims to use the PVM representation to extract a dense 3-D reconstruction of the objects. Then, shape and appearance is characterized to perform one single 3-D detection. In practice, the current system performs categorization and not localization, as it will be explained in a later section. By using a three-dimensional representation, the framework can take advantage of the full dimensionality of the learned appearance and geometry. Another advantage of 3-D models is their potential for better object/background segmentation by using depth information. Recently, Knopp *et al.* [15], [16], briefly investigated the application of their implicit shape 3-D models for detection in scenes reconstructed using structure from motion methods. The successful qualitative results obtained by Knopp *et al.* provide encouragement for further research on object recognition based on 3-D scene representation. Furthermore, the PVM has produced high resolution large-scale scene reconstructions [17], [18] that motivate their use for recognition applications.

### B. 3-D Data Representation

Many 3-D shape recognition/retrieval methods [16], [19]–[26] have been developed in recent years to search the rapidly growing databases of 3-D models. Throughout most of the object-retrieval literature, the underlying 3-D models are synthetically generated or obtained in a controlled environment using 3-D scanners. The dominant representation of 3-D geometry is a mesh or point cloud, where the geometric properties

of the representation are used to describe objects in recognition systems.

Fewer works have employed volumetric representations. However, volumetric descriptions are extensively used in medical image applications [27], [28] and for action recognition [29]. They have also been used to handle isometric deformations [30] and to segment models into parts [20]. Volumetric representations can come directly from sensors (MRI), or can easily be generated from meshes and point clouds (the contrary is significantly more difficult). Additionally, by operating on voxels (discrete 3-D volume cells), image processing methods can be easily generalized to 3-D. For example, Yu *et al.* [31], recently presented a performance evaluation of volumetric interest point detectors for 3-D data.

In the experiments reported here, images are collected under unrestricted conditions from an aerial platform and the reconstructed geometry is expected to be noisy and ambiguous. This work utilizes a volumetric representation as it allows for a natural way to model the uncertainty in 3-D surfaces reconstructed from 2-D images, which is difficult to characterize with point clouds or meshes. The representation of large 3-D urban scenes for recognition in terms of discrete point cloud data has been investigated to some extent, however to the best of the authors' knowledge this paper presents the first recognition system based on a dense probabilistic volume representation.

### C. Local Versus Global Descriptors

There exist many different approaches to 3-D shape analysis. For a review, the reader is referred to [32]. In the rigid shape retrieval community, global descriptors have been studied extensively. Global features characterize the overall shape of a 3-D model; examples are: features based on volume and area [33], reflective symmetry descriptors [34], 3-D Zernic invariants [35], among others. Instead of using global features directly, Osada *et al.* used distributions of global features that showed robust discrimination between classes of objects.

Local feature-based methods take into account the information in the neighborhood around points on the surface. Although methods of this type are less common in the shape retrieval community, they have recently gained popularity inspired by the success of local descriptors such as HOG [2], SIFT [1], SURF [36], and DAISY [37] for 2-D image-based recognition. There are a large number of local descriptors for 3-D shapes, including shape contexts [38], [39], local spherical harmonics [4], local patches [40], spin images [6], [25], tensors [41], distance maps [42], heat kernels [23], [30], and the SHOT descriptor [43]. There are also extensions to three dimensions of the SURF descriptor [16]. The popular SIFT descriptor has been used in a volumetric form for medical image analysis [27], [28], [44], and for action recognition [29]. Local features are preferred for applications where robustness to clutter, noise, and missing data is important [3], [4], [15], [16].

Local descriptors are used in this work to take advantage of the power of the dense probabilistic data provided by the PVM representation. Every voxel has well-defined local neighborhood data in spite of occlusion and appearance ambiguities that arise in cluttered urban scenes. Furthermore, the success of global shape descriptors is critically dependent on proper detection and segmentation of objects. While the majority of the work using global descriptors requires a segmented instance of the object, it has been demonstrated that local descriptors can handle object segmentation and classification tasks simultaneously [15].

### D. Invariance to Scene Transformations

Invariance to scene transformations can be achieved by normalizing the pose and scale of the 3-D objects prior to analysis, or by constructing a representation that is invariant under different transformations. The dataset used in the experiments described below contains objects of various sizes and aspect ratios. However, the internal camera parameters and approximate viewing distance are the same for all the aerial scenes considered in the experiments. Thus, the PVM representation is reconstructed in 3-D with consistent scale but with the unknown pose, i.e., 3-D rotation and translation.

It can be expected that local 3-D structural features such as the intersections of walls, roofs, etc., will exhibit repeatable PCA and derivative operator characteristics across scenes for a given feature orientation. The bag of features representation is invariant to feature position since only the frequency of occurrence of the feature-derived k-means codebook entries determines the classification outcome. However, features with different orientations and thus a different codebook histogram will be produced if an object is rotated. For the aerial scenes considered here, the orientation ambiguity is approximately confined to rotations in the $x - y$ plane since the vertical direction is consistently maintained by identifying the ground plane in the scene.

The remaining rotational ambiguity could be handled by employing feature descriptors that are intrinsically invariant to orientation, such as the heat kernel signatures [30] or by establishing a local reference frame with respect to surface normal orientation, see for instance [25], [45]. However, in this work, there is no attempt to develop such rotationally invariant descriptions and different object ground plane orientations are accounted for by the training process.

### E. Detection Versus Categorization

The majority of the work in the 3-D object retrieval community is performed on isolated objects. Although, the effects of noise and occlusion have been studied, the 3-D object databases typically consist of segmented objects. More recently, scans of entire 3-D scenes have become available with the advance of comprehensive 3-D reconstruction algorithms [8], [12], [18], [46], [47] and the ready availability of LIDAR and triangulation-based range sensors. With 3-D representations of entire scenes, there has been an increase of interest (and need) in the area of 3-D object segmentation. The works of Knopp *et al.* [15], Korah *et al.* [5], and Golovinsky *et al.* [3] have achieved significant progress in the area of 3-D segmentation in real life scenes. Ultimately, segmentation and recognition processing is intertwined, and the success of both depend on the availability of effective local features. The focus of this investigation is to characterize the performance of local operators on categorization of manually segmented objects, which will lay the groundwork for automatic object segmentation in future work.

### *F. Learning Approach*

The object recognition experiments described below are based on aerial imagery collected in Providence, RI, USA. Recognition training is based on eighteen volumetric scene models. These models represent a variety of landscapes, architectural styles and contain a large number of objects. It is believed that no major constraints are imposed by the place of collection and the proposed framework is applicable to data from other geographic locations. Each model, composed of approximately 30 million voxels, covers an estimated ground area of $(500 \times 500)$ m$^2$. The data used in these experiments has been released to the community[1] to support progress in 3-D scene understanding. This work presents the first framework capable of processing large scale probabilistic volumetric scenes for object categorization tasks. Scenes are processed to compute local descriptors that are combined to form bag-of-features object representations. Learning is done in a supervised manner and categorization tasks are performed for objects from five different categories. This processing pipeline provides a basic framework for the development of more complex feature descriptors and recognition algorithms in such probabilistic volumetric scenes.

## III. Contributions

In summary, the contributions of the work presented in this paper are as follows.

1) The first work to implement a framework for object categorization tasks on probabilistic volume models that combine geometry and appearance information, and that are learned in unrestricted settings from aerial image sequences.
2) The construction of novel, image viewpoint-invariant, volumetric features extracted from probabilistic information of 3-D surface geometry and appearance.
3) A demonstration of the descriptive power, through rigorous analysis of function approximation and object recognition experiments, of features based on a Taylor series approximation, and PCA analysis of the probabilistic information in the models.
4) An analysis of the effectiveness of salient differential features in representing object categories described by probabilistic volume models.
5) The creation of the largest database of probabilistic volume models available today.

## IV. Probabilistic Volume Model

In general, the problem of reconstructing 3-D surfaces from 2-D image projections is ill-posed. Bhotika *et al.* [48] characterize the difficulties of inferring 3-D shapes from a set of $n$ noisy images as: scene *ambiguity* and scene *uncertainty*. Shape ambiguity arises due to the existence of multiple photo-consistent solutions of the multi-view reconstruction problem. This situation is shown in Fig. 2(a), where the resulting surface can lie anywhere within the diamond-shape regions in Fig. 2(b). On the other hand, shape uncertainty is caused by the presence of sensor noise, camera calibration errors, violations of a surface reflectance model, and occlusions (see Fig. 3). The 3-D models used in this work, and to be described in this section, infer surface and appearance information by combining the ideas of probabilistic background modeling and multi-view 3-D reconstruction, handling scene ambiguities, and uncertainties through Bayesian inference.

[1]http://vision.lems.brown.edu/project_desc/Object-Recognition-in-Probabilistic-3D-Scenes.

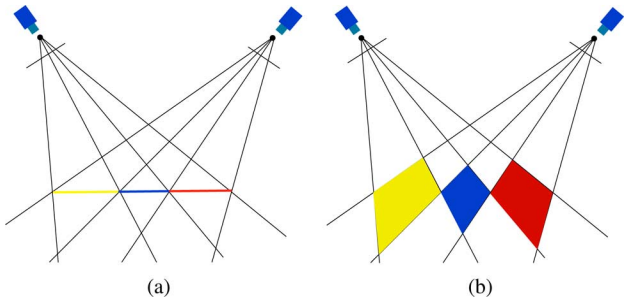



Fig. 2. Ambiguity of surface geometry for featureless surfaces. (a) Two cameras view a surface with three uniformly colored region. (b) The reconstructed surface can lie anywhere within the shaded regions.


Fig. 3. Various sources of uncertainty in the scene geometry of the Capitol building, Providence, RI. (a) A sample video frame. (b) Transient foreground objects such as cars. (c) Specularities not modeled by reflectance model. (d) Appearance variation due to sensor noise and other nonlinear effects.

The probabilistic volume model used in this work was first proposed by Pollard and Mundy [8], [49]. The framework is designed to be updated in an online manner to be able to adapt to changing world surfaces and moving objects. In Pollard's model, a region of three-dimensional space is decomposed into a regular 3-D grid of cells, called voxels [see Fig. 4(a)]. At any point in time, a voxel $X$ has two possible states: contains empty space or contains a solid surface. The probability that a voxel X contains a surface element is denoted $P(X \in S)$. Voxels are also associated with a probabilistic model for appearance as observed in images. In this paper, the image appearance is restricted to grayscale, but in general the model can be extended to handle, color, IR, or LIDAR [49]. Appearance is modeled with a Gaussian mixture distribution that can account for a range of variability due to illumination direction, shadows and image misregistration. It is also assumed that a calibrated camera model is supplied with each image to be processed.

The estimation of appearance and geometry is a joint process. It takes into account the success of the appearance models in explaining the observed image intensity, as well as how likely a voxel is to contain the observed surface given the possibility of occlusion. The process of updating the appearance model and occupancy probabilities is explained in the following subsections.

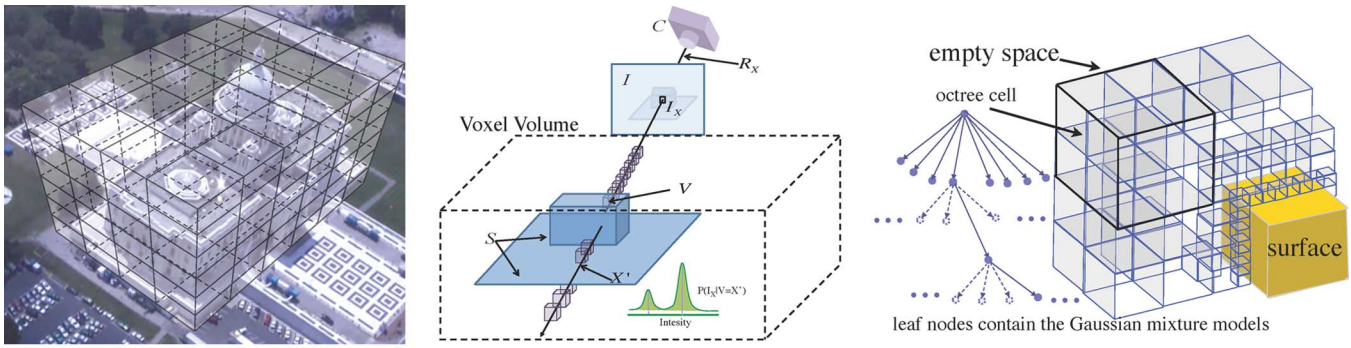



Fig. 4. PVM proposed by Pollard and Mundy [8], [49]. (a) The volume of interest is decomposed into a regular grid of voxels. (b) The observed pixel $I_X$ is caused by an *a priori* unknown voxel $V$ lying a long the ray $R_X$. In order to update the surface probability and appearance model of voxel $X$, all other voxels $X'$ that lie on $R_X$ have to be considered. (c) Octree subdivision of space proposed by Crispell [50].

### A. Updating the Appearance Model

The appearance of each voxel is modeled with a Gaussian mixture distribution as given by (1). $I$, refers to the grayscale intensity but could be considered a vector with various elements to account for color. The quantities, $\mu_k$, $\sigma_k$ and $\omega_k$, are the mean, covariance, and mixing parameters associated with each Gaussian distribution. $W$ is the sum of $\omega_k$ for all $k$. The number of mixtures is given by $k$; for this particular example there are three mixture components:

$$p(I) = \sum_{k=1}^{3} \frac{w_k}{W} \left( \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp^{-\frac{(I-\mu_k)^2}{2\sigma_k^2}} \right). \tag{1}$$

The parameters of the mixture are learned using a modified expectation maximization (EM) algorithm similar to that used in video background modeling [51]. The update of the parameters is as follows:

$$\begin{aligned} \omega_k^{N+1} &= \omega_k^N + d\omega_k^{N+1} \\ \mu_k^{N+1} &= \mu_k^N + \frac{d\omega}{d\omega + \omega_k^N} \left( I^{N+1} - \mu_k^N \right) \\ \left(\sigma_k^{N+1}\right)^2 &= \left(\sigma_k^N\right)^2 + \frac{d\omega}{d\omega + \omega_k^N} \left( \left( I^{N+1} - \mu_k^N \right)^2 - \left(\sigma_k^N\right)^2 \right). \end{aligned} \tag{2}$$

The increment in mixing weight, $d\omega$, upon observing image $N+1$ is determined by analyzing the distributions in other voxels along the same camera ray, that could contribute to pixel intensity, $I^{N+1}$. This computation is described in the next section. The components of the distribution are adapted as necessary to account for image intensities (colors) as new images are observed. If an intensity value $I$ is not within few standard deviations of any mode, the least probable mode is destroyed and replaced with a high variance mode with mean $I^{N+1}$ and weight $d\omega$. If a narrow range of intensity values are observed over the image sequence, then the mixing probability of the nearest component will approach one and the density will be sharply peaked around the mean.

### B. Updating the Surface Probabilities

The update to the mixture distribution of a particular voxel $X$ is determined by considering all the voxels $\{X'\}$ along the ray $R_X$ (through $X$) and the corresponding image pixel location as shown in Fig. 4(b). The ray may intersect several surfaces in the world. It is not known for certain which voxel produced the color in the image, but the probability of each voxel $X'$ producing the color, $P^N(V = X')$, can be computed. $P^N(V = X')$ depends on the belief that voxel $X'$ is a surface element and that it is not occluded by other voxels along the ray. The surface probability is updated by incremental Bayesian learning, as shown in (3) and (4) at the bottom of the page, where the probability of a voxel X containing a surface element after $\mathrm{N}+1$ images increases if the Gaussian mixture (1) at that voxel explains the intensity observed in the $\mathrm{N}+1$ image better than any other voxel along the projection ray.

To make the PVM representation clear, a term by term explanation of the update equation in (4) is outlined.

- The term $p^N(I_X^{N+1} \mid V = X')$ is computed using the mixture of Gaussians model stored at the voxel $X'$.
- The probability of a voxel $X'$ producing the color in the image is interpreted geometrically, where a voxel produces the intensity seen in the image if it is a surface element and it is not occluded by other voxels along the ray. Thus,

$$P^N(V = X') = P^N(X' \in S) P^N(X' \text{ is not occluded}). \tag{5}$$

$$P^{N+1}(X \in S) = P^N(X \in S) \frac{p^N\left(I_X^{N+1} \mid X \in S\right)}{p^N\left(I_X^{N+1}\right)} \tag{3}$$

$$= P^N(X \in S) \frac{\sum_{X' \in R_X} p^N\left(I_X^{N+1} \mid V = X'\right) P(V = X' \mid X \in S)}{\sum_{X' \in R_X} p^N\left(I_X^{N+1} \mid V = X'\right) P^N(V = X')} \tag{4}$$

The probability of occlusion is defined as the probability that all voxels between $X'$ and the sensor are empty, namely:

$$P^N(X' \text{ is not occluded}) = \prod_{X''<X'} (1 - P^N(X'' \in S)). \tag{6}$$

- The term $P^N(V = X' \,|\, X \in S)$ is computed analogously to $P^N(V = X')$. However, any instance of $P^N(X \in S)$ takes probability one.

### C. Effects of the Number of Views and the Camera Path

Pollard showed that a voxel $X$ lying on a world surface will converge to the correct probability provided that the surface images a constant color in all views, and that all voxels $X'$ near $X$ lying beneath the surface frequently project into some pixels of sufficiently different color from the true surface color at $X$. The camera path and the number of images available influence the quality of the models. For incomplete camera paths (those that are not a full circle), it is expected that models will have missing information. Camera paths with views at grazing angles, impede the surface reconstruction near the ground plane, due to occlusion. It is observed that the convergence of surface models is improved by the use of additional images. Typically, one hundred images with distinct viewpoints are required to produce good quality models, depending on the degree of scene occlusion.

### D. Octree Representation of the PVM

In a fixed-grid voxel representation, most of the voxels may correspond to empty areas of a scene, making storage of large, high-resolution scenes prohibitively expensive. Crispell [17], [50] proposed a continuously varying probabilistic scene model that generalizes the discrete model proposed by Pollard and Mundy. All the quantities proposed by Pollard can be computed using Crispell's continuous model. The details of this model are not included here, but the reader can refer to [17], [50] for further information. Crispell's model allows nonuniform sampling of the volume leading to an octree representation that is more space-efficient and can handle finer resolution required near 3-D surfaces; see Fig. 4(c).

The adaptive resolution representation proposed by Crispell makes it feasible to store models of large urban areas. However, learning times of large scenes using the PVM remained impractical until recently, when a GPU implementation was developed by Miller *et al.* [18]. With a GPU framework in place it is now possible to carry out multi-class object recognition tasks where the large number of objects instances required for training can be processed in a reasonable amount of time.

## V. Volumetric Feature Description

The focus of the work by Pollard [8] was on detecting changes in a new image. Change detection is based on the fact that occupancy and appearance information in the model can be used to render synthetic images of the expected scene appearance [49]. The predicted appearance of a given pixel in the image, is computed as the summation across all voxels of the corresponding back-projection ray. The ray summation is an expectation based on the appearance distribution of each voxel and the likelihood that each voxel on the ray is responsible for the observed image intensity.

The equations used to generate expected images are defined in a similar way to those presented in Section IV-B. Consider a pixel $I_X$, which back projects into a ray of voxels $\{X'\}$. If $V$ is the unique voxel that causes the intensity value at the pixel, then the expected intensity at $I_X$ is explained by the following equations:

$$E(I_X) = \sum_{X' \in R} E(I_X \,|\, V = X') P(V = X') \tag{7}$$

$$= \sum_{X' \in R} E(I_X \,|\, V = X') P(X' \in S) \, P(X' \text{ is not occluded}). \tag{8}$$

For every ray containing a particular voxel $X'$, the quantity $E(I_x \,|\, V = X')P(X' \in S)$ is fixed, and the only ray-dependent term is $P(X' \text{ is not occluded})$. When learning neighborhood configurations in the PVM only the ray-independent information is taken into account, reducing the information at every voxel to the following equation:

$$E(I_x \,|\, V = X')P(X' \in S). \tag{9}$$

Here, the quantity in (9) is referred to as a voxel's expected appearance, and the volume of expected appearances, as the expectation volume model, EVM. This work proposes to use a voxel's expected appearance as the underlying information to be characterized. The motivation for this choice is that, by combining the appearance information with the surface probabilities, it is possible to detect structures that may lie on the same surface but have differing appearance, such as windows or doors. As another example, vehicles do not have significant height relief with respect to the 3-D ground plane but typically differ significantly from the background in appearance.

### A. PCA Features

One way to represent the volumetric model is by identifying local spatial configurations that account for most of the variation in the expected appearance data. Principal component analysis (PCA) is carried out to find the orthonormal basis that represents the volumetric samples in the best mean squared error sense. The principal components are arranged in decreasing order of variation as given by the eigenvalues of the sample scatter matrix.

In order to perform PCA, feature vectors are obtained by sampling locations on the scene according to the octree structure, i.e., fine sampling in regions near surfaces and sparse sampling of empty space. At each sampled location, $n_x\hat{l} \times n_y\hat{l} \times n_z\hat{l}$ cubical regions are extracted (centered at the sampled location), where $\hat{l}$ is the length of the smallest voxel present in the 3-D scene. The extracted regions are arranged into vectors by traversing the space at a resolution of $\hat{l}$, using a raster visitation schedule. The scatter matrix $\mathbf{S}$, of randomly sampled vectors, is updated using a parallel scheme [52] to speed up computation, and the principal components are found by the eigenvalue decomposition of $\mathbf{S}$. In the PCA space, every neighbor-

hood (represented by a $d$-dimensional feature vector $\mathbf{x}$) can be exactly expressed as $\mathbf{x} = \bar{\mathbf{x}} + \sum_{i=1}^{d} a_i \mathbf{e}_i$, where $\mathbf{e}_i$ are principal axes associated with the $d$ eigenvalues, and $a_i$ are the corresponding coefficients. A k-dimensional $(k < d)$ approximation of the neighborhoods can be obtained by using the first $k$ principal components i.e., $\tilde{\mathbf{x}} = \bar{\mathbf{x}} + \sum_{i=1}^{k} a_i \mathbf{e}_i$. Section VIII presents an analysis of the reconstruction error of local neighborhoods, namely $|\mathbf{x} - \tilde{\mathbf{x}}|^2$. In the remainder of this paper, the vector arrangement of projection coefficients in the PCA space is referred to as a *PCA feature*.

### B. Taylor Features

One advantage of the PCA features is that by learning the basis directly from the data no major assumptions about the local neighborhoods are imposed. However, learning the local features is not always convenient. As new information is introduced to the system, the PCA space may need to be relearned, making necessary to store past data.

A set of derivatives computed up to a given order can be used to approximate the expected appearance function in a neighborhood. The idea of using differential descriptors has been around for along time in the image analysis community [53]–[56]. Although distribution-based descriptors [1], [2] have become more popular, differential descriptors have been shown to be an effective alternative in view of the high-dimensionality of histogram-based descriptors [56]. This paper proposes the use of differential descriptors obtained from a Taylor series approximation of the local neighborhoods, whose performance is compared to that of PCA descriptors for neighborhood reconstruction and object classification tasks. It will be demonstrated that differential descriptors provide an efficient representation of local information, comparable to that of PCA vectors. Additionally, an evaluation of performance or Taylor features lays the groundwork for the development of distribution-based descriptors (based on the responses of differential operators) in future work in the PVM.

The computation of derivatives in the expectation volume model, EVM, can be expressed as a least square error minimization of the following energy function:

$$E = \sum_{i=-ni}^{ni} \sum_{j=-nj}^{nj} \sum_{k=-nk}^{nk} (V(i,j,k) - \tilde{V}(i,j,k))^2 \qquad (10)$$

where $\tilde{V}(i,j,k)$ is the Taylor series approximation of the expected 3-D appearance of a volume $V$ centered on the 3-D point $(i,j,k)$. Using the second degree Taylor expansion of $V$ about (0, 0, 0), (10) becomes

$$E = \sum_{\mathbf{x}} \left( V(\mathbf{x}) - V_0 - \mathbf{x}^{\mathbf{T}} \mathrm{G} - \frac{1}{2!} \mathbf{x}^{\mathbf{T}} \mathrm{H} \mathbf{x} \right)^2 \qquad (11)$$

where $V_0$, $\mathrm{G}, \mathrm{H}$ are the zeroth derivative, the gradient vector and the Hessian matrix of the volume of expected 3-D appearances about the point (0, 0, 0), respectively. The coefficients for 3-D derivative operators can be found by minimizing (11) with respect to the zeroth, first- and second-order derivatives, i.e., $V_0, V_x, V_z, V_y, V_{xx}, V_{yy}, V_{zz}, V_{xy}, V_{xz}, V_{yz}$. The computed derivative operators are applied algebraically to neighborhoods in the EVM. The responses to the ten Taylor operators are arranged into 10-dimensional vectors, here referred to as *Taylor features*.

## VI. Feature Detection

Building object representations from local features is a two-stage process. The first step is feature detection. During this step the objective is to locate repeatable neighborhoods that capture relevant information about the object. The second step is feature description, where a representation of the object properties are extracted at these stable positions. Developing features that are repeatable in spite of scene transformations and that carry sufficient information for recognition tasks is a key goal of this effort.

In this paper, the results of a set of object categorization experiments are presented. During the first part of the experiments the detection process is avoided by describing local features in a dense manner. Features are described using either the PCA or Taylor coefficients as outlined in the previous section. The second part of experiments studies the performance of features detected using two other popular neighborhood operators, namely the Harris corner detector and the determinant of the Hessian. For these experiments, neighborhoods are represented by Taylor descriptors as they are faster to compute and are shown to have similar performance to PCA descriptors.

### A. Harris Corner Features

Harris and Stephens [57] proposed a corner detector that finds positions in 2-D images where the intensity function varies in more than one direction. The detector is based on the local average of the second moment matrix, i.e.,

$$M = \text{local weighted mean of} \begin{bmatrix} \left(\frac{\partial I}{\partial x}\right)^2 & \left(\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}\right) \\ \left(\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}\right) & \left(\frac{\partial I}{\partial y}\right)^2 \end{bmatrix}. \qquad (12)$$

The eigenvalues, $\lambda_1$ and $\lambda_2$, of $M$ constitute descriptors of variations along the two image directions. In particular, the presence of a corner feature can be identified when both eigenvalues are large. Instead of performing explicit computation of the eigenvalues, Harris and Stephens used the trace and determinant of $M$ to define the following corner response measure:

$$\mathrm{R} = \det(M) - \kappa \operatorname{trace}^2(M). \qquad (13)$$

The parameter $\kappa$ is chosen based on the desired ratio between the eigenvalues, i.e., $\kappa = \alpha/(\alpha+1)^2$, and $\alpha = \lambda_2/\lambda_1$. In image-based applications, $\alpha$ values between 10 and 20 have been suggested [1]. Large positive responses of R indicate the presence of corners, while large negative responses indicate the presence of edge features.

Laptev [58] introduced a generalization of the Harris corner detector to find space-time interest points for video categorization. The corner response measure proposed by Laptev is given by

$$R_{\mathrm{Harris}} = \det(M) - \kappa \operatorname{trace}^3(M) \qquad (14)$$

where the second moment matrix, $M$, in 3-Dimensions is given by the local weighted mean of

$$\begin{bmatrix} \left(\frac{\partial V}{\partial x}\right)^2 & \left(\frac{\partial V}{\partial x}\frac{\partial V}{\partial y}\right) & \left(\frac{\partial V}{\partial x}\frac{\partial V}{\partial z}\right) \\ \left(\frac{\partial V}{\partial x}\frac{\partial V}{\partial y}\right) & \left(\frac{\partial V}{\partial y}\right)^2 & \left(\frac{\partial V}{\partial y}\frac{\partial V}{\partial z}\right) \\ \left(\frac{\partial V}{\partial x}\frac{\partial V}{\partial z}\right) & \left(\frac{\partial V}{\partial y}\frac{\partial V}{\partial z}\right) & \left(\frac{\partial V}{\partial z}\right)^2 \end{bmatrix}. \quad (15)$$

Corner features satisfying $R \geq 0$, have $\kappa \leq \alpha\beta/(\alpha+\beta+1)^3$, where $\alpha = \lambda_2/\lambda_1$ and $\beta = \lambda_3/\lambda_1$. In this work, all components of $M$ are computed using the differential operators obtained from a Taylor series approximation as explained in the previous section and various values of $\kappa$ are evaluated.

### B. Hessian Features

Similar to the Harris corner detector, another differential approach for detecting interest points is the determinant of the Hessian matrix, i.e., $R_{DoH} = \det(H)$. This detector was first introduced by Beaudet [59] and recently used as the basic interest-point detector in the SURF descriptor [36]. A recent evaluation of volumetric interest point detectors [31], suggests that for 3-D shapes the Hessian detector performs better than the Harris corner detector. In this work, object recognition experiments were performed using features that maximize $R_{DoH}$. The recognition accuracy is compared to that of Harris-based features, and the descriptive power for different categories is analyzed. The components of the Hessian matrix are computed using the differential operators obtained from a Taylor series approximation as explained in the previous section.

## VII. 3-D Object Learning and Recognition

### A. The Model: Bag of Volumetric Features

Bag-of-features models have their origins in texture recognition [60], [61] and bag-of-word representations for text categorization [62]. Their application to categorization of visual data has been studied extensively [63], [64], and have produced impressive results in recognition benchmark datasets [65]. The independence assumptions inherent to bag-of-features representation make learning models for few object categories a simple task, assuming enough training samples are available to learn the classification space. Taking advantage of the simplicity of the method and inspired by the success in the computer vision and 3-D shape retrieval communities [23], this work represents objects as bags of volumetric features. The process is outlined in the following subsections.

### B. Learning a Volumetric Vocabulary With $k$-Means

In order to produce a finite dictionary of 3-D expected appearance patterns, the scenes are represented by a set of features, e.g., dense-PCA, dense-Taylor, Hessian-based or Harris-based, that are quantized using k-means clustering. Each mean represents a region of the feature space that contains a significant population of feature instances (clusters) from the objects of interest. Two major limitations of k-means clustering are: 1) the algorithm does not determine the best number of means, i.e., $k$; and 2) the algorithm often converges to a local minimum that may not represent the optimum placement of cluster centers.

To address 1), various values of $k$ were determined heuristically, and clustering performance was evaluated for the different values, leading to a suitable value. Regarding 2), based on the evaluation of different initialization methods reported by Maitra *et al.* [66], the algorithm proposed by Bradley and Fayyad [67] is chosen to initialize the means. In their algorithm a random set of subsamples of the data are chosen and clustered via modified k-means. The clustering solutions are again clustered using classical k-means, and the solution that minimizes the sum of square distances between the points and the centers is chosen as the initial set of means. In order to keep computation time manageable, while still choosing an appropriate number of subsamples (ten being suggested in [66] and [67]), an accelerated k-means algorithm [68] is used whenever the classical k-means procedure is required. After initializing the means, samples inside manually labeled bounding boxes of the objects were clustered using accelerated k-means [68] to find a common feature vocabulary.

### C. Learning and Classifying Object Categories

With a 3-D appearance vocabulary in place, individual objects are represented by feature vectors that indicate the volumetric vocabulary elements present in a given object instance. These feature vectors can be used in supervised multi-class learning, where a naive Bayes classifier is used for its simplicity and speed. During learning, the classifier is passed training objects used to adjust the decision boundaries; during classification, the class label with the maximum *a posteriori* probability is chosen to minimize the probability of error.

Formally, let the objects of a particular category be the set $\mathbf{O}_l = \bigcup_{i=1}^{N_l} \mathbf{o}_i$, where $l$ is the class label and $N_l$ is the number of objects with class label $l$. Then, the set of all labeled objects is defined as $\mathbf{O} = \bigcup_{l=1}^{N_c} \mathbf{O}_l$, where $N_c$ is the number of categories. Let the vocabulary of 3-D expected appearance patterns be defined as $\mathbf{V} = \bigcup_{i=1}^{k} \mathbf{v}_i$, where $k$ is the number of cluster centers in the vocabulary. From the quantization step a count is obtained, $c_{ij}$, of the number of times a cluster center, $\mathbf{v}_i$, occurs in object $\mathbf{o}_j$. Using Bayes formula, the *a posteriori* class probability is given by

$$P(C_l \,|\, \mathbf{o}_i) \propto P(\mathbf{o}_i \,|\, C_l) P(C_l). \quad (16)$$

The likelihood of an object is given by the product of the likelihoods of the independent entries of the vocabulary, $P(\mathbf{v}_j \,|\, C_l)$, which are estimated during learning. The full expression for the class posterior becomes

$$P(C_l \,|\, \mathbf{o}_i) \propto P(C_l) \prod_{j=1}^{k} P(\mathbf{v}_j \,|\, C_l)^{c_{ji}} \quad (17)$$

$$\propto P(C_l) \prod_{j=1}^{k} \left( \frac{\sum_{m=1:\mathbf{o}_m \in \mathbf{O}_l}^{N_m} c_{jm}}{\sum_{n=1}^{k} \sum_{m=1:\mathbf{o}_m \in \mathbf{O}_l}^{N_m} c_{nm}} \right)^{c_{ji}}. \quad (18)$$

According to the Bayes decision rule, every object is assigned the label of the class with the largest *a posteriori* probability. In practice, log likelihoods are computed to avoid underflow of floating point computations.

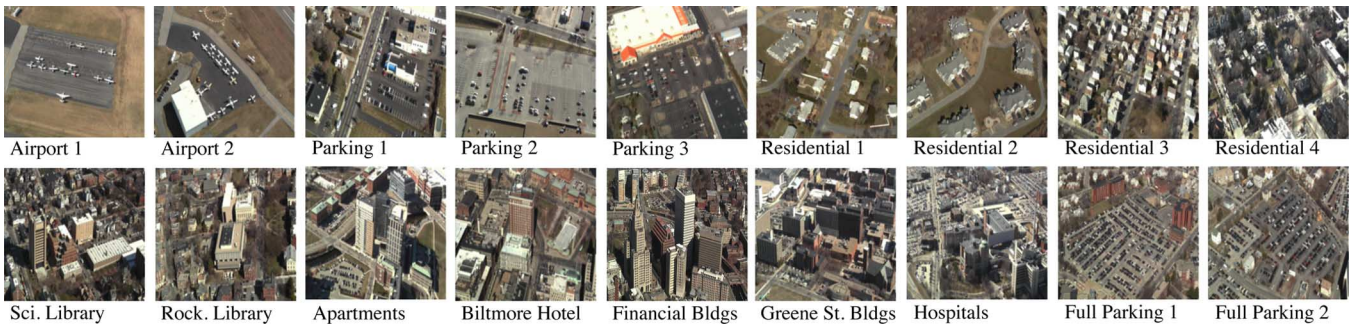



Fig. 5. Names and sample images of the 18 sites used in this work.

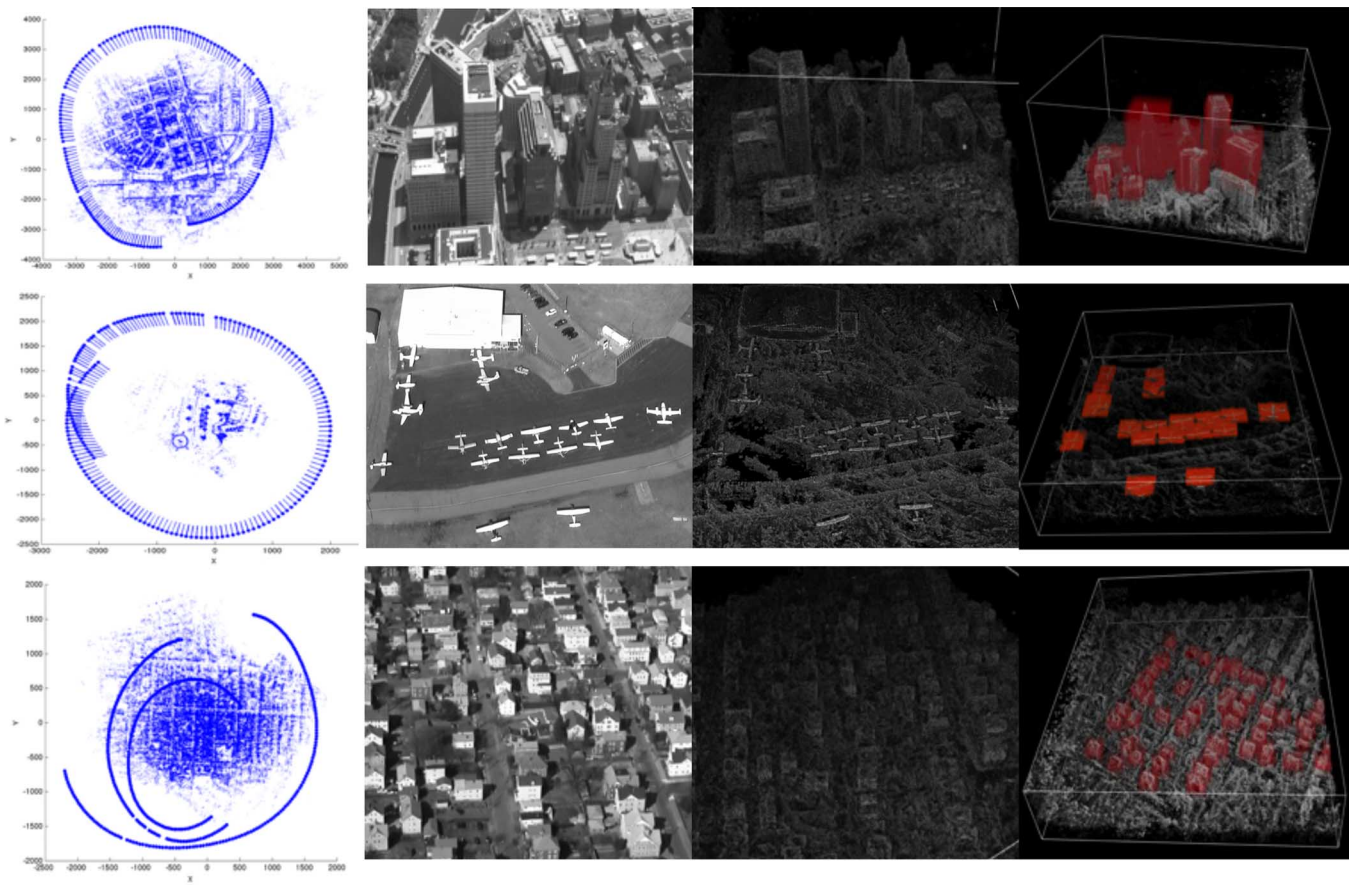


Fig. 6. From left to right (column by column): Camera path obtained using structure from motion algorithm [46]. Details of collected video frames. The learned expected appearance volumes, EVM. Examples of bounding boxes around objects of interest.

## VIII. Experiments and Results

The data collection and scene reconstruction processes are now described, followed by comparisons of scene data modeling accuracy based on either PCA and Taylor features. The section concludes with multi-class object recognition results, where test objects were classified among five categories; planes, cars, houses, buildings, and parking lots.

### A. Data Collection and Scene Formation

The aerial data used to build 18 different probabilistic volume scenes was collected from a helicopter flying over Providence, RI, and its surroundings; see Fig. 5 for a reference on site names and sample frames. The helicopter flew at an average height of between 300 and 450 meters. An approximate resolution of 30 cm/pixel is obtained in the imagery and matched by having the highest resolution voxels span 30 cm on a side in the 3-D models. The camera matrices for all image sequences were obtained using the Bundler software provided by Snavely *et al.* [46]. The probabilistic volume models were learned using the GPU implementation by Miller *et al.* [18]. Fig. 6 shows examples of camera-paths, aerial images and the corresponding expectation volume models. To carryout multi-class category learning, bounding boxes around objects of interest were manually constructed and assigned the corresponding class label. See Fig. 6 for examples of such bounding boxes.

The volumetric models shown in Fig. 6 present expected 3-D appearance of voxels, [see (9)], which ranges from [0, 2]. For empty space, the information in the voxels is dominated by the occupancy probability, which takes values in the interval [0, 1]; thus, empty neighborhoods appear black. Appearance values,

TABLE I
AVERAGE APPROXIMATION ERROR OVER ALL $5 \times 5 \times 5$ NEIGHBORHOODS. PCA AND TAYLOR ERRORS ARE COMPARED FOR EIGHT TEST SCENES

| Site (see Fig. 5) | PCA Error | Taylor Error |
|---|---|---|
| Airport 1 | 2.195 | 2.323 |
| Parking 3 | 3.241 | 3.407 |
| Residential 1 | 3.219 | 3.387 |
| Residential 4 | 5.687 | 5.949 |
| Rock. Library | 4.242 | 4.41 |
| Biltmore Hotel | 3.627 | 3.776 |
| Greene St. Bldgs | 3.25 | 3.389 |
| Full Parking 1 | 4.745 | 4.973 |
| **Average** | **3.781** | **3.952** |

which are initially learned between [0, 1], are offset to [1, 2], to avoid confusing dark surfaces with empty space. White voxels represent white surfaces with a high occupancy probability; dark surfaces are represented by gray voxels, with a value near one.

### *B. Neighborhood Reconstruction Error*

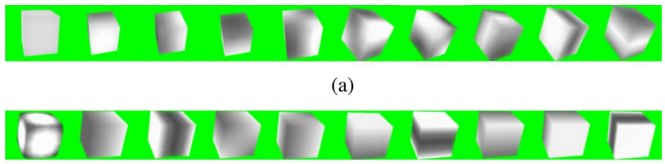
This section presents the data modeling error achieved using PCA and Taylor-based features. Ideally, the difference between the original expected appearance data and the data approximated using PCA or a Taylor series expansion should be small. The difference between the reconstructed data and the original data was measured as the average square difference between neighborhoods, i.e., $(1)/(N)\sum_{i=1}^{N}|\mathbf{x}-\hat{\mathbf{x}}|^2$, where $\mathbf{x}$ and $\hat{\mathbf{x}}$ are the vector scans of the original and the approximation neighborhoods, respectively. $N$ is the number of samples used to compute the error. In the experiments, the size of the extracted neighborhoods is $5\hat{l}\times 5\hat{l}\times 5\hat{l}$, $\hat{l}$ being the length of the smallest voxel in the model.

The reconstruction error for a 10-dimensional approximation in the PCA space was compared to the reconstruction error achieved using a 2nd-degree Taylor approximation. The error was evaluated over eight scenes (for reference on site names please see Fig. 5) and the results are reported in Table I. On average, and for every test scene, the results indicate that a second-degree Taylor approximation represents expected appearance of 3-D patterns with slightly less accuracy than a PCA projection onto a 10-dimensional space. The PCA basis was learned using random samples from the remaining of the scenes (those not used for testing).

### *C. 3-D Object Recognition Using Dense-Feature Models*

This section presents multi-class object recognition results achieved by bag-of-features models where objects were described in a dense manner by either PCA or Taylor descriptors. These descriptors were clustered through k-means to form the volumetric vocabulary. Basis kernels for PCA and Taylor are shown in Fig. 7. The Taylor basis is the same for all validation sets. The PCA basis was recomputed for each validation set using all available samples in the training objects. Only the first ten principal components were retained to form the feature descriptors.

With a feature vocabulary in place, the models for five object categories (planes, cars, buildings, houses, and parking lots) were trained using all available $5\hat{l}\times 5\hat{l}\times 5\hat{l}$ neighborhoods centered on leaf cells that met the following criteria: 1) the leaf cell is at the finest level of the octree; 2) the leaf cell is contained within the corresponding bounding box of the object of interest. It is worth noting that the bounding boxes were not necessarily tight around the objects, and due to constraints in the labeling method, all boxes were axis-aligned.



(a)

(b)

Fig. 7. (a) *PCA kernels*, i.e., volumetric representation of the first ten principal components. Note that these kernels are learned from training objects. (b) *Taylor kernels*.

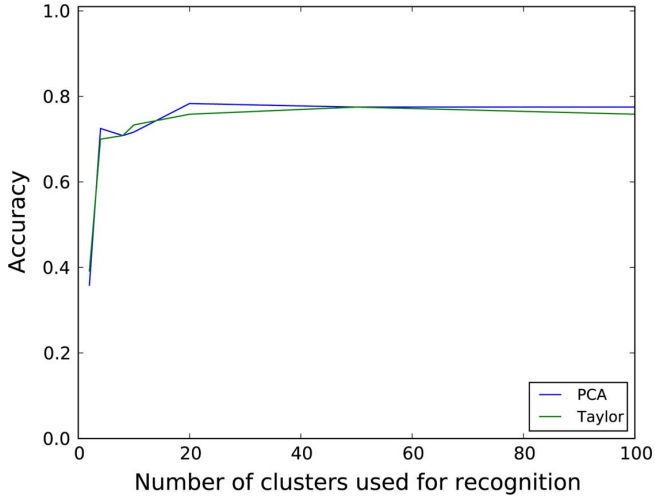



Fig. 8. Classification accuracy for dense Taylor and PCA-based models. The curves represent the fraction of correctly classified objects as a function of the number of clusters.

TABLE II
NUMBER OF TRAINING AND TESTING OBJECT INSTANCES IN EACH CATEGORY ACROSS TEN TRIALS

| Set | | Planes | Cars | Houses | Buildings | Parking Lots |
|---|---|---|---|---|---|---|
| Trial 0 | Train | 16 | 37 | 48 | 24 | 16 |
| | Test | 16 | 46 | 58 | 20 | 20 |
| Trial 1 | Train | 11 | 33 | 57 | 15 | 23 |
| | Test | 21 | 50 | 49 | 24 | 21 |
| Trial 2 | Train | 12 | 37 | 51 | 18 | 19 |
| | Test | 20 | 46 | 55 | 21 | 25 |
| Trial 3 | Train | 11 | 39 | 54 | 21 | 23 |
| | Test | 21 | 44 | 52 | 18 | 21 |
| Trial 4 | Train | 18 | 42 | 51 | 19 | 21 |
| | Test | 14 | 41 | 55 | 20 | 23 |
| Trial 5 | Train | 16 | 42 | 48 | 21 | 24 |
| | Test | 16 | 41 | 58 | 18 | 20 |
| Trial 6 | Train | 18 | 47 | 50 | 23 | 23 |
| | Test | 14 | 36 | 56 | 16 | 21 |
| Trial 7 | Train | 15 | 47 | 53 | 22 | 28 |
| | Test | 17 | 36 | 53 | 17 | 16 |
| Trial 8 | Train | 12 | 44 | 55 | 20 | 21 |
| | Test | 20 | 39 | 51 | 19 | 23 |
| Trial 9 | Train | 13 | 43 | 49 | 16 | 25 |
| | Test | 19 | 40 | 57 | 23 | 19 |

To find an appropriate vocabulary size, classification results were evaluated while varying the number of clusters in the codebook from $k = 2$ to $k = 100$. Fig. 8 presents classification accuracy, i.e., the fraction of correctly classified objects, as a function of the number of clusters, i.e., number of entries in the volumetric vocabulary. For both Taylor-based and PCA-based features, the performance improves rapidly up to a 20-word

TABLE III
SUMMARY OF CLASSIFICATION ACCURACY FOR DENSE TAYLOR-BASED MODELS

| Trial → | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | Mean | Std. Dev. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Planes | 0.875 | 0.9048 | 0.95 | 0.9524 | 0.9286 | 0.9375 | 1 | 0.8824 | 0.85 | 1 | 0.9281 | 2.3054e-03 |
| Houses | 0.7414 | 0.7755 | 0.6909 | 0.7885 | 0.7273 | 0.6724 | 0.7857 | 0.7547 | 0.7843 | 0.7719 | 0.7493 | 1.5183e-03 |
| Buildings | 0.75 | 0.6667 | 0.7619 | 0.6667 | 0.75 | 0.7778 | 0.75 | 0.7647 | 0.6842 | 0.7826 | 0.7355 | 1.8294e-03 |
| Cars | 0.9348 | 0.96 | 0.9348 | 0.8864 | 0.9512 | 0.9512 | 0.8889 | 0.9444 | 0.9487 | 0.875 | 0.9275 | 8.9641e-04 |
| Parking Lots | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| Overall | 0.8602 | 0.8614 | 0.8675 | 0.8588 | 0.8714 | 0.8678 | 0.8849 | 0.8692 | 0.8534 | 0.8859 | 0.8681 | 0.1121 |

TABLE IV
SUMMARY OF CLASSIFICATION ACCURACY FOR DENSE PCA-BASED MODELS

| Trial → | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | Mean | Std. Dev. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Planes | 0.875 | 0.9048 | 0.9 | 0.9524 | 0.9286 | 0.9375 | 1 | 0.8824 | 0.75 | 1 | 0.9131 | 0.0047 |
| Houses | 0.7759 | 0.6939 | 0.6909 | 0.7308 | 0.6909 | 0.6207 | 0.8393 | 0.717 | 0.6863 | 0.7018 | 0.7147 | 0.0031 |
| Buildings | 0.75 | 0.7083 | 0.8095 | 0.7222 | 0.75 | 0.7778 | 0.75 | 0.7647 | 0.7368 | 0.7826 | 0.7552 | 0.0008 |
| Cars | 0.8913 | 0.9 | 0.8696 | 0.9545 | 0.9024 | 0.9024 | 0.8333 | 0.8889 | 0.8974 | 0.875 | 0.8915 | 0.0008 |
| Parking Lots | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.9375 | 1 | 1 | 0.9938 | 0.0004 |
| Overall | 0.8584 | 0.8414 | 0.854 | 0.872 | 0.8544 | 0.8477 | 0.8845 | 0.8381 | 0.8141 | 0.8719 | 0.8536 | 0.1126 |

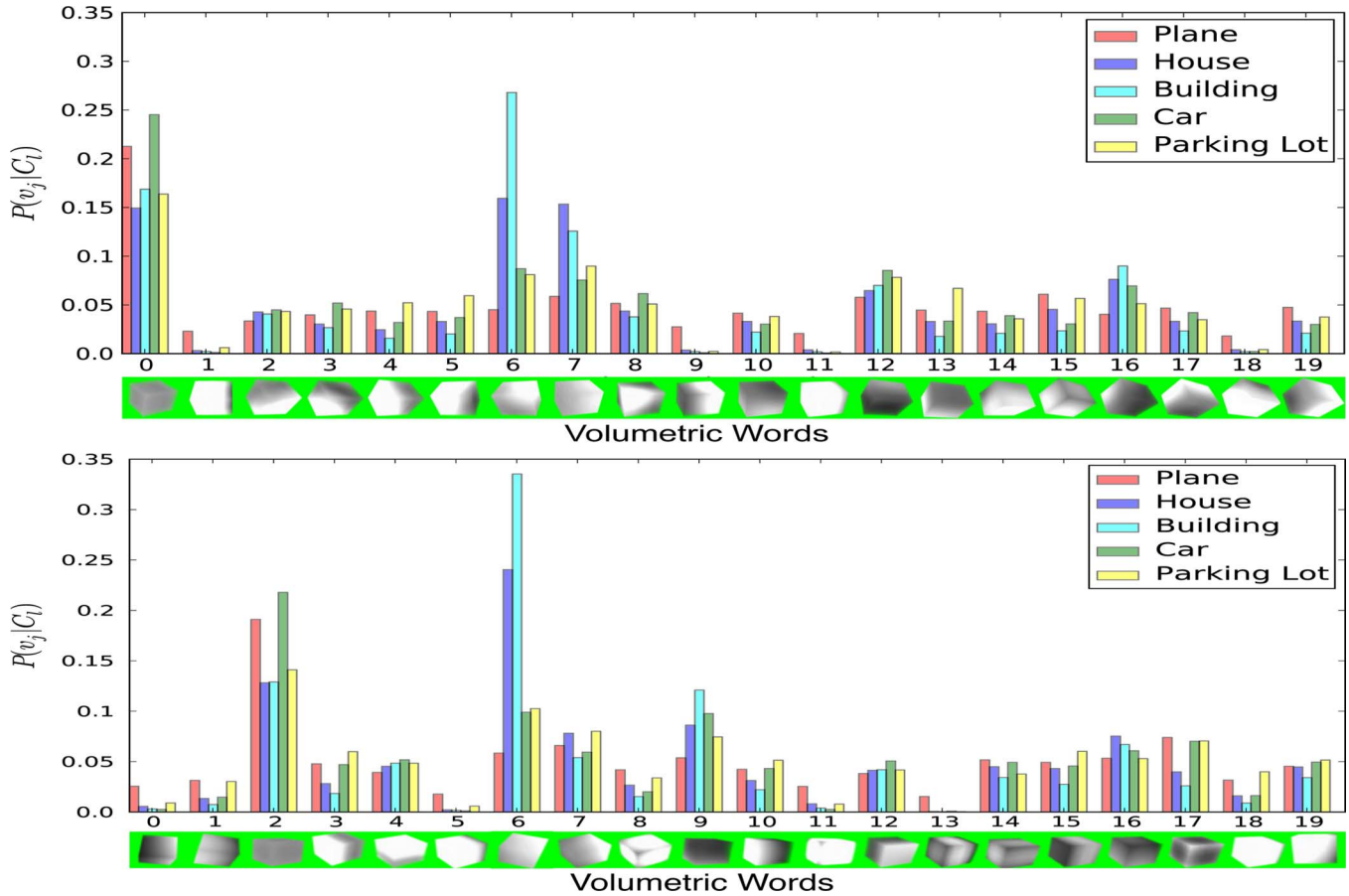



Fig. 9. Class histograms for the vocabulary that achieved best performance across ten validation sets. The top row corresponds to class representations learned with dense PCA-based features. The bottom row corresponds to those learned with dense Taylor-based features. The x-axis shows the volumetric form of the 20 volumetric words. The $y$-axis corresponds to the probability of each word given the class label (i.e., frequency).

codebook, with little or no improvement for larger vocabularies. Thus, for the remaining of the experiments the number of vocabulary entries was set to 20.

For the rest of the experiments, two measurements were used to evaluate the classification performance: classifier accuracy and the confusion matrix. Every object in the data set was randomly assigned to the training or testing set. This process was repeated ten times to form different validation sets. The number of objects in the different data splits are reported in Table II.

Tables III and IV present the classification accuracy for PCA and Taylor-based features. The results are reported for each validation set, as well as the corresponding mean and standard deviation. Both methods recognize planes, cars, and parking lots with high accuracy. The accuracy of the Taylor-based representation is slightly higher across all categories.

Fig. 9 presents an example of the class distributions learned with PCA and Taylor codebooks of 20 features. To facilitate interpretation, the volumetric form of the vocabulary entries are


(a) (b)

Fig. 10. Confusion matrix for a 20-keyword codebook of PCA based features on the left and Taylor based features on the right. The values reported are the average over 10 trials. (a) PCA, (b) Taylor.

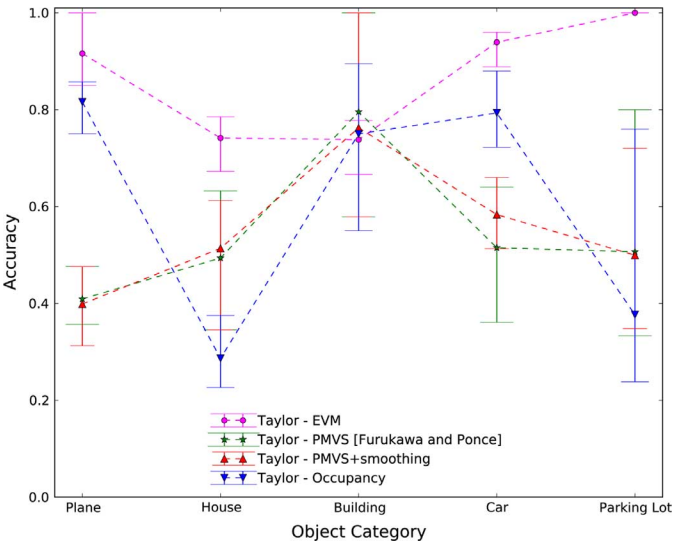



Fig. 11. Classification accuracy of different models based on the volume of occupancy probabilities, the EVM and the volume based on the PMVS [12] output. The error bars span the maximum and minimum accuracy over five trials. The mean is represented by the circular markers.

arranged along the $x$-axis. The $y$-axis indicates the frequency of the volumetric words for each object category. Keep in mind that the expected 3-D appearance at each voxel ranges from [0, 2] (black to white). Empty neighborhoods appear black, white voxels represent white surfaces with a high occupancy probability and dark surfaces are represented by gray voxels.

Finally, the confusion matrices for PCA-based features and Taylor-based features are shown in Fig. 10(a) and (b). The confusion matrices indicate that both methods often confuse houses and buildings. During Bayesian learning, the likelihood of a vocabulary entry is normalized with respect to the total number of features in the object. The models learned for houses and buildings are very similar and the scale difference between instances is not captured by the normalized bag of features representation.

### D. Effectiveness of the EVM

This work proposes to use appearance information in addition to occupancy probabilities to achieve better classification performance. The effectiveness of this approach is demonstrated experimentally. Fig. 11 presents comparisons of classification accuracy for models learned using the EVM and those learned using the volume of occupancy probabilities (no appearance). The results are reported over 10 splits of the test/train data. Taylor descriptors were used during these experiments since their basis does not have to be re-learned for every validation set. Furthermore, in the previous section Taylor descriptors achieved slightly higher accuracy than PCA-based descriptors. For every category, the average accuracy is labeled by a circular marker in Fig. 11. The error bars span the maximum and minimum accuracy attained in the ten trials. Several conclusions can be drawn from these experiments. It is not possible to reliably recognize parking lots using only occupancy information. This result is likely due to the fact that parking lots are generally featureless, making it difficult to form accurate surface geometry. However, the appearance information allows the system to build models for parking lots that are consistently different from the other categories. Planes and cars also exhibited lower recognition performance when using only occupancy information. On average, buildings were recognized with very similar accuracy, with or without appearance. Finally, classification accuracy exhibited smaller variance when using the EVM than with occupancy alone.

### E. Effectiveness of Probabilistic Volumetric Reconstruction

To demonstrate the advantages of probabilistic learning compared to a threshold-based 3-D reconstruction framework, the categorization algorithm was run on scenes obtained using a state-of-the-art, point cloud based, dense 3-D reconstruction algorithm, PMVS [12]. In order to apply the proposed classification algorithm to the scenes reconstructed using PMVS, the output point clouds were *voxelized*. Two types of *voxelization* processes were tested: 1) the intensity at each point in the point cloud was stored at a leaf of the finest resolution in the octree; 2) a Gaussian kernel was applied to the volumes achieved using 1). The width of the Gaussian kernel was chosen to match the width of the Taylor-kernel. Fig. 12 presents the EVM and the volumes recovered from PMVS using Gaussian smoothing. By inspecting Fig. 12, it is apparent that the information in the EVM is much denser than in the PMVS-based volume. The method proposed by Furukawa and Ponce [12] is not able to recover information in many rooftops and streets, likely due to the absence of surface texture or other sources of distinct image appearance.

The object categorization results based on the scenes obtained using PMVS are reported in Fig. 11. The results obtained from the *voxelized* model with Gaussian smoothing are very similar to those without smoothing. The classification accuracy is significantly lower than the accuracy obtained for the EVM, except for the building category. Another observation is that the results obtained across the ten splits of the data are more stable for the EVM than the PMVS-based models, as indicated by the error bars.

### F. 3-D Object Recognition Using Sparse-Feature Models

This section presents object categorization results for models learned with Taylor features that were filtered based on their saliency using: 1) the 3-D extension of Harris corner measure in (14) 2) the determinant of the Hessian as explained in Section VI-B. Fig. 13 summarizes the classification accuracy for different saliency criteria. The blue, green, and magenta curves report the results for Harris-based features. The results

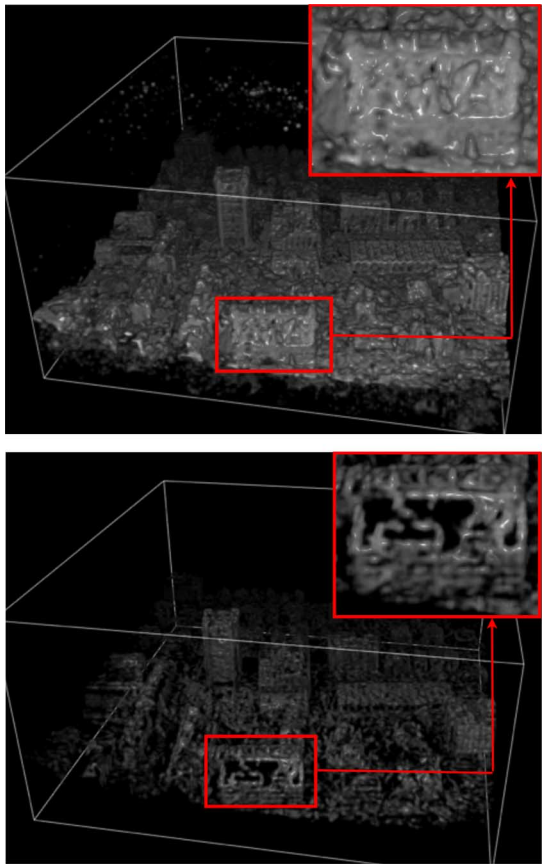


Fig. 12. Top: The expected appearance volume model (EVM) for a sample scene. Bottom: The volumetric model of the same scene obtained using PMVS [12] and Gaussian smoothing. The zoomed-in details show an example of a roof where due to appearance ambiguities, PMVS cannot recover the geometry. More information about the appearance and geometry of the roof is present when probabilistic learning was used.

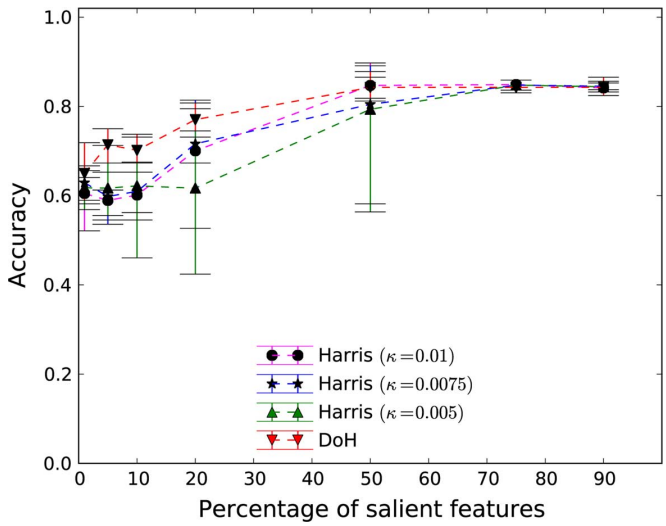



Fig. 13. Classification accuracy for models learned using sparse Taylor features that maximize the 3-D Harris corner and determinant of hessian measures. The curves represent the fraction of correctly classified objects as a function of the percentage of features retained. Accuracy results are reported for values at 1%, 5%, 10%, 20%, 50%, 75%, and 90%. The error bars span the maximum and minimum accuracy over five trials. The mean is represented by the circular markers.

are compared for three different values of the curvature parameter $\kappa : 0.005, 0.0075$ and $0.01$. This parameter corresponds to values of $\alpha$ and $\beta$ curvature ratios equal to twenty three, fifteen and ten, respectively. The $x$-axis of the accuracy plot corresponds to the to $p$-percent of corners retained per object. The accuracy plot in Fig. 13 reveals that for stable (see error bars) and accurate classification performance, large number of corner features are necessary (at least 50%). Classification accuracy increases as the number of features increases. On average, the three curvature parameters led to similar performance, except at 20% where $\kappa = 0.005$ had inferior performance. However, the variance in accuracy was lower as the curvature parameter, $\kappa$, in (14) was increased, i.e., corners become less elongated. This is expected since localization of edge features is generally less consistent. The red curve in Fig. 13 reports the results when retaining the determinant of Hessian-based features. For most tested percentage values, these features achieved higher accuracy and smaller variance that Harris-based features.

Fig. 14 presents the average confusion matrices over five trials for various combinations of saliency measure and percentage of features retained. Small percentages of salient features are able to classify planes and parking lots with satisfactory rates. To achieve recognition rates above 0.7 for both buildings and houses, at least the top 50% of the salient features need to be retained. For percentages below 50%, the accuracy obtained for houses, buildings, and cars appears unstable.

Fig. 15(a), (b), and (c) present running times of various stages of the training and testing process as a function of the percentage of features retained. Running times are reported for a computer using two 2.93 GHz, Quad-Core Intel Xeon processors, where the algorithms were run on multiple threads. It is important to mention that running times of multi-threaded tasks are affected by the availability of cores and system locks, and that running times were not optimized to factor out these waiting times. Fig. 15(a) reports the running time (in seconds) during vocabulary learning. During this step, computation time is dominated by the complexity of the k-means clustering algorithm and reducing the number of features leads to significantly shorter running times. During learning of object categories [see Fig. 15(b)] and during classification [see Fig. 15(c)] running times are very similar for all percentages. Although, shorter times are expected for decreasing number of features, one possible reason for the observed results is that running times are dominated by disk I/O operations.

## IX. Conclusion and Further Work

This paper presents a completely new representation for object recognition models, where features are extracted directly from 3-D probabilistic information. The representation is used to learn and categorize objects from five different categories. To the authors' knowledge, this work represents the first attempt to apply this representation to the classification of aerial scenes or indeed any type of scene, making a contribution towards the understanding of realistic 3-D scenes.

The performance of the proposed features, was rigorously tested through reconstruction accuracy and object categorization experiments. The recognition results are very encouraging with high accuracy on labeling bounded regions containing

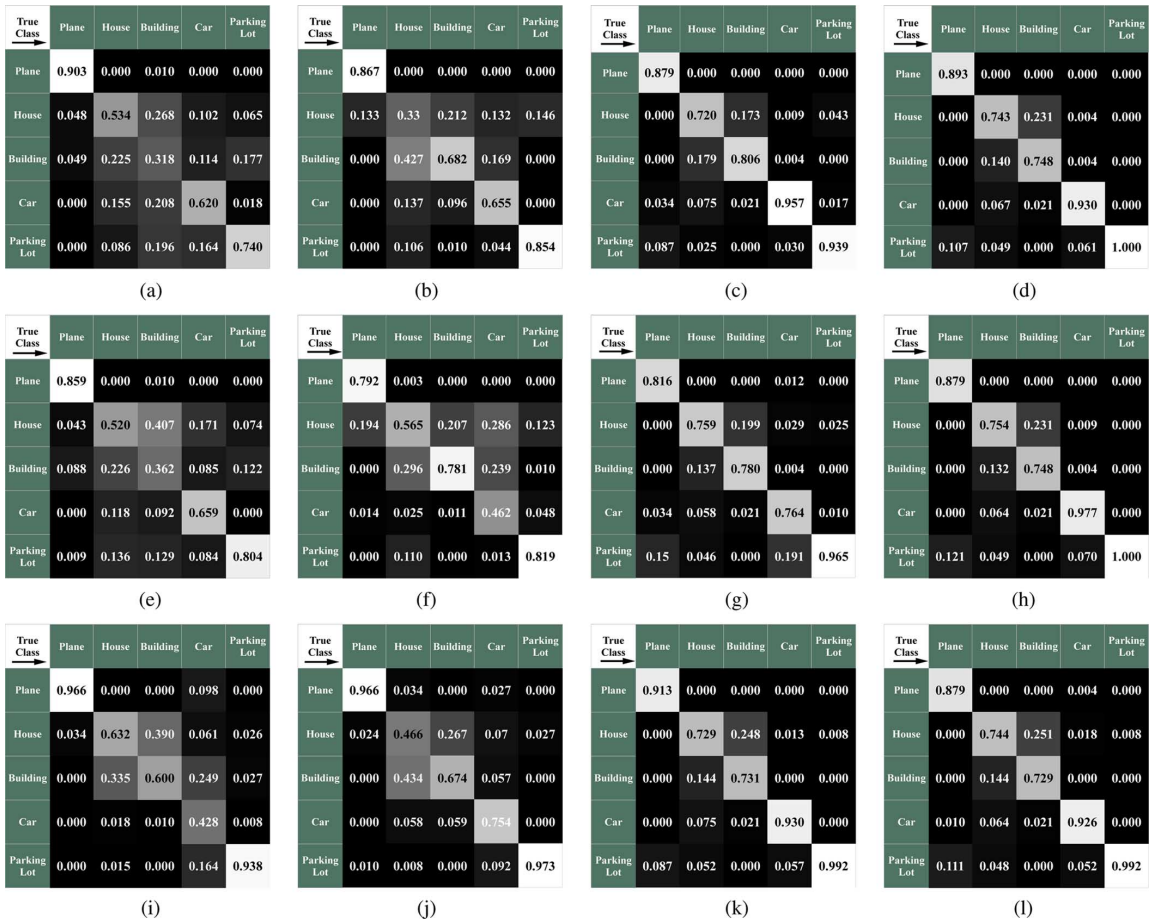



Fig. 14. Confusion matrices for various sparse-feature models. These models use a saliency criterion to filter the dense Taylor features. (a)–(d) Using the Harris corner measure with $\kappa = 0.01$ and varying the percentage of features used. (e)–(h) Using the Harris corner measure with $\kappa = 0.005$ and varying the percentage of features used. (i)–(l) Using the determinant of the Hessian criterion and varying the percentage of features used. (a) $\kappa = 0.01, p = 1\%$, (b) $\kappa = 0.01, p = 10\%$, (c) $\kappa = 0.01, p = 50\%$, (d) $\kappa = 0.01, p = 75\%$, (e) $\kappa = 0.005, p = 1\%$, (f) $\kappa = 0.005, p = 10\%$, (g) $\kappa = 0.005, p = 50\%$, (h) $\kappa = 0.005, p = 75\%$, (i) $\det(H), p = 1\%$, (j) $\det(H), p = 10\%$, (k) $\det(H), p = 50\%$, (l) $\det(H), p = 75\%$.

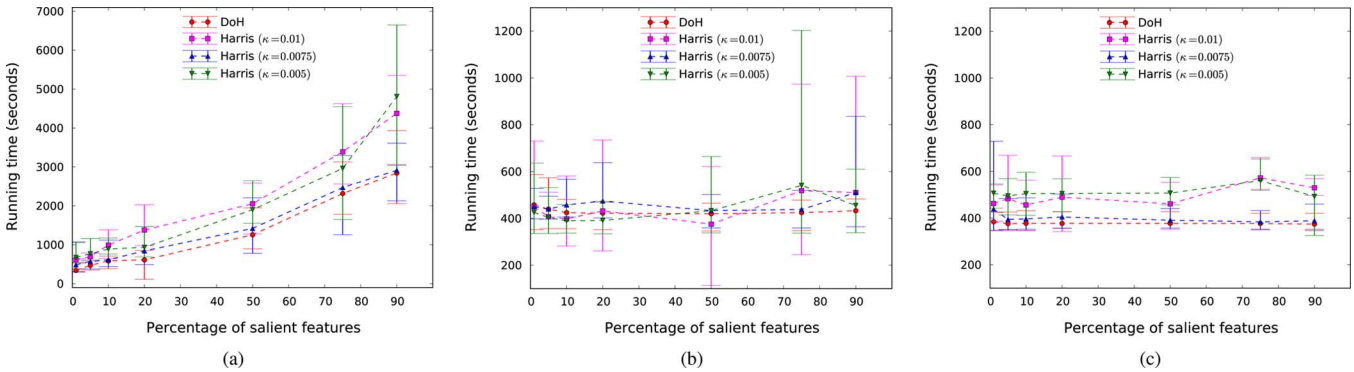



Fig. 15. Running times as a function of the percentage of features retained. (a) Running times for codebook learning (k-means clustering). (b) Running times measured during learning of object categories (quantization). (c) Classification running times (quantization + posterior computation). All figures are reported for experiments using the top 1%, 5%, 10%, 20%, 50%, 75%, and 90% of the features. The error bars span the maximum and minimum values obtained over five trials. The mean is represented by the circular markers.

objects of the selected categories. The experiments show that differential geometry features derived from appearance lead to essentially the same recognition performance as PCA. This suggests that additional features representing geometric relationships defined on differential geometry are likely to have good performance and represent a basis for formally extending the current feature vocabulary.

It was demonstrated that through probabilistic volumetric learning, it is possible to recover 3-D information more densely than through frameworks that are committed to forming an explicit geometry such as a point cloud. Specifically, the object categorization performance was shown to be superior for the EVM than for the volume based on the point cloud output of PMVS [12]. The categorization results also demonstrated the

superiority of models that combined appearance and geometry information over a models based on occupancy alone.

The overall accuracy of dense-feature representations was superior than that obtained using small percentages of salient features based on the Harris corner measure or the determinant of the Hessian. However, for few object categories, salient features demonstrated the ability to reduce the complexity of the feature space without sacrificing recognition performance. Across different validation sets, the results for Hessian-based features were more stable than for Harris-based features.

Future work will explore representations for rotation-invariant features. Localization of objects is also a desirable goal for future research. Finally, more advanced recognition models should make full use of the geometric relations inherent in the probabilistic volume model. Compositional recognition models can provide a way to learn and share parts, allowing for object representations that are efficient, discriminative and geometrically coherent.

## ACKNOWLEDGMENT

The authors would like to thank all the reviewers for their insightful and detailed suggestions.

## REFERENCES

[1] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, pp. 91–110, 2004.

[2] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 886–893.

[3] A. Golovinskiy, V. Kim, and T. Funkhouser, "Shape-based recognition of 3D point clouds in urban environments," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 2154–2161.

[4] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik, "Recognizing objects in range data using regional point descriptors," in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 224–237.

[5] T. Korah, S. Medasani, and Y. Owechko, "Strip histogram grid for efficient LIDAR segmentation from urban environments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2011, pp. 74–81.

[6] A. Patterson and P. Mordohai, "Object detection from large-scale 3D datasets using bottom-up and top-down descriptors," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 553–566.

[7] D. Crispell, J. Mundy, and G. Taubin, "Parallax-free registration of aerial video," in *Proc. British Mach. Vis. Conf.*, 2008, pp. 73.1–73.10.

[8] T. Pollard and J. Mundy, "Change detection in a 3-D world," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–6.

[9] J. L. Mundy and O. C. Ozcanli, "Uncertain geometry: A new approach to modeling for recognition," in *Proc. SPIE Defense, Security, Sens. Conf.*, 2009.

[10] O. Özcanli and J. Mundy, "Vehicle recognition as changes in satellite imagery," in *Proc. Int. Conf. Pattern Recognit.*, 2010, pp. 3336–3339.

[11] M. I. Restrepo, B. A. Mayer, and J. L. Mundy, "Object recognition in probabilistic 3-D volumetric scenes," in *Proc. Int. Conf. Pattern Recogni. Applicat. Meth.*, 2012, pp. 180–190.

[12] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 8, pp. 1362–1376, Aug. 2010.

[13] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[14] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiel, and L. Van Gool, "Towards multi-view object class detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2006, pp. 1589–1596.

[15] J. Knopp, M. Prasad, and L. Gool, "Scene cut: Class-specific object detection and segmentation in 3D scenes," in *Proc. Int. Conf. 3D Imaging, Model., Process., Visualiz. Transmiss.*, 2011, pp. 180–187.

[16] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. Van Gool, "Hough transform and 3D surf for robust three dimensional classification," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 589–602.

[17] D. Crispell, J. Mundy, and G. Taubin, "A variable-resolution probabilistic three-dimensional model for change detection," *IEEE Trans. Geosci. Remote Sens.*, pp. 1–12, 2012.

[18] A. Miller, V. Jain, and J. Mundy, "Real-time rendering and dynamic updating of 3-D volumetric data," in *Workshop General Purpose Process. Graphics Process. Units*, 2011.

[19] P. Papadakis, I. Pratikakis, and T. Theoharis, "PANORAMA: A 3D shape descriptor based on panoramic views for unsupervised 3D object retrieval," *Int. J. Comput. Vis.*, pp. 177–192, 2010.

[20] L. Shapira, S. Shalom, A. Shamir, D. Cohen-Or, and H. Zhang, "Contextual part analogies in 3D objects," *Int. J. Comput. Vis.*, pp. 309–326, 2010.

[21] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: Efficient and robust 3D object recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 998–1005.

[22] P. Bariya and K. Nishino, "Scale-hierarchical 3D object recognition in cluttered scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1584–1601.

[23] A. M. Bronstein, M. M. Broinstein, L. J. Guibas, and M. Ovsjanikov, "Shape Google: Geometric words and expressions for invariant shape retrieval," *ACM Trans. Graphics*, vol. 30, no. 1, pp. 1:1–1:20, Feb. 2011.

[24] D. Saupe and D. V. Vrani, "3D model retrieval with spherical harmonics and moments," in *Proc. DAGM-Symp. Pattern Recognit.*, 2001.

[25] A. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.

[26] R. Toldo, U. Castellani, and A. Fusiello, "A bag of words approach for 3D object categorization," in *Proc. Int. Conf. Comput. Vis./Comput. Graph. Collab. Tech.*, 2009, pp. 116–127.

[27] W. Cheung and G. Hamarneh, "N-SIFT: N-dimensional scale invariant feature transform for matching medical images," in *Proc. 4th IEEE Int. Symp. Biomed. Imag.: From Nano to Macro*, 2007, pp. 720–723.

[28] G. Flitton, T. Breckon, and N. Megherbi, "Object recognition using 3D SIFT in complex CT volumes," in *Proc. British Mach. Vision Conf.*, 2010, pp. 11.1–11.12.

[29] P. Scovanner, S. Ali, and M. Shah, "A 3-Dimensional SIFT descriptor and its application to action recognition," in *Proc. 15th Int. Conf. Multimedia*, 2007, p. 357.

[30] D. Raviv, M. M. Bronstein, A. M. Bronstein, and R. Kimmel, "Volumetric heat kernel signatures," in *Proc. ACM Workshop 3D Object Retrieval*, 2010, pp. 39–44.

[31] T.-H. Yu, O. J. Woodford, and R. Cipolla, "An evaluation of volumetric interest points," in *Proc. Int. Conf. 3D Imaging, Model., Process., Visualiz. Transmiss.*, 2011, pp. 282–289.

[32] J. Tangelder and R. Veltkamp, "A survey of content based 3D shape retrieval methods," *Shape Modeling Applicat.*, pp. 145–156, 2004.

[33] C. Zhang and T. Chen, "Efficient feature extraction for 2D/3D objects in mesh representation," in *Proc. Int. Conf. Image Process.*, 2001, pp. 935–938.

[34] M. Kazhdan, B. Chazelle, D. Dobkin, and T. Funkhouser, "A reflective symmetry descriptor for 3D models," *Algorithmica*, pp. 201–225, 2003.

[35] M. Novotni and R. Klein, "3D Zernike descriptors for content based shape retrieval," in *Proc. ACM Symp. Solid Model. Applicat.*, 2003, p. 216.

[36] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded-up robust features," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 346–359.

[37] E. Tola, V. Lepetit, and P. Fua, "DAISY: An efficient dense descriptor applied to wide-baseline stereo," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 815–830, May 2010.

[38] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[39] M. Körtgen, G. Park, and M. Novotni, "3D shape matching with 3D shape contexts," in *Proc. 7th Central Eur. Seminar Comput. Graph.*, 2003.

[40] N. J. Mitra, L. Guibas, J. Giesen, and M. Pauly, "Probabilistic fingerprints for shapes," in *PROC. Eurographics Symp. Geometry Process.*, 2006.

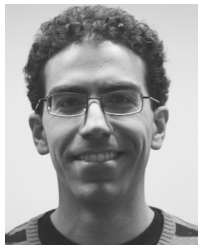
[41] A. S. Mian, M. Bennamoun, and R. Owens, "Three-dimensional model-based object recognition and segmentation in cluttered scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1584–1601, Oct. 2006.

[42] G. Kordelas and P. Daras, "Viewpoint independent object recognition in cluttered scenes exploiting ray-triangle intersection and SIFT algorithms," *Pattern Recognit.*, pp. 3833–3845, 2010.
[43] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 356–369.
[44] W. Cheung and G. Hamarneh, "N-SIFT: N-dimensional scale invariant feature transform," *IEEE Trans. Image Process.*, vol. 18, no. 9, pp. 2012–2021, Sep. 2009.
[45] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3D shape descriptors," in *Proc. Eurograph. Symp. Geometry Process.*, 2003, pp. 156–164.
[46] N. Snavely and S. Seitz, "Photo tourism: Exploring photo collections in 3D," *ACM Trans. Graphics*, p. 835, 2006.
[47] M. Vergauwen and L. Gool, "Web-based 3D reconstruction service," *Mach. Vis. Applicat.*, pp. 411–426, 2006.
[48] R. Bhotika, D. J. Fleet, and K. N. Kutulakos, "A probabilistic theory of occupancy and emptiness," in *Proc. Eur. Conf. Comput. Vis.*, 2002, pp. 112–130.
[49] T. Pollard, "Comprehensive 3-D change detection using volumetric appearance modeling," Ph.D. dissertation, Div. of Appl. Math., Brown Univ., Providence, RI, 2008.
[50] D. E. Crispell, "A Continuous probabilistic scene model for aerial imagery," Ph.D. dissertation, School of Eng., Brown Univ., Providence, RI, 2010.
[51] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognition*, 1999, pp. 246–252.
[52] T. F. Chan, G. H. Golub, and R. J. LeVeque, Updating formulae and a pairwise algorithm for computing sample variances Dept. of Comput. Sci., Stanford Univ., Stanford, CA, Tech. Rep., 1979.
[53] J. J. Koenderink and A. J. van Doorn, "Representation of local geometry in the visual system," *Biol. Cybern.*, pp. 156–164, 1987.
[54] L. M. J. Florack, B. M. Haar Romeny, J. J. Koenderink, and M. A. Viergever, "Cartesian differential invariants in scale-space," *J. Math. Imag. Vis.*, pp. 327–348, 1993.
[55] W. Freeman and E. Adelson, "The design and use of steerable filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 9, pp. 891–906, Sep. 1991.
[56] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.
[57] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. Alvey Vision Conf.*, 1988.
[58] I. Laptev, "On space-time interest points," *Int. J. Comput. Vis.*, pp. 107–123, 2005.
[59] P. Beaudet, "Rotationally invariant image operators," in *Proc. Int. Joint Conf. Pattern Recognit.*, 1978.
[60] M. Varma and A. Zisserman, "A statistical approach to material classification using image patch exemplars," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 11, pp. 2032–2047, Nov. 2009.
[61] T. K. Leung and J. Malik, "Recognizing surfaces using three-dimensional textons," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1999, pp. 1010–1017.
[62] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proc. Int. Conf. Mach. Learn.*, 1997, pp. 137–142.
[63] J. Sivic, B. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering objects and their location in images," in *Proc. Int. Conf. Comput. Vis.*, 2005, pp. 370–377.
[64] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. Workshop Statist. Learn. Comput. Vis., Eur. Conf. Comput. Vis.*, 2004, pp. 1–22, 3021–3024.
[65] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *Int.. J. Comput. Vision*, pp. 213–238, 2007.
[66] R. Maitra, A. D. Peterson, and A. P. Ghosh, "A systematic evaluation of different methods for initializing the K-means clustering algorithm," *IEEE Trans. Knowl. Data Eng.*, 2010.
[67] P. S. Bradley and U. M. Fayyad, "Refining initial points for K-means clustering," in *Proc. Int. Conf. Mach. Learn.*, 1998, pp. 91–99.
[68] C. Elkan, "Using the triangle inequality to accelerate k-means," in *Proc. Int. Conf. Mach. Learn.*, 2003.

**Maria Isabel Restrepo** (M'12) received the B.S. degree (with honors) in electrical engineering from Trinity College, Hartford, CT, in 2006 and the M.S. degree in applied mathematics from Brown University, Providence, RI, in 2010. She is currently pursuing the Ph.D. degree in electrical engineering at Brown University.

Ms. Restrepo is a member of the IEEE Computer Society, the IEEE Women in Engineering Society, and the Phi Beta Kappa society. Her research interests are in the area of computer vision and machine learning with an emphasis on probabilistic learning, 3-D feature detection, 3-D object recognition, and compositional hierarchies.

**Brandon A. Mayer** (S'11) received the B.S. degree in music engineering and technology and the B.S. degree in electrical engineering from University of Miami, Coral Gables, FL, in 2008 and the M.S. degree in engineering from Brown University, Providence, RI, in 2010. He is currently working towards the Ph.D. degree in electrical engineering at Brown University, focusing on computer vision and machine learning research.



**Ali Osman Ulusoy** (S'10) received the B.S. degree in computer engineering from Bilkent University, Ankara, Turkey, in 2008 and the M.Sc. degree in applied mathematics from Brown University, Providence, RI, in 2011. He is currently pursuing the Ph.D. degree in electrical engineering at Brown University, working on computer vision and machine learning.

**Joseph L. Mundy** received the B.S. and Ph.D. degrees in electrical engineering from Rensselaer Polytechnic Institute, Troy, NY, in 1963 and 1969, respectively.

He joined General Electric Global Research in 1963. In his early career at GE, he carried out research in solid-state physics and integrated circuit devices. In the early 1970s, he formed a research group on computer vision with emphasis on industrial inspection. His group developed a number of inspection systems for GE's manufacturing divisions, including a system for the inspection of lamp filaments that exploited syntactic methods in pattern recognition. During the 1980s, his group moved toward more basic research in object recognition and geometric reasoning. In 1988, he was named a Coolidge Fellow, which awarded him a sabbatical at Oxford University, Oxford, U.K. At Oxford, he collaborated on the development of theory and application of geometric invariants. In 2002, he retired from GE Global Research and joined the School of Engineering, Brown University, Providence, RI. At Brown University, his research is in the area of video analysis and probabilistic computing.