Categorization in Natural Time-Varying Image Sequences

Teresa Ko, Stefano Soatto, Deborah Estrin
UCLA Vision Lab, Los Angeles, CA
{tko, soatto, destrin}@cs.ucla.edu

Abstract

Approaches to single image categorization do not easily generalize to natural time-varying image sequences. In natural environments, object categories tend to have few features that help to distinguish between each other and the surrounding environment. To better discriminate between categories and the surrounding environment, we propose a multi-view categorization approach that exploits the statistics of image sequences rather than single images. The approach is unbiased towards redundant views – that is, it does not matter how many times an object appears from the same viewpoint. At the same time, the approach does not penalize for missing views, so that we do not have to capture an object at all viewpoints to successfully categorize the object.

We first present a data set for studying natural environment monitoring: an image sequence of birds at a feeder station. After manual localization, a baseline bag of features approach was found to perform significantly worse on the proposed data set compared to the standard Caltech 101 data set. We find that our approach increases the categorization accuracy from 48% *to* 58% *on average when compared to an equivalent single view categorization method. Finally, we show how the same metric proposed for the supervised categorization can be used to transform, in an unsupervised manner, an image sequence into a manageable set of categories.*

1. Introduction

Classic categorization techniques used on single photographs do not map easily to categorization in surveillance and monitoring applications. Most significantly, in surveillance and monitoring applications, objects are often captured in multiple frames, rather than captured only once. Whereas previous approach would have no way to take advantage of this additional information, we present a multi-view approach to categorization that does.

Some characteristics inherent to surveillance and monitoring applications make it difficult to use traditional categorization approaches that rely on distinctive features and perform classification over the entire image. One such characteristic is that objects are captured at a fairly low resolution and low frame rate. There is inherent pressure to increase spatial coverage at the cost of object resolution, thereby creating a more challenging detection and recognition task. Similarly, increasing temporal coverage (lifetime) pushes for lower sampling rates, limiting the use of motion features. The resulting image sequence will inevitably contain 1) small objects with few features to distinguish them from one another or from the background, and 2) instances of the same object located in a completely different area in consecutive frames.

Another challenge is that the instances captured of the object are almost never uniformly distributed across all possible viewpoints. An object cannot then be represented by the simple concatenation of all its captured instances. Our main contribution is a multi-view categorization approach that is unbiased to redundant as well as rare views. We model a category as a set of color histograms, where each histogram represents a particular instance of the category. To categorize a test object, the object is modeled as a set of color histograms and the category is assigned to the most similar match between *sets* of histograms. Rather than defining similarity based on average pairwise similarity, we use a best match approach that counts only the best match towards the final similarity score.

Ignoring localization, we demonstrate that a multi-view approach does indeed reduce the ambiguity between categorizes, resulting in more accurate labeling of objects into a set of learned categories. We further explore how the same approach can be adapted to propose unknown categories and object exemplars for these categories from an unlabeled image sequence. One of our contributions is the introduction of a new data set exhibiting the characteristics described above. The data set consists of an image sequence of birds at a feeder station that have been manually annotated and made publicly available. Each bird instance has been localized, and labeled with an object and category id.

Capturing the species distribution of birds is of particular importance, because changes in this distribution is an

978-1-4244-3993-5/09/$25.00 ©2009 IEEE

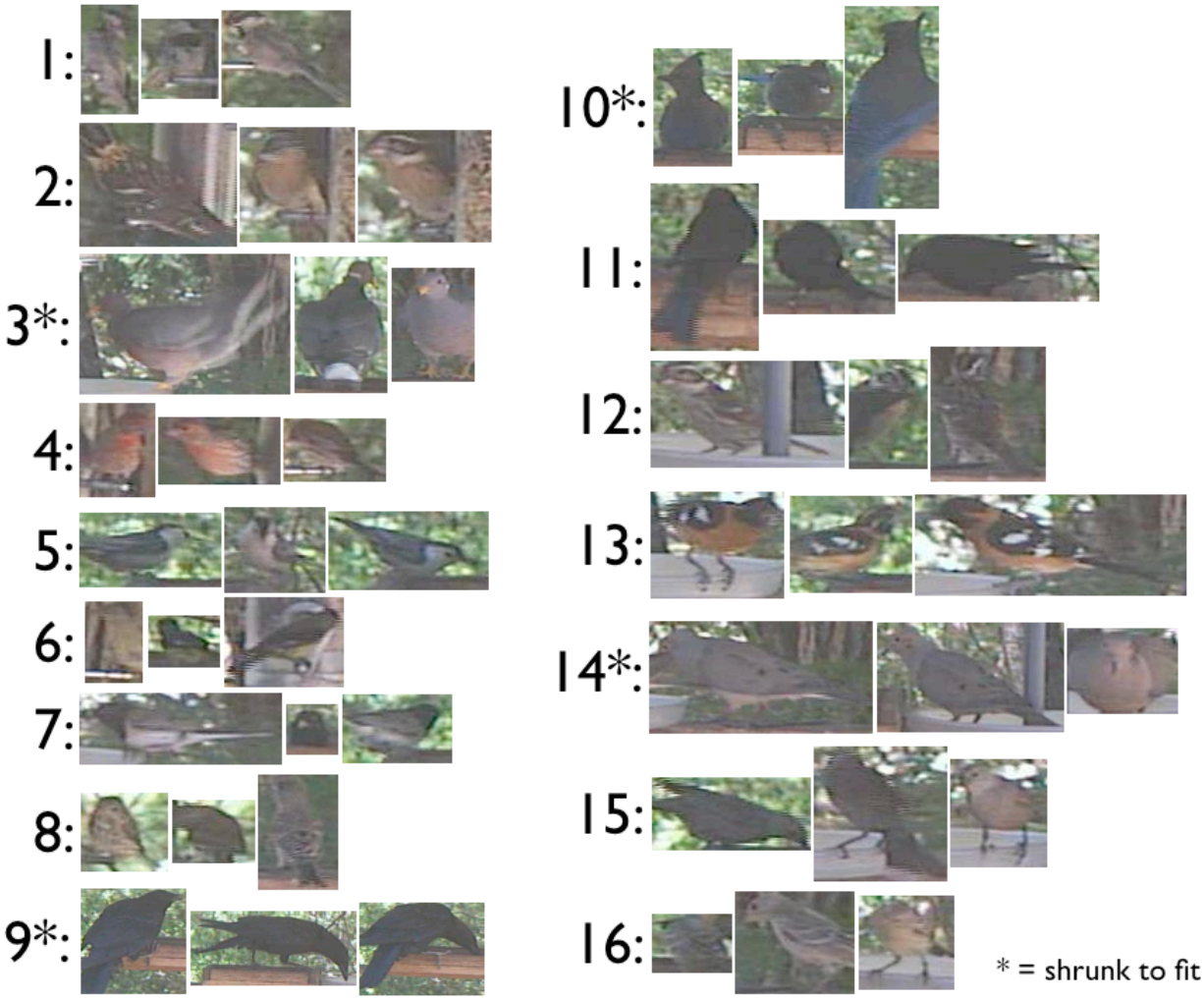

Figure 1: Sample images of each category. Different views of the same category can exhibit very different pixel values, gradients, size and shape.

early indicator of ecosystem changes. This is just one example of a wide range of applications that could benefit from a multi-view categorization method. For example, biologists are may be interested in which animals visit at waterholes, what fish visit particular streams, and the frequency and timing of pollinators. In the urban environment, automated annotations of the interactions of people, pets, bikes, and cars can help city planners and citizens alike to improve their community.

2. Related Work

Object categorization has been a popular area of research in the recent years, but have focused on single image categorization. The most popular approaches [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**] are those that extend a "bag-of-features" approach which use features such as SIFT [**?**] and SURF [**?**], by incorporating statistics on spatial relationships, shapes, or textures. The key to these approaches is finding distinctive features that exhibit less intra-class variance and greater inter-class variance. These approaches rely on the ability to have features on the object occur frequently across the category, and background features to occur infrequently. This is not true for surveillance and monitoring applications. In fact, categories in standard data sets such as Caltech101 [**?**] exhibit so little variability that often the averaged image of a category is still visibly recognizable as an instance of that category [**?**].

While Caltech256 [**?**] addresses some of these issues, objects are still centered and thus allow the algorithm to bypass localization. With a "bag-of-features" approach, a clear extension enables localization. A model of the object should in general respond to a subimage containing that object just as easily as the original image. Examples of such work include Viola-Jones' face and pedestrian detector [**?**, **?**] where face and pedestrian models are tested against subimages spanning a range of positions and scales

to localize the face or pedestrian in the image. More generalized category localization include [**?**, **?**, **?**, **?**, **?**]. Marszalek *et al.*[**?**] and Leibe *et al.* [**?**] use shape based models derived from local features. Shotton *et al.* [**?**] and He *et al.* [**?**] uses a conditional random field to specify spatial relationships between features. In contrast, Fulkerson *et al.* [**?**] does not use a shape model and rather classifies at the pixel level. Because these localization approaches rely on distinctive category features, they would not generalize well to natural environments because the objects of interest mimic their surrounding environments.

Rather, we take advantage of the multiple frames captured of the scene by performing background subtraction to localize objects of interest and capturing multiple instances of the same object. Like [**?**], we integrate the information from multiple views to improve our performance. While their approach focused on a more descriptive feature, we focus on developing a more descriptive model and classifier.

3. Feeder Station Data Set

Part of the contribution of this paper is the introduction and distribution of a new annotated data set for environmental monitoring applications. Images are captured at a rate of one per second from a camera pointed at a feeder station in a natural reserve environment. The annotated portion consists of 3600 color images of 480×704 pixels. Each image is annotated with a bounding box enclosing each bird instance, including flags that indicate whether or not it was interlaced or occluded to the point where categorization could not be performed even by human experts. Each bounding box is also labeled with an object id and a category id. In this case, object refers to a unique bird and category refers to a bird species. We detected 7932 instances, organized into 358 objects and 17 different categories, where one category contains unidentifiable objects. From that, there are 5863 "good" instances that are not occluded or interlaced and 199 objects that have 3 or more "good" instances. Table **??** details how the objects and instances per object are distributed for each category.

This data set captures the natural distribution of object appearance, disappearance, and interactions rather than an artificially balanced set. While most objects have very few instances (< 50), there are a few objects that have a large number of instances (~ 800). The natural occurrence of nuisances is also present in this data set. Birds are cut off by the image frame, occlude one another, and get occluded by background objects. The background exhibits a high degree of variance due to lighting changes, even at the time-scale of a single hour. The background objects are not completely static either: feeders and leaves swing in the wind. The motion of the feeders are also affected by birds landing on the feeder posts.

c	# of objs	# of objs +3 frames	min frames	ave frames	max frames
1	73	47	1	7.5	40
2	18	6	1	12.2	53
3	18	10	1	77.3	725
4	47	28	1	31.1	143
5	18	15	1	18.7	60
6	4	3	2	28.23	80
7	7	6	4	6.7	13
8	33	31	2	30.6	118
9	18	8	1	3.7	23
10	3	2	1	25.0	70
11	5	4	2	50.0	104
12	15	12	1	22.3	137
13	12	9	1	21.0	69
14	3	3	10	114.0	171
15	6	5	2	59.5	156
16	10	10	4	61.5	172

Table 1: Data set statistics. For each category, we list the number of objects in that category, the number of object with 3 or more "good" instances. We also give the minimum, average, and maximum number of instances for objects in the category.

Figure **??** show a few sample object instances from the data set. As one can see from the selected images, there is a large in-class variation in a category's appearance, size and shape. In fact, object instances from different categories can look more similar to each other than object instances from different viewpoints such as category 2 and 12.

4. Multi-view categorization

For supervised multi-view categorization, we define

- An *instance* i is an image patch uniquely defined by the triplet, (t, c, s) where t refers to the frame in the image sequence, $c \in \mathbb{R}^2$, the location in the frame, and $s \in \mathbb{R}^2$, the size of the bounding box.
- An *object* o is a set of instances, $\{i_{t_1}, i_{t+1}, ...i_{t_2}\}$, where t_1 indicates the first frame the object appears in, and t_2 indicates the last frame.
- A *category* c is an set of objects, $\{o_1, o_2, ...o_n\}$, where o_n is the nth object of the category.

Given a set of categories C and a set of objects O, we would like to find the most likely category c for each object $o \in O$. Histograms of various image statistics are a popular representation of object categories, where the slack in the distribution absorbs the intra-category variability. Unfortunately, the distribution of gradient directions in an image is essentially useless in natural habitats, where the objects of interest have evolved to mimic the surrounding environment. Instantaneous image statistics are similarly limiting, for it is difficult even for human experts to spot interesting

objects in a single image. For this reason, we aggregate histograms over time, and make use of color in our representation, as we detail next.

4.1. Image appearance model

For each category $c \in C = \{1, 2, ..., n_c\}$ where n_c is the number of categories we consider, we are given a set of instances, $\mathbf{i} = \{i_1, i_2, ...i_n\}$ that contain an instance of that category. As in classic bag-of-feature approaches, we extract a set of features from the instance. A feature is a statistic, *i.e.* a deterministic function of the data

$$f : \{I(x),\ x \in \Omega\} \mapsto \mathbb{R}^{m \times n} \tag{1}$$

where m is the length of a feature, and n is the number of features extracted. This typically varies from image to image. These features are then binned into a fixed dimensional histogram:

$$\mathbf{h}_I = \{h_{\mathbf{u}} | \mathbf{u} \in U\} \tag{2}$$

$$h_{\mathbf{u}} = n \sum_{s \in f(I)} \delta(b(s) - u) \tag{3}$$

where U is the set of bins and b is a function that maps the feature f into the a bin u in U.

For most of our experiments we use a simple representation, the set of hue and saturation at each pixel in the image, and a simple partition of the color space into uniformly spaced bins. While most bag-of-features approach use more complex features, such as SIFT features and dictionary based histograms, we found for our data set that uniformly binned color histograms already outperformed the more standard approaches. We focus on this representation so that we can compare the relative performance of multi-view versus single view categorization. Yet, this formulation is not limited to this feature type, and could easily be generalized to approaches that use SIFT or more sophisticated binning approaches for data sets better suited for SIFT.

4.2. Comparing appearance models

Each category is represented by a set of histograms, $\mathbf{H_c} = \{\mathbf{h}_{I_1}, \mathbf{h}_{I_2}, ..., \mathbf{h}_{I_n}\}$, where each histogram represents a instance of a category. Typically, a single frame would be compared against each view to determine its histogram. In our approach, we represent a test object, $o \in O = \{1, 2, ...\}$ with a set of histogram from a set of images $\mathbf{H_o} = \{\mathbf{h}_{I_1}, \mathbf{h}_{I_2}, ..., \mathbf{h}_{I_n}\}$. We determine which category c object o belongs to by comparing these sets of histograms.

We use a nearest neighbor approach, but because we are dealing with histograms, we use the Bhattacharyya coefficient, $d(\mathbf{a}, \mathbf{b}) = \sum_u \sqrt{a_u b_u}$, to measure the discrepancy between histograms $\mathbf{a}$ and $\mathbf{b}$. In order to compare *sets* of histograms, we introduce the following discrepancy measure:

$$D(\mathbf{H_c}, \mathbf{H_o}) = \max(\tfrac{1}{|H_o|} \textstyle\sum_{a \in H_o} \max_{b \in H_c} d(a, b), \\ \tfrac{1}{|H_c|} \textstyle\sum_{b \in H_c} \max_{a \in H_o} d(a, b)). \tag{4}$$

$D(\mathbf{H_c}, \mathbf{H_o})$ is 0 when the histograms in H_c have completely different non-zero elements from H_o. $D(\mathbf{H_c}, \mathbf{H_o})$ is 1 when either $H_c \subset H_o$ or $H_o \subset H_c$.

This discrepancy has several useful properties for multi-view categorization in the context of surveillance and monitoring applications. First, it is not biased towards dominant views in either the training or the test sets. This is particularly important because we can not guarantee distributed views as a multi-camera approach would be able to. Our gathered instances are dependent entirely on the behavior of the object of interest.

Therefore, only the best match for any particular instance is counted towards the final score, so that it only matters that there is a good match, not how many good matches there are. Our approach differs from other multi-view camera systems in that multiple views of an object are captured by the object moving rather than the camera moving around an object or multiple cameras focused on the object. Because of this, there are no guarantees or assumptions about the ranges of views captured. Second, it does not penalize for missing views. Because we take the final score to be either the best matches of H_o to H_c or the best matches of H_c to H_o, instances that show rare views can still result in a high final score. In this way, we allow for the test object to both not span the appearance space of the category as well as not evenly represent it.

4.3. Evaluation

Using the data set described in Section **??**, we first ignore the localization problem and evaluate the performance of object level categorization. We partition each category by randomly selecting a number of objects for our training set and using the rest for testing. We consider only objects that have at least 3 "good" instances. For single view categorization, we categorize an instance by selecting the label of the best match from all the training frames according to the Bhattacharyya similarity measure. For multi-view, we create an object model of object o from all of its associated instances. We compare this model to the category models according to Equation (**??**), and assign each frame the label of the best matched category. This experiment was repeated 30 times with randomly selected training and test sets.

As we increase the number of objects used in training (regardless of the number of frames available for that object), we saw an improvement in the accuracy of the remaining objects used for testing (Figure **??**). While a cursory look at the sample images in Figure **??** indicate that SIFT (a

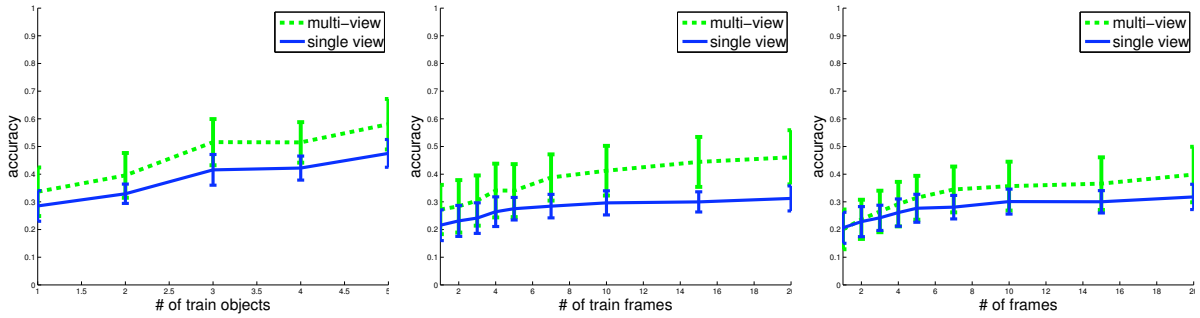

Figure 2: How performance is affected by the # of frames per object used for comparison against the learned category model. Left) We find that integrating multi-views consistently improves the performance when compared again a single view classification, and that as we increase the number of objects in our training set, the performance improvements are even greater. Center: More importantly, we see significant improvements as the number of training frames are increased, indicating that the additional frames aid in distinguishing categories. Right: We find that there is a smaller improvement when test frames are limited, suggesting that capturing the correct frame is also important in the classifiers ability to distinguish categories.

feature performed on gradients) would find very little useful statistics, we did comparison against the classic bag of feature approach described in [**?**]. The accuracy was 11.23% on average, significantly worse than the bag of features approach using color features used here.

Since we are proposing that multi-view classification can outperform single view classification, we look also at how the multiple instances are contributing to the classification. Rather than learning a model from all the frames of the object, we take a random subset of instances and report how accuracy is effected (Figure **??**). We evaluated the effect on accuracy as we increase the number of training instances, and as we increase both the number of training and testing frames. Each additional view widen the gap between the performance of the multi-view versus the single view classification. This suggests that the multi-view approach helps disambiguate between ambiguous views from different categories.

4.3.1 Vector Quantization

Often, redundant views can be captured when an object stays in the same position over a long period of time. Because our discrepancy measure takes the best pairwise comparison, redundant views are discounted. This suggests that vector quantization can be used to reduce the number of pairwise comparisons that need to be performed.

Standard compression techniques assume a euclidean distance. We test vector quantization via lossy compression [**?**], even though our features do not lie on the Euclidean space. One advantage of lossy compression as compared to other vector quantization approaches, such as k-means, is that it does not require a specified number of clusters. We also try a vector quantization technique using the Bhattacharyya distance. Clusters are formed where pairwise comparison results in a discrepancy smaller than some threshold. As we increase the compression of our models, we suffer little performance loss, as is shown in Figure **??**.

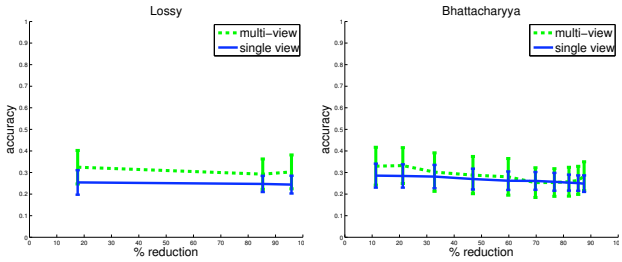

Figure 3: For either compresion method, as we reduce the model size dramatically, performance degrades only a few percentage points.

5. Unsupervised multi-view categorization

From an image sequence, we would like to simultaneously detect objects moving through the sequence and group the objects into sensible categories. Given a set of images, $\mathbf{I} = \{I_1, I_2, ..., I_T\}$, we want to label each pixel location in time and space with (b, o, c) where $b \in B = \{0, 1\}$ where 0 indicates to background and 1 indicates foreground. A category, $c \in C = \{1, 2, 3, ...\}$ is characterized by its appearance model. The set of objects, $\{o_1, o_2, ...\}$ where $o \in O = \{1, 2, 3, ...\}$ considered to be in category c are objects whose appearance model are more similar to that category c than any other category in C. We are trying to answer three questions which are not necessarily independent.

- For any given pixel location in time and space (x, t), is this foreground or background?
- What are the set of pixel locations that belong to a sin-

gle object?

- What objects belong in the same category?

We breakdown each of these tasks in the following section.

5.1. Foreground/Background Labels

We implemented a variant on the background subtraction algorithm described in [**?**] so that it may work in batch mode. This does away with the need to find clean background images or background update parameters.

For each location $\mathbf{x} \in \Re^2$ in the image I, we represent the appearance by a color probability density function (Hue-Saturation) of a set of pixels in space and time. The set of pixels used to represent the pixel location is

$$S_{\mathbf{x}_0,\tau,T} = \{I(\mathbf{x},t) | \, \|\mathbf{x} - \mathbf{x}_0\| < X, \tau \leq t < \tau + T\}. \quad (5)$$

For simplicity, we use a 3 dimensional color histogram as our background model:

$$p_{\mathbf{u},S_{\mathbf{x}_0,\tau,T}} = n \sum_{s \in S} \delta(b(s) - u) \quad (6)$$

where u is the histogram bin, $u \in U \subset \mathbb{R}^3$, and n is a normalizing constant so that $\sum_{u \in U} p_u = 1$. In our experiments, $U = [1\ 1\ 1] \times [16\ 3\ 3]$.

To classify a pixel as foreground, we create another histogram q such that the set of pixels considered are only those from the image I at time t:

$$q_{u,S_{\mathbf{x}_0,t,1}} = n \sum_{s \in S} \delta(b(s) - u) \quad (7)$$

where n is a normalizing constant so that $\sum_{u \in U} q_u = 1$.

For a pixel to be considered foreground, we require that the distributions, $\mathbf{p}$ and $\mathbf{q}$ be sufficiently different, $d(\mathbf{p},\mathbf{q}) < \eta_d$, or it is connected to a region and is different enough, $d(\mathbf{p},\mathbf{q}) < \eta_s$, where $\eta_s < \eta_d$.

We define a region, R_t, as a set of connected pixels from a single image captured at time t. For each region, the following must be true:

$$\begin{array}{rrrrrr} \exists & x & \in & R_t, d(p_x,q_x) & < & \eta_d \\ \forall & x & \in & R_t, d(p_x,q_x) & < & \eta_s \\ & & & \eta_d & < & \eta_s \end{array} \quad (8)$$

5.2. Object Labels

We considered each regions extracted from the previous section as object instances, and therefore rather than labeling each pixel independently, we provide a single object label and a single category label for each region. We consider two regions from consecutive frames to have the same object label if their regions overlap in consecutive frames or if their appearance indicate that they are from the same category model.

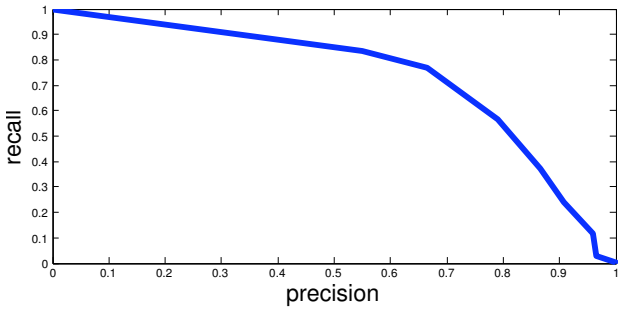

Figure 4: Precision-Recall curve for region detection. Instances that are not "good" are discarded from the calculations.

The first case results in a complex object model that captures the appearances of an object from different viewpoints. The appearance of the object o is modeled as the set of histograms described in detail in Section **??**. While in Section **??** we used all the pixels *in a bounding box* to represent the object, we use all the pixels *within the region extracted from the foreground/background labeling procedure*.

The second case uses category models (if they exist) to aid in recognizing objects across frames. For all the regions that are labeled as an object o, $\{R_{t_1}, R_{t_1+1}, ..., R_{t_2}\}$ we define two functions, $Start(o) = t_1$ and $End(o) = t_2$. For objects where $End(o_i) + 1 = Start(o_i)$, we merge object o_j and o_i into a single object o_i if the category label for o_j and o_i are identical. We therefore take advantage of the multiple views provided by the category models in determining object labels.

5.3. Category Labels

We use the same metric described in Section **??** to measure the distance between the two object models. For each pair of objects, o_i and o_j, we construct a corresponding H_{o_i} and H_{o_j}. We define their similarity as

$$sim(o_i, o_j) = D(H_{o_i}, H_{o_j}). \quad (9)$$

If $sim(o_i, o_j) > \eta_c$, o_i and o_j are assigned the category label.

We represent the appearance model of a category as:

$$H_c = \bigcup_{o \in O_c} H_o \quad (10)$$

where O_c is a set of objects labeled as category c.

5.4. Results

For each 600 consecutive frames in the 3600 frame image sequence, we build a background model and perform background subtraction to estimate the difference between

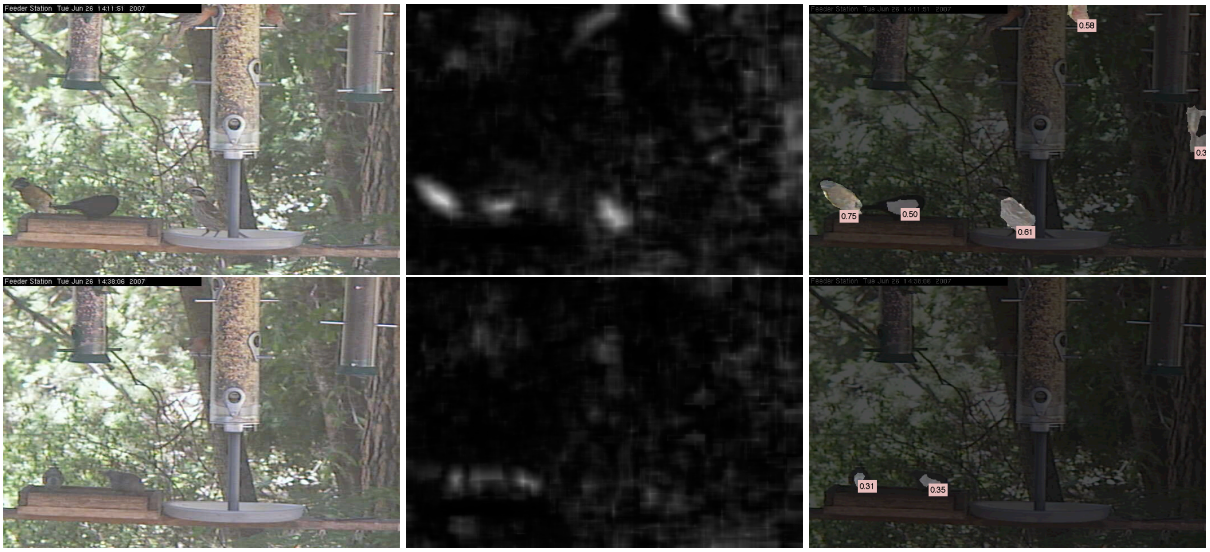

Figure 5: Extracting regions from two frames. From left to right: the original frame, a pixel wise comparison against the background model where white is very different from the background, and the highlighted regions used for object and category classification. The pink labels beneath the regions indicate the detection threshold η_d needed for the region to be considered foreground.

a given pixel location (x, t) and the background. Sample images and the difference image is shown in Figure **??**. For ease in viewing, the difference image is $1 - d(a, b)$ so that a white indicates a large difference from the background, and black indicates a small difference from the background. On the far right, we show the regions selected, and the minimum η_d needed to detect the region. The frame shown in the top row detects many birds at a fairly high threshold, but is still missing a bird in the top middle area of the image. It also mistakes the bark of the tree as a bird, due to unmodelled lighting changes. The frame shown in the bottom row show a few birds that require a fairly low threshold to be detected.

More quantitative results are shown in the precision-recall curve in Figure **??**. For recall, we say an object is recalled if a detected region overlaps with a bounding box from the labeled ground truth data of "good" instances. For precision, we say a region is detected correctly if it overlaps with a ground truth label, disregarding instances that are not "good."

While the number of instances, objects, and categories depends on the thresholds chosen, we present a possible set of categories extracted from this method. When $\eta_d = 0.35$ and $\eta_s = 0.15$, Figure **??** shows a few sample object instances that are grouped due to their overlapping regions in consecutive frames. It can be seen from these samples that it is possible to capture a range of appearances using this heuristic.

While we start at 9710 instances which form an initial set of 3300 objects, only 677 objects have 3 or more instances. We take these 677 objects and group 365 objects into clusters where $\eta_c > .94$. We end up with 58 categories with $2+$ objects, 16 that contain 99 background objects, and 42 object categories which contain the remaining 265 images. A few of the categories found are shown in Figure **??**. We end up with 8 out of the 16 categories cleanly segmented into separate clusters in that only instances from a category are clustered together. There are 17 clusters that contain multiple species in a single cluster.

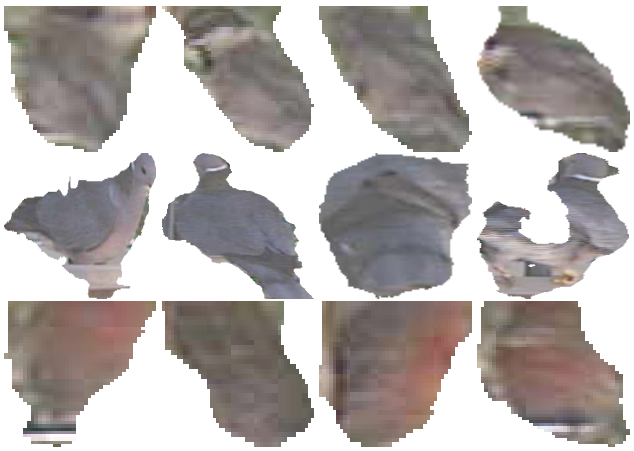

Figure 7: Normalized frames used to form an multi-view object model. From top to bottom, these are examples of category 1, 3, 4, 13.

6. Conclusion

We demonstrated that utilizing multiple views naturally captured from image sequences can result in improved performance for object category classification. Our approach is

Figure 6: The clusters labeled "$3, 4, 9, 13, 15, 16$" are a few of the categories that was cleanly segmented from the image sequence. A "*" category is one where our approach grouped multiple species into a single cluster. The middle image, where one bird is partially occluded by another bird, is one of the reason for this confusion. A region contains the color features from two different bird categories resulting in both species being combined into one category. The "bg" category is an example of one of the clustered background objects.

capable of handling unbalanced object representation, such that there is little bias for objects that stay in the same position for long periods of time. We presented the performance of this approach in a supervised and unsupervised setting. We also shared a data set that contains many of the nuisances that pertain to natural environment monitoring. In the future, we plan to address some of the nuisances not currently addressed, such as, but not limited to, occlusions and accurate segmentation. We will also expand our data set to include more natural scenes.

7. Acknowledgements

This material is based upon work supported by the Center for Embedded Networked Sensing (CENS) under the National Science Foundation (NSF) Cooperative Agreement CCR-012-0778 and #CNS-0614853 and by the ONR under award #ONR 67F-1080868.

References

[1] H. Bay, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. In *ECCV*, pages 404–417, 2006.

[2] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV: International Workshop on Statistical Learning in Computer Vision*, 2004.

[3] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR,Workshop on Generative-Model Based Vision*, 2004.

[4] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. CVPR*, 2003.

[5] B. Fulkerson, A. Vevaldi, and S. Stefano Soatto. Localizing ob jects with smart dictionaries. In *ECCV*, 2008.

[6] K. Grauman and T. Darrell. The pyramid match kernel:discriminative classification with sets of image features. In *ICCV*, volume 2, pages 1458– 1465, 2005.

[7] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, G. Griffin and A. Holub and P. Perona, 2007.

[8] X. He, R. Zemel, and M. C.-P. nán. Multiscale conditional random fields for image labeling. In *Proc. CVPR*, 2004.

[9] T. Ko, S. Soatto, and D. Estrin. Background subtraction on distributions. In *ECCV*, 2008.

[10] B. Leibe, K. Micolajckzyk, and B. Schiele. Efficient clustering and matching for object class recognition. In *Proc. BMVC*, 2006.

[11] L.-J. Li, G. Wang, and L. Fei-Fei. Optimol: automatic online picture collection via incremental model learning. In *CVPR*, pages 1–8, 2007.

[12] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[13] M. Marszałek and C. Schmid. Accurate object localization with shape masks. In *Proc. CVPR*, 2007.

[14] J. Meltzer, M.-H. Yang, R. Gupta, and S. Soatto. Multiple view feature descriptors from image sequences via kernel principal component analysis. In *Proc. ECCV*, 2004.

[15] J. Mutch and D. G. Lowe. Multiclass object recognition with sparse, localized features. In *CVPR*, pages 11–18, 2006.

[16] J. Ponce, T. L. Berg, M. Ehveringam, D. A. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, B. C. Russell, A. Torralba, C. K. I. Williams, J. Zhang, and A. Zisserman. *Dataset Issues in Object Recognition*. Springer-Verlag Lecture Notes in Computer Science, 2006.

[17] J. Shotton, J. Winn, C. Rother, and A. Criminisi. *TextonBoost*: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *Proc. ECCV*, 2006.

[18] P. Viola and M. Jones. Robust real-time object detection. In *ICCV: 2nd International Workshop on Statistical and Computational Theories of Vision - Modeling, Learning, Computing, and Sampling*, 2001.

[19] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. *IJCV*, 63(2):153–161, 2005.

[20] G. Wang, Y. Zhang, and L. Fei-Fei. Using dependant regions or object categorization in a generative framework. In *CVPR*, pages 1597–1604, 2006.

[21] M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *Proc. ECCV*, 2000.

[22] A. Y. Yang, J. Wright, Y. Ma, and S. S. Sastry. Unsupervised segmentation of natural images via lossy data compression. *CVIU*, 110(2):212–225, 2008.

[23] H. Zhang, A. Berg, M. Maire, and J. Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *CVPR*, volume 2, pages 2126 – 2136, 2006.