Towards Intelligent Mission Profiles of Micro Air Vehicles: Multiscale Viterbi Classification

Sinisa Todorovic and Michael C. Nechyba

ECE Department, University of Florida, Gainesville, FL 32611
{sinisha, nechyba}@mil.ufl.edu,
WWW home page: http://www.mil.ufl.edu/mav

Abstract. In this paper, we present a vision system for object recognition in aerial images, which enables broader mission profiles for Micro Air Vehicles (MAVs). The most important factors that inform our design choices are: real-time constraints, robustness to video noise, and complexity of object appearances. As such, we first propose the HSI color space and the Complex Wavelet Transform (CWT) as a set of sufficiently discriminating features. For each feature, we then build tree-structured belief networks (TSBNs) as our underlying statistical models of object appearances. To perform object recognition, we develop the novel multiscale Viterbi classification (MSVC) algorithm, as an improvement to multiscale Bayesian classification (MSBC). Next, we show how to globally optimize MSVC with respect to the feature set, using an adaptive feature selection algorithm. Finally, we discuss context-based object recognition, where visual contexts help to disambiguate the identity of an object despite the relative poverty of scene detail in flight images, and obviate the need for an exhaustive search of objects over various scales and locations in the image. Experimental results show that the proposed system achieves smaller classification error and fewer false positives than systems using the MSBC paradigm on challenging real-world test images.

1 Introduction

We seek to improve our existing vision system for Micro Air Vehicles (MAVs) [1–3] to enable more intelligent MAV mission profiles, such as remote traffic surveillance and moving-object tracking. Given many uncertain factors, including variable lighting and weather conditions, changing landscape and scenery, and the time-varying on-board camera pose with respect to the ground, object recognition in aerial images is a challenging problem even for the human eye. Therefore, we resort to a probabilistic formulation of the problem, where careful attention must be paid to selecting sufficiently discriminating features and a sufficiently expressive modeling framework. More importantly, real-time constraints and robustness to video noise are critical factors that inform the design choices for our MAV application.

Having experimented with color and texture features [3], we conclude that both color and texture clues are generally required to accurately discriminate object appearances. As such, we employ both the HSI color space, for color representation, and also the Complex Wavelet Transform (CWT), for multi-scale texture representation. In some cases, where objects exhibit easy-to-classify appearances, the proposed feature

set is not justifiable in light of real-time processing constraints. Therefore, herein, we propose an algorithm for selecting an optimal feature subspace from the given HSI and CWT feature space that considers both correctness of classification and computational cost.

Given this feature set, we then choose tree-structured belief networks (TSBNs) [4], as underlying statistical models to describe pixel neighborhoods in an image at varying scales. We build TSBNs for both color and wavelet features, using Pearl's message passing scheme [5] and the EM algorithm [6]. Having trained TSBNs, we then proceed with supervised object recognition. In our approach, we exploit the idea of *visual contexts* [7], where initial identification of the overall type of scene facilitates recognition of specific objects/structures within the scene. Objects (e.g., cars, buildings), the locations where objects are detected (e.g., road, meadow), and the category of locations (e.g., sky, ground) form a taxonomic hierarchy. Thus, object recognition in our approach consists of the following steps. First, sky/ground regions in the image are identified. Second, pixels in the ground region[1] are labeled using the learned TSBNs for predefined locations (e.g., road, forest). Finally, pixels of the detected locations of interest are labeled using the learned TSBNs for a set of predefined objects (e.g., car, house).

To reduce classification error (e.g., "blocky segmentation"), which arises from the fixed-tree structure of TSBNs, we develop the novel multiscale Viterbi classification (MSVC) algorithm, an improved version of multiscale Bayesian classification (MSBC) [8, 9]. In the MSBC approach, image labeling is formulated as Bayesian classification at each scale of the tree model, separately; next, transition probabilities between nodes at different scales are learned using the greedy classification-tree algorithm, averaging values over all nodes and over all scales; finally, it is assumed that labels at a "coarse enough" scale of the tree model are statistically independent. On the other hand, in our MSVC formulation, we perform Bayesian classification only at the finest scale, fusing downward the contributions of all the nodes at all scales in the tree; next, transition probabilities between nodes at different scales are learned as histogram distributions that are not averaged over all scales; finally, we assume dependent class labels at the coarsest layer of the tree model, whose distribution we again estimate as a histogram distribution.

2 Feature Space

Our feature selection is largely guided by extensive experimentation reported in our prior work [3], where we sought a feature space, which spans both color and texture domains, and whose extraction meets our tight real-time constraints.

We obtained the best classification results when color was represented in the HSI color space. Tests suggested that hue (H), intensity (I) and saturation (S) features were more discriminative, when compared to the inherently highly correlated features of the RGB and other color systems [10]. Also, first-order HSI statistics proved to be sufficient and better than the first and higher-order statistics of other color systems.

For texture-feature extraction, we considered several filtering, model-based and statistical methods. Our conclusion agrees with the comparative study of Randen *et*

[1] Recognition of objects in the sky region can be easily incorporated into the algorithm.

al. [11], which suggests that for problems where many textures with subtle spectral differences occur, as in our case, it is reasonable to assume that spectral decomposition by a filter bank yields consistently superior results over other texture analysis methods. Our experimental results also suggest that it is necessary to analyze both local and regional properties of texture. Most importantly, we concluded that a prospective texture analysis tool must have high directional selectivity. As such, we employ the complex wavelet transform (CWT), due to its inherent representation of texture at different scales, orientations and locations [12]. The CWT's directional selectivity is encoded in six bandpass subimages of complex coefficients at each level, coefficients that are strongly oriented at angles $\pm 15^\circ, \pm 45^\circ, \pm 75^\circ$. Moreover, CWT coefficient magnitudes exhibit the following properties [13, 14]: i) *multi-resolutional* representation, ii) *clustering*, and iii) *persistence* (i.e. propagation of large/small values through scales).

Computing CWT coefficients at all scales and forming a pyramid structure from HSI values, where coarser scales are computed as the mean of the corresponding children, we obtain nine feature trees. These feature structures naturally give rise to TSBN statistical models.

3 Tree-Structured Belief Networks

So far, two main types of prior models have been investigated in the statistical image modeling literature – namely, noncausal and causal Markov random fields (MRF). The most commonly used MRF model is the tree-structured belief network (TSBN) [8, 9, 14–16]. A TSBN is a generative model comprising hidden, X, and observable, Y, random variables (RVs) organized in a tree structure. The edges between nodes, representing X, encode Markovian dependencies across scales, whereas Y's are assumed mutually independent given the corresponding X's, as depicted in Figure 1. Herein, we enable input of observable information, Y, also to higher level nodes, preserving the tree dependences among hidden variables. Thus, Y at the lower layers inform the belief network on the statistics of smaller groups of neighboring pixels (at the lowest level, one pixel), whereas Y at higher layers represent the statistics of larger areas in the image. Hence, we enforce the nodes of a tree model to represent image details at

(a) (b)

Fig. 1. Differences in TSBN models: (a) observable variables at the lowest layer only; (b) our approach: observable variables at all layers. Black nodes denote observable variables and white nodes represent hidden random variables connected in a tree structure.

various scales.[2] Furthermore, we assume that features are mutually independent, which is reasonable given that wavelets span the feature space using orthogonal basis functions. Thus, our overall statistical model consists of nine mutually independent trees $\mathcal{T}_f$, $f \in \mathcal{F} = \{\pm 15^\circ, \pm 45^\circ, \pm 75^\circ, H, S, I\}$.

In supervised learning problems, as is our case, a hidden RV, x_i, assigned to a tree node i, $i \in \mathcal{T}_f$, represents a pixel label, k, which takes values in a pre-defined set of image classes, C. The state of node i is conditioned on the state of its parent j and is specified by conditional probability tables, P_{ij}^{kl}, $\forall i, j \in \mathcal{T}_f$, $\forall k, l \in C$. It follows that the joint probability of all hidden RVs, $X = \{x_i\}$, can be expressed as

$$P(X) = \prod_{i,j \in \mathcal{T}_f} \prod_{k,l \in C} P_{ij}^{kl} \ . \tag{1}$$

We assume that the distribution of an observable RV, y_i, depends solely on the node state, x_i. Consequently, the joint pdf of $Y = \{y_i\}$ is expressed as

$$P(Y|X) = \prod_{i \in \mathcal{T}_f} \prod_{k \in C} p(y_i | x_i = k, \theta_i^k) \ , \tag{2}$$

where $p(y_i|x_i = k, \theta_i^k)$ is modeled as a mixture of M Gaussians,[3] whose parameters are grouped in θ_i^k. In order to avoid the risk of overfitting the model, we assume that the $\theta' s$ are equal for all i at the same scale. Therefore, we simplify the notation as $p(y_i|x_i{=}k, \theta_i^k){=}p(y_i|x_i)$. Thus, a TSBN is fully specified by the joint distribution of X and Y given by

$$P(X, Y) = \prod_{i,j \in \mathcal{T}_f} \prod_{k,l \in C} p(y_i|x_i) P_{ij}^{kl} \ . \tag{3}$$

Now, to perform pixel labeling, we face the probabilistic inference problem of computing the conditional probability $P(X|Y)$. In the graphical-models literature, the best-known inference algorithm for TSBNs is Pearl's message passing scheme [5, 18]; similar algorithms have been proposed in the image-processing literature [8, 14, 15]. Essentially, all these algorithms perform belief propagation up and down the tree, where after a number of training cycles, we obtain all the tree parameters necessary to compute $P(X|Y)$. Note that, simultaneously with Pearl's belief propagation, we employ the EM algorithm [6] to learn the parameters of Gaussian-mixture distributions. Since our TSBNs have observable variables at all tree levels, the EM algorithm is naturally performed at all scales. Finally, having trained TSBNs for a set of image classes, we proceed with multiscale image classification.

4 Multiscale Viterbi Classification

Image labeling with TSBNs is characterized by "blocky segmentations," due to their fixed-tree structure. Recently, several approaches have been reported to alleviate this problem (e.g., [19, 20]), albeit at prohibitively increased computational cost. Given the

[2] This approach is more usual in the image processing community [8, 9, 14].

[3] For large M, a Gaussian-mixture density can approximate any probability density [17].

real-time requirements for our MAV application, these approaches are not realizable, and the TSBN framework remains attractive in light of its linear-time inference algorithms. As such, we resort to our novel multiscale Viterbi classification (MSVC) algorithm to reduce classification error instead.

Denoting all hidden RVs at the leaf level L as X^L, classification at the finest scale is performed according to the MAP rule

$$\hat{X}^L = \arg\max_{X^L}\{P(Y|X)P(X)\} = \arg\max_{X^L} g^L. \tag{4}$$

Assuming that the class label, x_i^l, of node i at scale ℓ, completely determines the distribution of y_i^l, it follows that:

$$P(Y|X) = \prod_{\ell=L}^{0}\prod_{i\in\ell} p(y_i^\ell|x_i^\ell)\ , \tag{5}$$

where $p(y_i^\ell|x_i^\ell)$ is a mixture of Gaussians, learned using the inference algorithms discussed in Section 3. As is customary for TSBNs, the distribution of X^ℓ is completely determined by $X^{\ell-1}$ at the coarser $\ell-1$ scale. However, while, for training, we build TSBNs where each node has only one parent, here, for classification, we introduce a new multiscale structure where we allow nodes to have more than one parent. Thus, in our approach to image classification, we account for horizontal statistical dependencies among nodes at the same level, as depicted in Figure 2. The new multiresolution model accounts for all the nodes in the trained TSBN, except that it no longer forms a tree structure; hence, it becomes necessary to learn new conditional probability tables corresponding to the new edges. In general, the Markov chain rule reads:

$$P(X) = \prod_{\ell=L}^{0}\ \prod_{i\in\ell} P(x_i^\ell|X^{\ell-1})\ \ . \tag{6}$$

The conditional probability $P(x_i^\ell|X^{\ell-1})$ in (6), unknown in general, must be estimated using a prohibitive amount of data. To overcome this problem, we consider, for each node i, a 3×3 box encompassing parent nodes that neighbor the initial parent j of i in the quad-tree. The statistical dependence of i on other nodes at the next coarser scale,

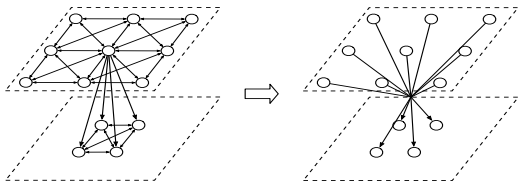


Fig. 2. Horizontal dependences among nodes at the same level are modeled by vertical dependences of each node on more than one parent.

in most cases, can be neglected. Thus, we assume that a nine-dimensional vector $v_j^{\ell-1}$, containing nine parents, represents a reliable source of information on the distribution of all class labels $X^{\ell-1}$ for child node i at level ℓ. Given this assumption, we rewrite expression (6) as

$$P(X) = \prod_{\ell=L}^{0} \prod_{i \in \ell} \prod_{j \in \ell-1} P(x_i^\ell | v_j^\ell) \ . \tag{7}$$

Now, we can express the discriminant function in (4) in a more convenient form as

$$g^L = \prod_{\ell=L}^{0} \prod_{i \in \ell} \prod_{j \in \ell-1} p(y_i^\ell | x_i^\ell) P(x_i^\ell | v_j^{\ell-1}) \ . \tag{8}$$

Assuming that our features $f \in \mathcal{F}$ are mutually independent, the overall maximum discriminant function can therefore be computed as

$$g^L = \prod_{f \in \mathcal{F}} g_f^L \ . \tag{9}$$

The unknown transition probabilities $P(x_i^\ell | v_j^{\ell-1})$ can be learned through vector quantization [21], together with Pearl's message passing scheme. After the prior probabilities of class labels of nodes at all tree levels are learned using Pearl's belief propagation, we proceed with instantiation of random vectors v_i^ℓ. For each tree level, we obtain a data set of nine-dimensional vectors, which we augment with the class label of the corresponding child node. Finally, we perform vector quantization over the augmented ten-dimensional vectors. The learned histogram distributions represent estimates of the conditional probability tables. Clearly, to estimate the distribution of a ten-dimensional random vector it is necessary to provide a sufficient number of training images, which is readily available from recorded MAV-flight video sequences. Moreover, since we are not constrained by the same real-time constraints during training as during flight, the proposed learning procedure results in very accurate estimates, as is demonstrated in Section 7.

The estimated transition probabilities $P(x_i^\ell | v_j^{\ell-1})$ enable classification from scale to scale in Viterbi fashion. Starting from the highest level downwards, at each scale, we maximize the discriminant function g^L along paths that connect parent and children nodes. From expressions (8) and (9), it follows that image labeling is carried out as

$$\hat{x}_i^L = \arg\max_{x_i^L \in C} \prod_{f \in \mathcal{F}} \prod_{\ell=L}^{0} \prod_{i \in \ell} \prod_{j \in \ell-1} p(y_{i,f}^\ell | x_i^\ell) P(x_i^\ell | \hat{v}_j^{\ell-1}) \ , \tag{10}$$

where $\hat{v}_j^{\ell-1}$ is determined from the previously optimized class labels of the coarser scale $\ell - 1$.

5 Adaptive Feature Selection

We have already pointed out that in some cases, where image classes exhibit favorable properties, there is no need to compute expression (10) over all features. Below, we

present our algorithm for adaptive selection of the optimal feature set, $\mathcal{F}_{sel}$, from the initial feature set, $\mathcal{F}$.

1. Form a new empty set $\mathcal{F}_{sel} = \{\emptyset\}$; assign $g_{new} = 1$, $g_{old} = 0$;
2. Compute $\hat{g}_f^L$, $\forall f \in \mathcal{F}$, given by (8) for $\hat{x}_i^L$ given by (10);
3. Move the best feature f^*, for which $\hat{g}_{f^*}^L$ is maximum, from $\mathcal{F}$ to $\mathcal{F}_{sel}$;
4. Assign $g_{new} = \prod_{f \in \mathcal{F}_{sel}} \hat{g}_f^L$;
5. If $(g_{new} < g_{old})$ delete f^* from $\mathcal{F}_{sel}$ and go to step 3;
6. Assign $g_{old} = g_{new}$;
7. If $(\mathcal{F} \neq \{\emptyset\})$ go to step 3;
8. Exit and segment the image using features in $\mathcal{F}_{sel}$.

The discriminant function, g, is nonnegative; hence, the above algorithm finds at least one optimal feature. Clearly, the optimization criteria above consider both correctness of classification and computational cost.

6 Object Recognition Using Visual Contexts

In our approach to object recognition, we seek to exploit the idea of *visual contexts* [7]. Having previously identified the overall type of scene, we can then proceed to recognize specific objects/structures within the scene. Thus, objects, the locations where objects are detected, and the category of locations form a taxonomic hierarchy. There are several advantages to this type of approach. Contextual information helps disambiguate the identity of objects despite the poverty of scene detail in flight images and quality degradation due to video noise. Furthermore, exploiting visual contexts, we obviate the need for an exhaustive search of objects over various scales and locations in the image.

For each aerial image, we first perform categorization, i.e., sky/ground image segmentation. Then, we proceed with localization, i.e., recognition of global objects and structures (e.g., road, forest, meadow) in the ground region. Finally, in the recognized locations we search for objects of interest (e.g., cars, buildings). To account for different flight scenarios, different sets of image classes can be defined accordingly. Using the prior knowledge of a MAV's whereabouts, we can reduce the number of image classes, and, hence, computational complexity as well as classification error.

At each layer of the contextual taxonomy, downward, we conduct MSVC-based object recognition. Here, we generalize the meaning of image classes to any global-object appearance. Thus, the results from Sections 3 and 4 are readily applicable. In the following example, shown in Figure 3, each element of the set of locations {*road, forest, lawn*} induces subsets of objects, say, {*car, cyclist*} pertaining to *road*. Consequently, when performing MSVC, we consider only a small finite number of image classes, which improves recognition results. Thus, in spite of video noise and poverty of image detail, the object in Figure 3, being tested against only two possibilities, is correctly recognized as a *car*.

7 Results

In this section, we demonstrate the performance of the proposed vision system for real-time object recognition in supervised-learning settings. We carried out several sets of

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

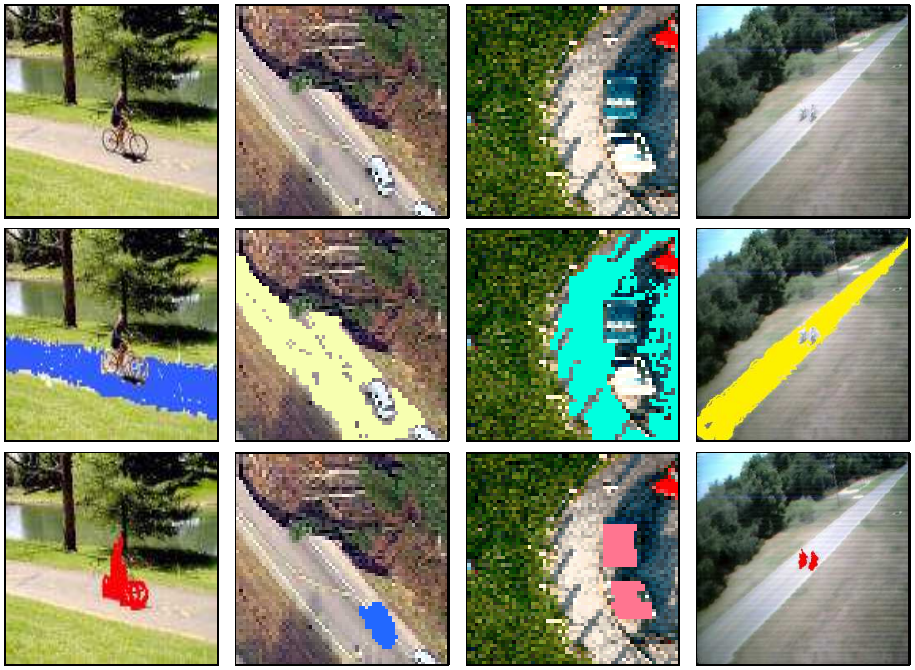
Fig. 3. The hierarchy of visual contexts conditions gradual image interpretation: (a) a 128×128 flight image; (b) categorization: sky/ground classification; (c) localization: road recognition; (d) car recognition.

experiments which we report below. For space reasons, we discuss only our results for car and cyclist recognition in flight video.

For training TSBNs, we selected 200 flight images for the car and cyclist classes. We carefully chose the training sets to account for the enormous variability in car and cyclist appearances, as illustrated in Figure 4 (top row). After experimenting with different image resolutions, we found that reliable labeling was achievable for resolutions as coarse as 64×64 pixels. At this resolution, all the steps in object recognition (i.e., sky/ground classification, road localization and car/cyclist recognition), when the feature set comprises all nine features, takes approximately 0.1s on an Athlon 2.4GHz PC. For the same set-up, but for only one optimal feature, recognition time is less than 0.07s,[4] which is quite sufficient for the purposes of moving-car or moving-bicycle tracking. Moreover, for a sequence of video images, the categorization and localization steps could be performed only for images that occur at specified time intervals, although, in our implementation, we process every image in a video sequence for increased noise robustness.

After training our car and bicycle statistical models, we tested MSVC performance on a set of 100 flight images. To support our claim that MSVC outperforms MSBC, we carried out a comparative study of the two approaches on the same dataset. For validation accuracy, we separated the test images into two categories. The first category consists of 50 test images with easy-to-classify car/cyclist appearances as illustrated in Figure 4a and Figure 4b. The second category includes another 50 images, where multiple hindering factors (e.g. video noise and/or landscape and lighting variability, as depicted in Figure 4c and Figure 4d) conditioned poor classification. Ground truth was established through hand-labeling pixels belonging to objects for each test image. Then, we ran the MSVC and MSBC algorithms, accounting for the image-dependent optimal subset of features. Comparing the classification results with ground truth, we computed the percentage of erroneously classified pixels for the MSVC and MSBC algorithms.

[4] Note that even if only one set of wavelet coefficients is optimal, it is necessary to compute all other sets of wavelets in order to compute the optimal one at all scales. Thus, in this case, time savings are achieved only due to the reduced number of features for which MSVC is performed.



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

Fig. 4. Recognition of road objects: (top) Aerial flight images; (middle) localization: road recognition; (bottom) object recognition. MSVC was performed for the following optimized sets of features: (a) $\mathcal{F}_{sel} = \{H, I, -45^\circ\}$, (b) $\mathcal{F}_{sel} = \{H, \pm 75^\circ\}$, (c) $\mathcal{F}_{sel} = \{\pm 15^\circ, \pm 45^\circ\}$, (d) $\mathcal{F}_{sel} = \{H, \pm 45^\circ\}$.

The results are summarized in Table 1, where we do not report the error of complete misses (CM) (i.e., the error when an object was not detected at all) and the error of swapped identities (SI) (i.e., the error when an object was detected but misinterpreted). Also, in Table 2, we report the recognition results for 86 and 78 car/cyclist objects in the first and second categories of images, respectively. In Figure 5, we illustrate better MSVC performance over MSBC for a sample first-category image.

Table 1. Percentage of misclassified pixels by MSVC and MSBC

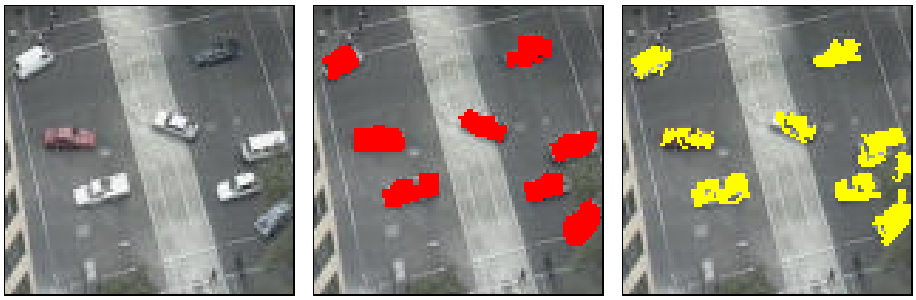
	I category images	II category images
MSVC	4%	10%
MSBC	9%	17%

Finally, we illustrate the validity of our adaptive feature selection algorithm. In Figure 6, we present MSVC results for different sets of features. Our adaptive feature selec-

tion algorithm, for the given image, found $\mathcal{F}_{sel} = \{H, -45°, \pm 75°\}$ to be the optimal feature subset. To validate the performance of the selection algorithm, we segmented the same image using all possible subsets of the feature set $\mathcal{F}$. For space reasons, we illustrate only some of these classification results. Obviously, from Figure 6, the selected optimal features yield the best image labeling. Moreover, note that when all the features were used in classification, we actually obtained worse results. In Table 3, we present the percentage of erroneously classified pixels by MSVC using different subsets of features for our two categories of 100 test images. As before, we do not report the error of complete misses. Clearly, the best classification results were obtained for the optimal set of features.

Table 2. Correct recognition (CR), complete miss (CM), and swapped identity (SI)

	I category images (86 objects)			II category images (78 objects)		
	CR	CM	SI	CR	CM	SI
MSVC	81	1	4	69	5	4
MSBC	78	2	6	64	9	5

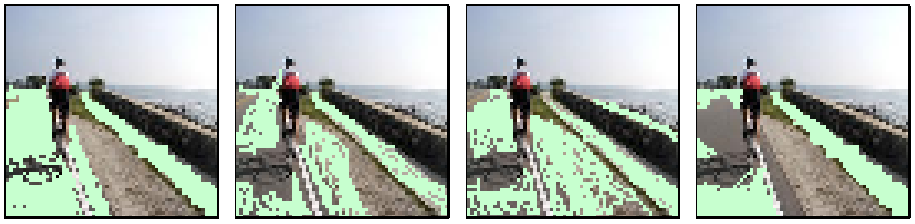


(a) (b) (c)

Fig. 5. Better performance of MSVC vs. MSBC for the optimal feature set $\mathcal{F}_{sel} = \{H, I, \pm 15°, \pm 75°\}$: (a) a first-category image; (b) MSVC; (c) MSBC.

8 Conclusion

Modeling complex classes in natural-scene images requires an elaborate consideration of class properties. The most important factors that informed our design choices for a MAV vision system are: (1) real-time constraints, (2) robustness to video noise, and (3)



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

Fig. 6. Validation of the feature selection algorithm for road recognition: (a) MSVC for the optimized $\mathcal{F}_{sel} = \{H, -45°, \pm 75°\}$; (b) MSVC for all nine features in $\mathcal{F}$; (c) MSVC for subset $\mathcal{F}_1 = \{H, S, I\}$; (d) MSVC for subset $\mathcal{F}_2 = \{\pm 15°, \pm 45° \pm 75°\}$.

Table 3. Percentage of misclassified pixels by MSVC

	I category	II category
$\mathcal{F}_{sel}$	4%	10%
$\mathcal{F} = \{H, S, I, \pm 15°, \pm 45° \pm 75°\}$	13%	17%
$\mathcal{F}_1 = \{H, S, I\}$	16%	19%
$\mathcal{F}_2 = \{\pm 15°, \pm 45° \pm 75°\}$	14%	17%

complexity of various object appearances in flight images. In this paper, we first presented our choice of features: the HSI color space, and the CWT. Then, we introduced the TSBN model and the training steps for learning its parameters. Further, we described how the learned parameters could be used for computing the likelihoods of all nodes at all TSBN scales. Next, we proposed and demonstrated multiscale Viterbi classification (MSVC), as an improvement to multiscale Bayesian classification. We showed how to globally optimize MSVC with respect to the feature set through an adaptive feature selection algorithm. By determining an optimal feature subset, we successfully reduced the dimensionality of the feature space, and, thus, not only approached the real-time requirements for applications operating on real-time video streams, but also improved overall classification performance. Finally, we discussed object recognition based on visual contexts, where contextual information helps disambiguate the identity of objects despite a poverty of scene detail and obviates the need for an exhaustive search of objects over various scales and locations in the image. We organized test images into two categories of difficulty and obtained excellent classification results, especially for complex-scene/noisy images, thus validating the proposed approach.

References

1. Ettinger, S.M., Nechyba, M.C., Ifju, P.G., Waszak, M.: Vision-guided flight stability and control for Micro Air Vehicles. In: Proc. IEEE Int'l Conf. Intelligent Robots and Systems (IROS), Laussane, Switzerland (2002)
2. Ettinger, S.M., Nechyba, M.C., Ifju, P.G., Waszak, M.: Vision-guided flight stability and control for Micro Air Vehicles. Advanced Robotics **17** (2003)

3. Todorovic, S., Nechyba, M.C., Ifju, P.: Sky/ground modeling for autonomous MAVs. In: Proc. IEEE Int'l Conf. Robotics and Automation (ICRA), Taipei, Taiwan (2003)
4. Cowell, R.G., Dawid, A.P., Lauritzen, S.L., Spiegelhalter, D.J.: Probabilistic Networks and Expert Systems. Springer-Verlag, New York (1999)
5. Pearl, J.: Probabilistic Reasoning in Intelligent Systems : Networks of Plausible Inference. Morgan Kaufamnn, San Mateo (1988)
6. McLachlan, G.J., Thriyambakam, K.T.: The EM algorithm and extensions. John Wiley & Sons (1996)
7. Torralba, A., Murphy, K.P., Freeman, W.T., Rubin, M.A.: Context-based vision system for place and object recognition. In: Proc. Int'l Conf. Computer Vision (ICCV), Nice, France (2003)
8. Cheng, H., Bouman, C.A.: Multiscale bayesian segmentation using a trainable context model. IEEE Trans. Image Processing **10** (2001)
9. Choi, H., Baraniuk, R.G.: Multiscale image segmentation using wavelet-domain Hidden Markov Models. IEEE Trans. Image Processing **10** (2001)
10. Cheng, H.D., Jiang, X.H., Sun, Y., Jingli, W.: Color image segmentation: advances and prospects. Pattern Recognition **34** (2001)
11. Randen, T., Husoy, H.: Filtering for texture classification: A comparative study. IEEE Trans. Pattern Analysis Machine Intelligence **21** (1999)
12. Kingsbury, N.: Image processing with complex wavelets. Phil. Trans. Royal Soc. London **357** (1999)
13. Mallat, S.: A Wavelet Tour of Signal Processing. 2nd edn. Academic Press (2001)
14. Crouse, M.S., Nowak, R.D., Baraniuk, R.G.: Wavelet-based statistical signal processing using Hidden Markov Models. IEEE Trans. Signal Processing **46** (1998)
15. Bouman, C.A., Shapiro, M.: A multiscale random field model for Bayesian image segmentation. IEEE Trans. Image Processing **3** (1994)
16. Feng, X., Williams, C.K.I., Felderhof, S.N.: Combining belief networks and neural networks for scene segmentation. IEEE Trans. Pattern Analysis Machine Intelligence **24** (2002)
17. Aitkin, M., Rubin, D.B.: Estimation and hypothesis testing in finite mixture models. J. Royal Stat. Soc. **B-47** (1985)
18. Frey, B.J.: Graphical Models for Machine Learning and Digital Communication. The MIT Press, Cambridge, MA (1998)
19. Storkey, A.J., Williams, C.K.I.: Image modeling with position-encoding dynamic trees. IEEE Trans. Pattern Analysis Machine Intelligence **25** (2003)
20. Irving, W.W., Fieguth, P.W., Willsky, A.S.: An overlapping tree approach to multiscale stochastic modeling and estimation. IEEE Trans. Image Processing **6** (1997)
21. Linde, Y., Buzo, A., Gray, R.M.: An algorithm for vector quantizer design. IEEE Trans. on Communications **COM-28** (1980)