Object-Centric Spatio-Temporal Pyramids for Egocentric Activity Recognition

BMVC 2013 Submission # 131

Abstract

Activities in egocentric video are largely defined by the objects with which the camera wearer interacts, making representations that summarize the objects in view quite informative. Beyond simply recording how frequently each object occurs in a single histogram, spatio-temporal binning approaches can capture the objects' relative layout and ordering. However, existing methods use hand-crafted binning schemes (e.g., a uniformly spaced pyramid of partitions), which may fail to capture the relationships that best distinguish certain activities. We propose to learn the spatio-temporal partitions that are discriminative for a set of egocentric activity classes. We devise a boosting approach that automatically selects a small set of useful spatio-temporal pyramid histograms among a randomized pool of candidate partitions. In order to efficiently focus the candidate partitions, we further propose an "object-centric" cutting scheme that prefers sampling bin boundaries near those objects prominently involved in the egocentric activities. In this way, we specialize the randomized pool of partitions to the egocentric setting and improve the training efficiency for boosting. Our approach yields state-of-the-art accuracy for recognition of challenging activities of daily living.

1 Introduction

Egocentric computer vision entails analyzing images and video that originate from a wearable camera, which is typically mounted on the head or chest. Seeing the world from this first-person point of view affords a variety of exciting new applications and challenges, particularly as today's devices become increasingly lightweight and power efficient. For example, in the life-logging setting, a user constantly captures his daily activity, perhaps to share it with others, or to personally review it as a memory aid [11]. Daily logs from a wearable camera also have compelling applications for law enforcement and defense, where an archive of the first-person point of view may contain valuable forensic data. Furthermore, in augmented reality applications, a user could be shown on an associated display (e.g., Google Glass) valuable meta-data about the objects or events he observes in real time, such as product reviews for an object he handles in the store. Egocentric video analysis also has potential to determine how well a person can complete physical daily living tasks, thereby enabling new forms of tele-rehabilitation [15, 21].

Nearly all such applications demand robust methods to recognize activities and events as seen from the camera wearer's perspective. Whereas activity analysis in the traditional "third person" view is often driven by human body pose, in egocentric video activities are largely defined by the *objects* that the camera wearer interacts with. Accordingly, high-level

representations based on detected objects are a promising way to encode video clips when learning egocentric activities [6, 7, 21, 24]. In particular, recent work explores a "bag-of-objects" histogram of all objects detected in a video sequence, as well as a spatio-temporal pyramid extension that captures the objects' relative temporal ordering [21]. Coupled with standard discriminative classifiers, this representation shows very good results; notably, it outperforms histograms of local space-time visual words, a favored descriptor in current third-person activity recognition systems.

However, existing methods that pool localized visual features into space-time histogram bins do so using hand-crafted binning schemes, whether applied to egocentric video or otherwise. For example, the spatial pyramid widely used for image classification [18] is extended to space-time in [4, 21], using a hierarchy of regularly sized volumetric bins to pool the detected features at different granularities. In [17], a series of coarse partitions are defined (dividing the video into thirds top to bottom, etc.), then aggregated by summing kernels. The problem with defining the spatio-temporal bins *a priori* is that they may not offer the most discriminative representation for the activity classes of interest. That is, the hand-crafted histogram bins may fail to capture those space-time relationships between the component objects (or other local features) that are most informative.

To overcome this limitation, we propose to *learn* discriminative spatio-temporal histogram partitions for egocentric activities. Rather than manually define the bin structure, we devise a boosting approach that automatically selects a small set of useful spatio-temporal pyramid histograms among a randomized pool of candidate partitions. In this way, we identify those partitions that most effectively pool the detected features (in this case, the detected objects). Since training time for boosting grows linearly with the number of candidates, relying on purely random space-time cuts can be computationally expensive. Therefore, we further propose a way to meaningfully bias the partitions that comprise the candidate pool. We devise an *object-centric* cutting scheme that prefers sampling bin boundaries near objects involved in the egocentric activities. In particular, our method is more likely to sample partitions that cut through video regions containing "active" objects [21] (e.g., the open microwave, the pot handled on the stove), thereby concentrating layout information on the key interactions. As a result, we focus the randomized pool of space-time partitions to the egocentric setting while also improving training efficiency.

We apply our method to the challenging Activities of Daily Living dataset, and show that the proposed method improves the state of the art. The results show the value of learning discriminative space-time partitions, compared to both bag-of-words or existing spatio-temporal pyramids. Furthermore, we demonstrate the key role played by object-centric cuts in terms of focusing the candidate pyramids.

1.1 Related Work

For generic (non-egocentric) activity recognition, methods based on tracked limbs and body shapes (e.g., [22, 23, 25]) analyze human actions in a model-based way. More recently, model-free alternatives based on low-level descriptors of gradients and optical flow have been explored (e.g., [17, 20, 26]), attempting to directly learn the motion and appearance patterns associated with an activity. A fairly standard pipeline has emerged analogous to the bag-of-visual-words approaches often employed for image classification: detect space-time interest points, extract local descriptors for each point, quantize to space-time visual words, then represent the entire video with a histogram counting how often each word appears.

Since a pure bag-of-words lacks any notion of ordering, researchers have further drawn

inspiration from spatial pyramid image representations [2, 18] to construct space-time histograms from subcells within the video volume. These subcells count features appearing in particular regions of the video, and as such they can flexibly capture the relative layout. In [17], a set of spatio-temporal bin structures is defined that uses six possible spatial grids and four temporal binning schemes, resulting in a total of 24 possible spatio-temporal partitions. The histograms from all partitions are combined by a summed kernel. In [4], a space-time pyramid with a hierarchy of regularly sized cubic bins is constructed, and used to pool the features at multiple resolutions. A related strategy is to hierarchically bin neighboring local features and discriminatively learn which space-time weightings are most informative [16]. For the egocentric setting, a temporal pyramid that divides the video into half along the temporal axis (and uses no spatial partitions) is proposed, and used to histogram object detector outputs [21]. Unlike any of these approaches, our idea is to learn which pyramid structures are discriminative.

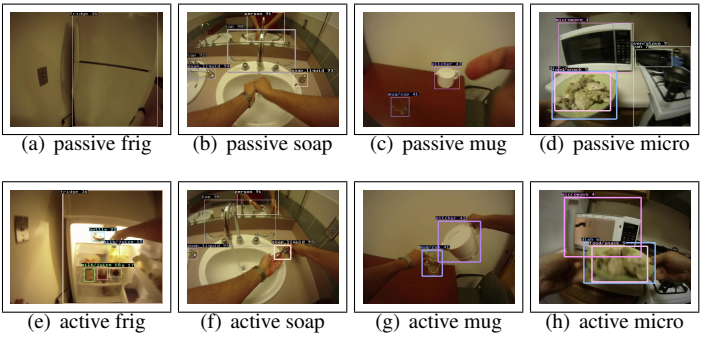
Prior work on activity recognition from wearable cameras [6, 10, 29] often considers a particular environment of interest (like a particular kitchen) for which individual familiar objects are informative, and some explores the role of additional sensors such as Inertial Measurement Units [28]. In contrast, we are interested in recognizing activities by a camera wearer moving about multiple environments and without additional sensors or pre-placed objects of interest. Such a setting is also tackled in the recent work of [21]. We leverage their finding that object-based representations are critical for egocentric activity, and also use their idea of "active" objects. However, while that method uses a simple hand-crafted histogram structure consisting of two temporal bins (one for the first half of the video, one for the second half of the video), we propose to learn a boosted combination of discriminative histogram partitions. Through direct comparison in our results, we show that our idea achieves substantially more accurate activity recognition.

Aside from recognizing activities, egocentric video analysis also entails interesting problems in object recognition [24], event segmentation [5], novelty detection [1], summarization or unsupervised discovery [13, 14, 19], and the relationship between gaze and activity [8].

Our approach to learn discriminative space-time bin structures for activity recognition takes inspiration from methods for image classification that select discriminative spatial bins [12, 27]. In [27], the spatial grid and classifier are jointly learned using a maximum margin formulation, and in [12], boosting is used to select useful randomized spatial partitions. Both methods target scene classification from images. In contrast, we learn discriminative partitions in space-time for activity recognition. To our knowledge, no prior work considers discriminative learning of spatio-temporal partitions, whether for egocentric or non-egocentric data. Furthermore, our idea to bias the randomized partitions to focus on active objects is novel, and is critical for recognition results, as we will show in experiments.

2 Approach

The overall approach works as follows. Given a set of egocentric training videos labeled according to their activity class, we first run object detectors on the frames to localize any objects of interest—both those that are "passive" and those that are "active" in an interaction with the camera wearer. We then construct a series of candidate space-time pyramids, in which each axis-aligned bin boundary is translated by some random shift. The random shifts are non-uniform; they are sampled using the distribution of all active object coordinates in the training data. Given this candidate pool of pyramids, we compute the corresponding



(a) passive frig (b) passive soap (c) passive mug (d) passive micro

(e) active frig (f) active soap (g) active mug (h) active micro

Figure 1: Example passive and active instances of four objects in ADL [21].

series of object histograms for each training video, where a detected object is counted in the space-time bin its center occupies. Then, we apply multi-class boosting to select a subset of discriminative pyramid structures based on how well they can be used to classify the activities of interest. At the end, we have a strong classifier that can predict the activity labels of new videos, using only those randomized pyramids selected by the learning algorithm.

The following subsections explain each of these steps in more detail.

2.1 Detecting Active and Passive Objects

Our goal is to robustly predict what type of activity is occurring in an egocentric video clip. In contrast to traditional third-person video, egocentric actions are inherently defined by the objects the user is interacting with. Therefore, our representation is built on the pattern of objects that appear in space and time. Specifically, the space-time pyramids we learn will count the frequency with which each object category appears in particular space-time regions.

Following [21], we make a distinction between active and passive instances of a given object category. As noted in [21], objects' appearance can often change dramatically when the object is being interacted with. For example, the refrigerator looks quite different when one passes by it closed, versus when one opens the door to grab some food. Therefore, we train different deformable part model [9] detectors for active and passive versions of various objects of interest.[1] Figure 1 depicts example frames extracted from the Activities of Daily Living (ADL) [21] video sequences that show the visual differences between passive and active versions of four example objects. In contrast to prior work, we exploit the active/passive object distinction to provide a helpful bias regarding where space-time partitions ought to be sampled, as we describe in the next section.

Once all object detectors have been applied to all frames in the training or test video, we have an (x, y, t) coordinate for the bounding box center of each detected object. Associated with each coordinate is its (predicted) object class (frig, microwave, etc.)

2.2 Sampling Randomized Object-Centric Space-Time Pyramids

Once we have the predicted object locations in all training videos, we are ready to construct space-time histogram pyramids. A space-time pyramid will consist of multiple levels of bins,

[1] We use the public code and detection outputs provided by [21].

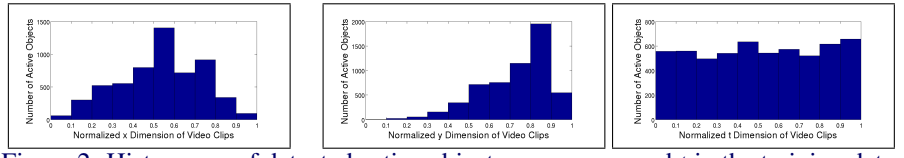


Figure 2: Histograms of detected active objects across x, y, and t in the training data.

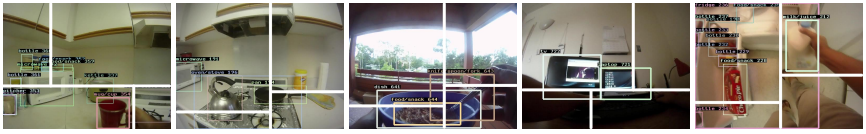
from coarse to fine. For each bin, we record how many times each object class appears in its respective region of the video. Then we concatenate these histograms over all pyramid levels to get a single descriptor for the video. Thus, for a pyramid with T total bins and a bank of D total object detectors, the dimensionality of the entire descriptor will be TD. Whereas past work uses a pyramid with uniformly placed bins [9, 21], we propose to generate randomized pyramids and then learn their most discriminative combination.

First we describe how to generate the randomized space-time pyramids (RSTP) *without* the object-centric bias. We consider each dimension (x, y, t) in turn in a round-robin fashion to generate a cut (i.e., place a bin boundary). Each cut is axis-aligned, meaning we use random shifts, but no random rotations. We normalize all dimensions of an input video to length 1. Then we sample a number uniformly at random in $[0, 1]$, and use it to place the randomized cut in the current dimension. Note that as we work our way recursively down the resulting tree, each subsequent cut is appropriately constrained by the span of its parent bin. Level 0 of the pyramid is the entire video clip volume; level i consists of all 8^i bins of depth i.

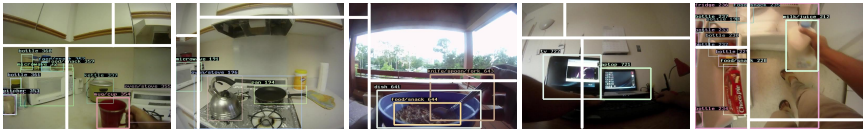
While boosting gives an automated way to select informative pyramids, its training time depends linearly on the number of candidates we include in the pool. With so many possible randomized pyramids, the search space is extremely large. Thus, intuitively, we can expect to pay a very high training cost to evaluate sufficiently many randomized pyramids to get good results.

To avoid an excessive search, we focus the candidate pool in a way that is meaningful for egocentric data. Rather than sample cuts uniformly at random, our idea is to sample the cuts according to the distribution of active objects as they appear in the training videos. We refer to these as *object-centric cuts* (OCC). Specifically, we construct the empirical distribution of all active object occurrences, per dimension. Then, when selecting each randomized cut, we sample its position according that distribution. In this way we get pyramids that emphasize video regions likely to characterize the interactions between the camera wearer and objects. For each pyramid, after generating one OCC per dimension, we generate all subsequent child cuts using a uniform distribution. We found this was more effective than using OCC's at all levels, likely because we risk overfitting once bins are quite small in volume. Note that while the OCC's are biased by active objects only, we still count both active and passive objects in the resulting histograms.

Figure 2 shows the active object distributions for the ADL dataset we use to validate our approach. We see that active objects tend to appear in the lower center field of view. This conforms to our expectations, because active objects are close to the hands, which appear in the bottom portion of most frames from the chest mounted camera. Furthermore, there is a slight bias favoring the right side of the field of view, likely because many camera wearers are right-handed. Finally, we also observe that the distribution of active objects across the temporal dimension is nearly uniform; this reflects that we use object occurrences across all action types.



(a) Object-centric cuts



(b) Uniformly random shifts

Figure 3: Example partitions using either object-centric (a) or uniformly sampled randomized cuts (b). Note that for display purposes we show cuts on example 2D frames, but all cuts are 3D in space-time. Using the proposed object-centric cuts, we better focus histograms surrounding the human-object interactions.

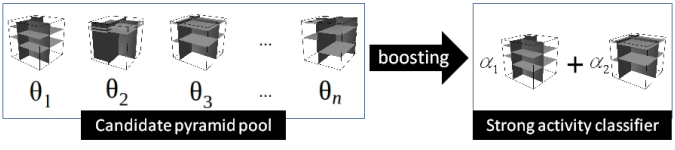


Figure 4: We take a pool of randomized space-time pyramids with object-centric cuts, and use boosting to select those that are most discriminative for egocentric activity recognition.

Figure 3 shows some example frames with randomized shifts sampled using our object-centric strategy (a) or the simpler uniform strategy (b). The object detections shown are from the ADL repository [21]. We see how OCC's successfully focus the histograms on regions in space-time where human-object interactions occur. As a result, they may offer more discriminative cues that will be useful to the boosted classifier.

2.3 Boosting Discriminative Space-Time Pyramids

Finally, having constructed our object-centric pool of randomized pyramids, we are ready to apply boosting to select those that are most discriminative for the given activity recognition task (see Figure 4). Boosting is a general learning algorithm in which one can combine a series of "weak" classifiers (better than chance) to form a single "strong" classifier. In each round of boosting, the training examples are reweighted to emphasize training errors on those examples that were misclassified by weak classifiers selected in previous rounds. Here, our weak classifiers are non-linear (polynomial kernel) SVMs trained using one RSTP with OCC's. We essentially use boosting to both select useful features (pyramids) and build the composite strong classifier.

For our implementation, we use the Stagewise Additive Modeling using a Multi-class Exponential loss function (SAMME) boosting approach of [30], which naturally extends the original AdaBoost algorithm to the multi-class case without reducing it to multiple two-class problems. We chose it due to its advantage of avoiding training individual classifiers

for many one-vs.-rest (or one-vs.-one) problems and due to its ease of implementation.

SAMME boosting works as follows in our setting. We take as input a collection of N labeled training videos, where (V_i, c_i) denotes a video clip and its associated ground-truth activity label (drinking water, washing dishes, etc.). We generate a pool of M candidate RSTP's $\{\theta_1, \theta_2, ..., \theta_M\}$, as described above. For each RSTP θ, we compute the corresponding histogram for each training example, using an object's bounding box center position (x, y, t) to increment the appropriate bins. We concatenate the histograms from all levels to create a single feature vector for each V_i and each θ. Then we initialize a weight w_i for each training example V_i that is inversely proportional to the number of points with the same label as V_i. Giving larger weights to training examples of infrequently occurring actions helps to mitigate bias from imbalanced training data.

Next we train a separate weak multi-class SVM classifier (using the one-vs.one protocol in LIBSVM [2]) on the feature vectors resulting from representing the training data using each candidate partition pattern. During each round of boosting we select the candidate partition θ_j that has the minimum weighted training error. SAMME computes a weight for θ_j based on how many training examples were misclassified using f_{θ_j}, the SVM classifier that was trained using the representation of the training data under θ_j. At the end of each boosting iteration, we update the weights for each training example. Training examples that were previously misclassified are assigned higher weights to encourage correct classification in future boosting rounds. Finally, we generate the final strong classifier F, which maximizes a weighted sum of correct classifications produced by each weak classifier. Algorithm 1 summarizes these steps.

Given a novel input video, we run the object detectors, then extract only those RSTP histograms that were selected by boosting, and apply F to predict its activity label.

Algorithm 1: Training a space-time pyramid classifier with boosting

INPUT:

- N labeled training videos $\Phi = \{(V_i, c_i)\}_{i=1}^{N}$
- A pool of M partition patterns $\Theta = \{\theta\}$

OUTPUT:

- A strong video classifier F. For an unlabeled video V, $c = F(V)$ is the predicted label for V.

1. For each pattern $\theta \in \Theta$:
   - Represent each $V_i \in \Phi$ using θ and train an SVM classifier f_θ on the resulting feature vectors.
2. Initialize:
   - A weight vector w with $w_i = \frac{1}{CN_{c_i}}$ for each video where N_{c_i} is the number of videos with label c_i, and C is the number of distinct action labels.
   - Current boosting round $j = 0$.
3. For each round of boosting:
   - Increment j and re-normalize the weight vector w.
   - For each pattern θ, compute its weighted classification error: $e_\theta = w \cdot \mathbf{I}(f_\theta(V) \neq c)$
   - Choose the pattern θ_j with minimum weighted classification error e_j.
   - Compute the weight for θ_j as: $\alpha_j = \log\frac{1-e_j}{e_j} + \log(C-1)$
   - Update the weight vector w: $\forall i : w_i = w_i \cdot \exp(\alpha_j \cdot \mathbf{I}(f_{\theta_j}(V_i) \neq c_i))$.
   - Generate the current strong classifier as: $F(V) = \text{argmax}_c \Sigma_{m=1}^{j} \alpha_m \cdot \mathbf{I}(f_{\theta_m}(V) = c)$

BoW	Bag-of-objects	TempPyr [21]	Boost-RSTP	Boost-RSTP+OCC (ours)
16.5%	34.9%	36.9%	33.7%	**38.7%**

Table 1: Overall classification accuracy on ADL. Our method improves the state of the art.

3 Results

To validate our method, we use the Activities of Daily Living (ADL) dataset [21]. It is the largest available egocentric dataset for activity recognition, and to our knowledge, the most diverse and realistic. It consists of hundreds of egocentric clips (roughly 10 hours of video in total) collected from 20 people performing 18 actions in their own homes. These naturally occurring actions are often related to hygiene or food preparation, e.g., combing hair, brushing teeth, doing laundry, washing dishes, etc. The authors also provide the object detector outputs from a part-based model [9] for 26 object classes, which we directly use as input to our method. The objects include household items. Five of the 26 detectors are for active versions of certain objects (namely, refrigerator, microwave, mug, oven/stove, and soap liquid).[2] We use the authors' publicly available code to run their method for comparison.

Throughout, we use five rounds of boosting and populate our candidate pool with 4-level pyramids. Preliminary experiments showed that the finer-grained (4-level) pyramids were more often selected by boosting than their coarser 3-level counterparts, so we focus the pool accordingly for all results.

We follow the exact evaluation protocol given in [21]. Specifically, we evaluate recognition accuracy using leave-one-person out: we test on videos from each person i in turn, having trained on all remaining people. We exclude the first 6 people, since their data was used to train the object detectors.

Table 1 shows the results, in terms of the average recognition rate over all 18 action classes. We compare our boosted RSTP+OCC approach to four baselines. The first baseline, bag-of-words (BoW), uses space-time interest points and HoG/HoF visual words, and represents what is now a standard representation for third-person action recognition. The second baseline uses a bag-of-objects. The third baseline, the Temporal Pyramid, is the method proposed in [21], and represents the state of the art on this dataset. The fourth baseline, RSTP, is just like the proposed approach only it lacks the object-centric cuts.

Our approach outperforms all four baselines and improves the state of the art. Compared to BoW, we have the advantage of high-level object-based features. While the Temporal Pyramid [21] also has this benefit, it is weaker than our method due to its reliance on a hand-crafted pyramid structure. Notably, the proposed object-centric cuts are essential for our strong recognition result. Simply using boosting with purely randomized partitions (RSTP) is noticeably weaker. This supports our claim that it is useful to bias bins according to object interactions for egocentric data.

Looking more closely at our method's predictions, we find it has particularly good accuracy for "combing hair" and "drying hands/face". This suggests that the learned bins were able to usefully isolate the regular space-time relationships these actions exhibit. On the other hand, we often confuse "making tea" and "making coffee", likely because they involve the same active objects. Furthermore, since the distributions of objects across space-time are

[2]The ADL dataset has been modified since the publication of [21]; because of this, running the published code gives slightly lower accuracy than the originally published numbers. We use the modified version of the dataset available from the authors webpage at the time of writing to run all experiments.

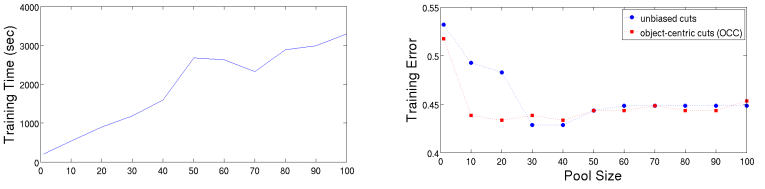


Figure 5: Left: Training time as a function of pool size. Right: Training error as function of pool size, for both uniformly sampled random shifts and the proposed object-centric partitions. By focusing on object-centric shifts, we can achieve a stronger classifier with a smaller total pool, which improves training efficiency.

similar for both, and kettles and tea bags are not modelled as active objects, it is difficult for our boosting algorithm to select partitions that are discriminative for these classes. An extension of our method which allows selecting partitions on a per-class basis could allow for more fine-grained control and could help mitigate such issues, though would be more expensive. We leave this as future work.

Compared to the Temporal Pyramid [21], we find out method is especially stronger for "combing hair", "brush teeth", "dental floss". This indicates that our learned spatial cuts are essential in scenes with similar objects appearing across different actions, as is the case with these bathroom-based activities. For instance, while combing hair, floss or toothpaste might appear on the counter, but floss or toothpaste would appear higher in the field of view when actually in use.

Figure 5 emphasizes the benefits of object-centric cuts. On the left, we show the training time of running boosting with increasingly larger pools of candidate pyramids, averaged over five runs; run-time increases linearly with pool size. On the right, we show the training error as a function of the pool size. As desired, we see that the object-centric cuts lead to lower error with smaller pool sizes, compared to the unbiased RSTP's. Essentially, our method focuses the pool on those candidates that *a priori* have good chance at capturing discriminative aspects of the object distribution in space-time. Thus, fewer total candidates must be explored to find good ones, and we can train the models with less total training time.

4 Conclusions and Future Work

Our main contribution is two-fold. We show how to learn the most discriminative partition schemes for spatio-temporal binning in action recognition, and we introduce object-centric cuts for egocentric data. Our approach improves on the current state of the art for recognizing activities of daily living from the first person viewpoint, and our experiments demonstrate the positive impact of taking active object locations into account via object-centric cuts.

In future work, we intend to investigate ways of learning the most discriminative partition schemes on a per-class basis. Additionally, it may be possible to incorporate other related sampling biases. For example, our current strategy only implicitly accounts for the positions of hands via our OCC's, but it may be useful to incorporate explicit features about the hands. While we obtain good results using cuts that are planar and axis-aligned, one could easily extend the approach to populate the pool with non-linear cuts and/or randomized rotations. Such a method would make histogram computation more expensive, but may yield the discriminative partitions necessary for more fine-grained decisions.

References

[1] O. Aghazadeh, J. Sullivan, and S. Carlsson. Novelty detection from an egocentric perspective. In *CVPR*, 2011.

[2] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. In *CIVR*, 2007.

[3] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, pages 27:1–27:27, 2011.

[4] J. Choi, W. Jeon, and S.-C. Lee. Spatio-temporal pyramid matching for sports videos. In *ACM Multimedia*, 2008.

[5] B. Clarkson and A. Pentland. Unsupervised clustering of ambulatory audio and video. In *ICASSP*, 1999.

[6] A. Fathi, A. Farhadi, and J. Rehg. Understanding egocentric activities. In *ICCV*, 2011.

[7] A. Fathi, Xiaofeng Ren, and J.M. Rehg. Learning to recognize objects in egocentric activities. In *CVPR*, 2011.

[8] A. Fathi, Y. Li, and J. Rehg. Learning to recognize daily actions using gaze. In *ECCV*, 2012.

[9] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008.

[10] M. Hanheide, N. Hofemann, and G. Sagerer. Action recognition in a wearable assistance system. In *ICPR*, 2006.

[11] S. Hodges, E. Berry, and K. Wood. Sensecam: A wearable camera which stimulates and rehabilitates autobiographical memory. *Memory*, 2011.

[12] Y. Jiang, J. Yuan, and G. Yu. Randomized spatial partition for scene recognition. In *ECCV*, 2012.

[13] N. Jojic, A. Perina, and V. Murino. Structural epitome: A way to summarize one's visual experience. In *NIPS*, 2010.

[14] K. Kitani, T. Okabe, Y. Sato, and A. Sugimoto. Fast unsupervised ego-action learning for first-person sports video. In *CVPR*, 2011.

[15] B. Kopp, A. Kunkel, H. Flor, T. Platz, U. Rose, K. Mauritz, K. Gresser, K. McCulloch, and E. Taub. The arm motor ability test: reliability, validity, and sensitivity to change of an instrument for assessing disabilities in activities of daily living. *Archives of physical medicine and rehabilitation*, 78(6):615–620, 1997.

[16] A. Kovashka and K. Grauman. Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In *CVPR*, 2010.

[17] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[18] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[19] Y. Lee, J. Ghosh, and K. Grauman. Discovering important people and objects for egocentric video summarization. In *CVPR*, 2012.

[20] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *CVPR*, 2009.

[21] H. Pirsiavash and D. Ramanan. Detecting activities of daily living in first-person camera views. In *CVPR*, 2012.

[22] D. Ramanan and D. Forsyth. Automatic annotation of everyday movements. In *NIPS*, 2003.

[23] C. Rao and M. Shah. View-invariance in action recognition. In *CVPR*, 2001.

[24] X. Ren and C. Gu. Figure-ground segmentation improves handled object recognition in egocentric video. In *CVPR*, 2010.

[25] M. Rodriguez, J. Ahmed, and M. Shah. Action MACH a spatio-temporal maximum average correlation height filter for action recognition. In *CVPR*, 2008.

[26] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: a local svm approach. In *ICPR*, 2004.

[27] G. Sharma and F. Jurie. Learning discriminative spatial representation for image classification. In *BMVC*, 2011.

[28] E. Spriggs, F. D. la Torre, and M. Hebert. Temporal segmentation and activity classification from first-person sensing. In *CVPR Workshop on Egocentric Vision*, 2009.

[29] S. Sundaram and W. Cuevas. High level activity recognition using low resolution wearable vision. In *IEEE Workshop on Egocentric Vision*, 2009.

[30] J. Zhu, S. Rosset, H. Zou, and T. Hastie. Multi-class adaboost. *Statistics and Its Interface*, 13, 2009.