RECOGNITION OF 48 HUMAN BEHAVIORS FROM VIDEO

G.J. Burghouts, H. Bouma, R.J.M. den Hollander, S.P. van den Broek, K. Schutte

TNO, Oude Waalsdorperweg 63, 2597 AK The Hague, The Netherlands
gertjan.burghouts@tno.nl

KEYWORDS: visual pattern recognition, human behavior, spatiotemporal features, events, classification, random forest, tag propagation.

ABSTRACT:

We have developed a system that recognizes 48 human behaviors from video. The essential elements are (i) inference of the actors in the scene, (ii) assessment of event-related properties of actors and between actors, (iii) exploiting the event properties to recognize the behaviors. The performance of our recognizer approaches human performance, yet the performance for unseen variations of the behaviors needs to be improved.

1. INTRODUCTION

This paper is about the recognition of 48 human behaviors from 2,588 short video clips of about 10-30 seconds, given a learning set of 3,480 video clips. The behaviors and their prevalence in the dataset (based on human annotations) are listed in Figure 1. For some behaviors, several tens of positive examples are available, while for others there are a few thousand positive examples. The behaviors vary from simple behaviors of one person (e.g. walk) where others involve two or more persons (e.g. follow). There are behaviors that are defined by the involvement of some item (e.g. give), or an interaction with the environment (e.g. leave).

This challenge cannot be solved by using general spatiotemporal features and learning a straightforward classifier. The reason is that each behavior will have several variants which may be encountered with varying actors and in various conditions. It can be argued that one behavior may have about 10 variants. That means that for a corpus of 48 behaviors we have approximately 500 variants. For a typical statistical learning problem, 50 samples per class are used [1]. In total we would need 25,000 clips to solve the problem of recognizing 48 human behaviors. DARPA has provided us with 3,480 clips to solve the challenge. This is an order of magnitude lower than what we would need from a statistical pattern recognition point of view.

Solutions are needed to solve the recognition problem without being hindered by the insufficient training set. That is the focus of this paper, in which we highlight three contributions. We have incorporated as much world knowledge as possible in order to reduce the learning. The world knowledge is integrated into our system at two levels: inference of the actors in the scene (see *Entities*), assessment of event-related properties of actors and between actors (see *Event Properties*). The third contribution is to optimally exploit the available yet limited training set by comparing the current video to all videos from this set (see *Random-Forest Tag-Propagation*).

We demonstrate the performance of our system by comparing to human annotations (see *Recognition Results*).

2. ENTITIES

World knowledge is included in decomposing the scene into actors and items: the entities. We know which type of entities to expect. To that end, we apply dedicated person, car and bike detectors [2]. To detect all other objects that move, we apply a standard moving object detector [3]. For both type of detectors, examples are provided in Figure 2.

We know about the size of entities: not just a few pixels and not half of the screen. We also have an understanding of their location: e.g. not in the sky.

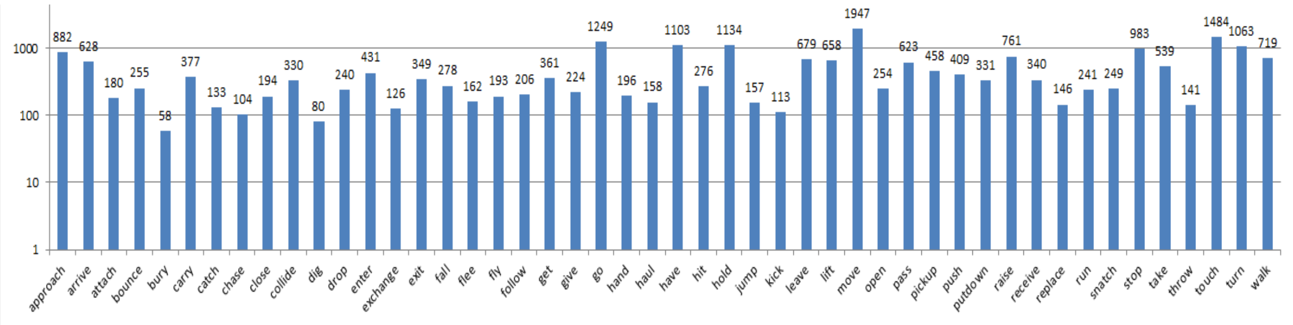

Figure 1. The 48 human behaviors in this paper and their (logarithmic) prevalence in the test set.

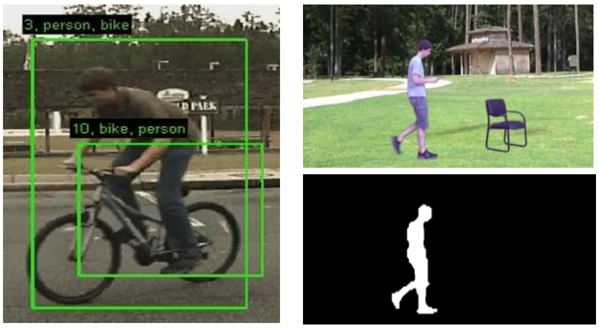

Figure 2. Dedicated detectors (left) and moving object detection (right).

We also know that typical trajectories are mostly horizontal and in a range of normal velocities of a few pixels per frame. Non-moving but shaky objects are usually false. We exploit such prior knowledge in order to merge object detections and to reduce false detections [4]. Illustrations of a typical merge and removal based on such rules are given in Figure 3.

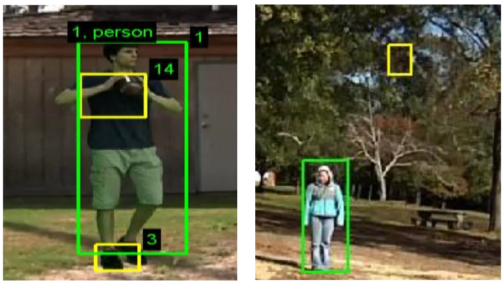

Figure 3. Merging (green) and removal (yellow).

3. EVENT PROPERTIES

World knowledge is also included in the event-related attributes of entities and between entities: the event properties. We know which properties define the behaviors [5]. Some behaviors involve one entity (e.g. walk) where others involve two or more entities (e.g. follow). There are behaviors that are defined by the involvement of some item (e.g. to give something), or the environment (e.g. dig).

Higher-level event properties are needed that describe the scene's entities and their kinematics, relations and interactions. Low-level features are not directly fit for creating the required event properties. This class of features is highly popular, because they are generally applicable and easily integrated into an application. Many reasonably discriminative and straightforward schemes with classifiers such as SVM have been proposed. STIP [6] is such an example: a highly informative feature of the (object) parts in the scene that are in motion. Yet such low-level features do not encode (at least not explicitly and arguably not even implicitly) essential event-related properties like interactions between people, items and their environments.

We make a distinction between single-entity event properties (e.g. type of entity; an entity moves horizontal; etc.), multiple-entity and relational properties (e.g. one entity approaches another entity; an entity holds an item; etc.) and global properties (e.g. there is more than one entity in the scene; etc.). These properties have been implemented. Many of the single-entity properties, like the kinematics of the entity are already very informative. In the example from Figure 4, somebody fell over a chair. The bounding box was portrait-oriented, moved to the right, and then went down and became landscape-oriented.

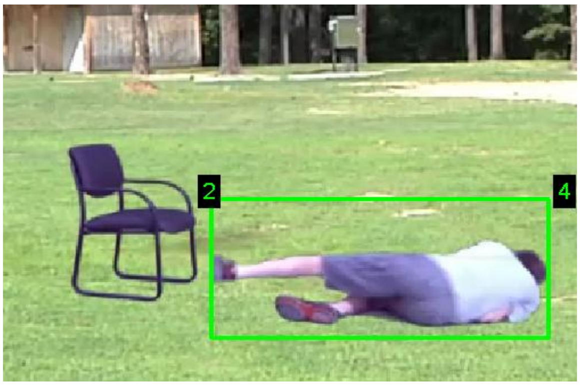

Figure 4. Kinematics of an entity.

In some cases, the straightforward implementation was not possible, for instance with an entity that holds an item. In many video fragments, the item that was in the hands of the person was not (clearly) visible. In such cases, we chose for a good trade-off between the information of the property and the likeliness of detecting it. In the case of the carried item, we implemented a derivative: the 'one-arm-out' pose. Given that we are interested mainly in events, like the exchange of an item, this is the best clue that some item is being carried and handed over to another person. The pose estimation of Ramanan [7] is projected onto a limited number of pose types that are relevant for the 48 behaviors. By adding skin detection and the local optical flow (see Figure 5), an indication is obtained where the object part is going.

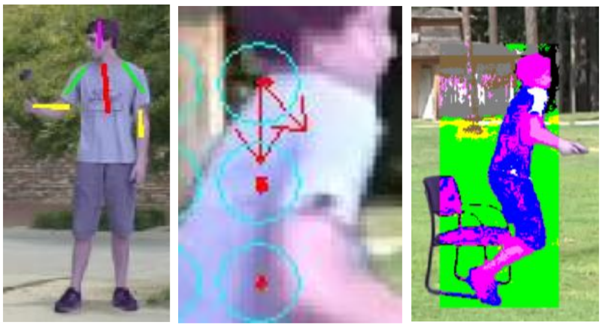

Figure 5. Pose (left), its local motion (middle) and skin detection (right).

Other properties involve relations between entities (e.g. approaching someone), or with the environment (e.g. burying something in the ground). The example in Figure 6 shows a woman that passes a man, who keeps at the same position. Such a spatiotemporal pattern is highly informative of somebody who passes another person.

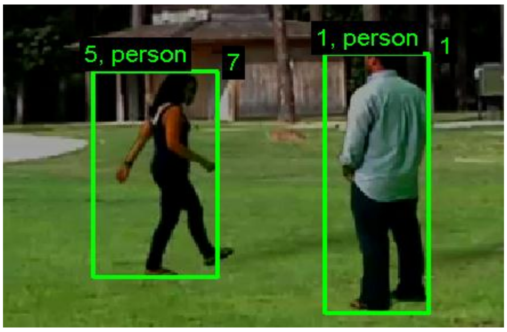

Figure 6. A multiple-entity event property that describes an interaction between persons.

In total we have implemented 57 single-entity properties, 13 multiple-entity properties and 8 environmental properties. For each entity (on average 5 entities are detected per video clip), at each video frame, we represent the beliefs about the 78 event properties. Note: we process this offline. On a standard desktop pc, the processing time per clip (average 15 seconds) is approximately 15 minutes. Most time is consumed by running the dedicated object detectors.

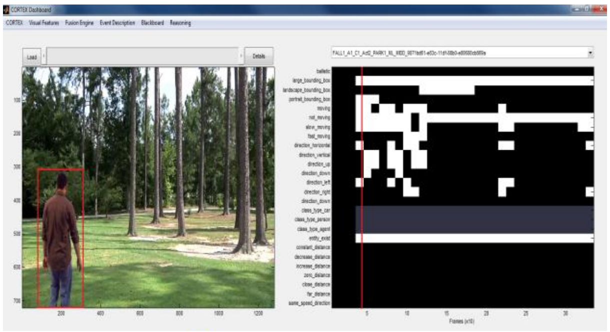

Figure 7. Event properties of a typical entity (left). The properties (right, vertical) are displayed on the time axis (right, horizontal).

4. RANDOM-FOREST TAG-PROPAGATOR

The goal of our system is to produce 48 beliefs about each of the human behaviors within the current test video clip. These beliefs are not mutually exclusive and need to be estimated from the event properties from all training videos.

To optimally exploit the available yet limited training set (3,480 videos), we compare a test video to all videos from this set. We base the beliefs about the 48 behaviors in the current video on the similarities to all of these previously seen videos. In this paragraph, we will establish a distance measure between two video clips to express their dissimilarity. First we need a representation for each clip. A difficulty is that each clip has a varying number of entities and varying length. The number of entities varies due to actual variations of the number of actors and items and due to erroneous variations i.e. missed and false detections. We have chosen a bag-of-features [8] representation that is independent of the number of entities and clip length. The advantage is that we are able to deal with all clips in exactly the same way and that the method has proven to be discriminative and robust to clutter (e.g. [8]). The disadvantage is that we do not explicitly associate entities in order to compare clips and therefore we loose selectivity.

For the representation of a clip, we consider the event properties. We know that not all properties are relevant for each type of behavior. We want to create a representation that makes some properties more important than others. This boils down to feature selection, which is guided by some form of labelling. To obtain this labelling, we have clustered (k-means, k=30) the human annotations for all 3,480 training clips (a vector of length 48 indicating presence or absence for each verb). With the resulting 3,480 labels we have created a random forest [9] of 200 trees on the event properties. The trees resemble good cuts on values of a subset of event properties that are selective of the guiding cluster labels. Each tree is generated from top-down, where each time a node is created to make a binary decision on a property's value to go to the next level. The creation of the ensemble of 200 trees requires a good balance of diversity of trees and predictive power of each tree [9]. This balance depends on a good choice of the M-parameter, which determines the randomness in the creation of new nodes. M has been optimized using the train set.

The trees are used to calculate a histogram for each clip. For a clip, all 78 event properties for each entity and for each timestep are passed through all trees. We count for each leaf (the end node) how often it was reached [10]. The histogram now resembles the mass over all leafs from all trees. The histogram is normalized to one to obtain a pdf that is independent of the number of entities and timesteps. Because each entity is fed through the trees and ends up in a single representation, the resulting representation is highly similar to the coding of natural images by a bag-of-features model [8]. The trees are independent and because they are likely to be redundant, this creates a robust representation of subsets of relevant event properties.

We generate a representation using the single random forest for all clips in our training and test set. Now that we have a representation for all clips, we are able to compare video clips. We want to compare the current test clip to all training clips to make optimal use of the learning set.

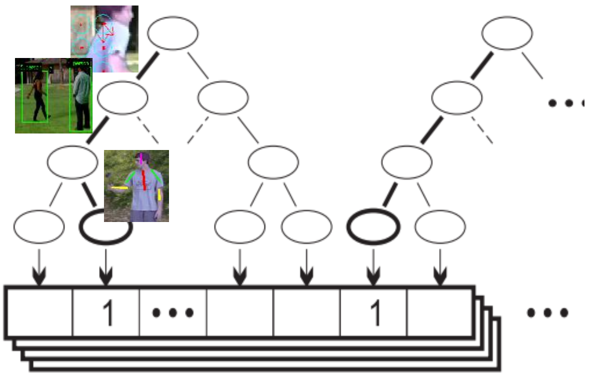

Figure 8. Random forest representation based on the event properties (figure adapted from [10]).

The final source of knowledge that we include in our system is the coincidence of the behaviors. Some of them are highly correlated. For instance, we know that if somebody follows another person, he is usually walking. Likewise, if somebody gives an item to another person, we will also observe that the other person receives the item. Clearly, correlations between the observed behaviors are significant.

For the current test clip, we have the following inputs available: its similarity distance to all training clips and the tags of the present behaviors for each training clip. Given the dissimilarities and tags, we learn a tag-propagator model [11]. This model is appropriate for making predictions about multiple labels (in our case 48 verbs) which are not mutually exclusive (and for some verbs even strongly correlating).

The essence of our approach is that we aim to model which clips are informative of particular behaviors – and how similar a new clip should be to count as evidence for a particular behavior. The driving element during training are the annotations from the humans (i.e. tags). The tags of a test clip are predicted using a weighted nearest-neighbor model to exploit the labeled training clips [11]. Neighbor weights are based on neighbor distance (i.e. the dissimilarity explained earlier). The model allows the integration of metric learning by directly maximizing the log-likelihood of the tag predictions in the training set. In the results, we refer to our recognizer as Random-Forest Tag-Propagator (RF-TP).

5. RECOGNITION RESULTS

The performance of our system is measured by the F1-measure to balance precision and recall: we want both to be good. The F1-measure is defined by: F1 = 2*TP/(2*TP+FP+FN), where T=true, F=false, P=positive and N=negative. We evaluate against the entire test set and compare to human annotations. For each clip and each of the 48 behaviors we have a present/absent was provided by DARPA. The human annotations appears to be very noisy. To measure the stability of human annotations, we compare humans to other humans as well. We aim to achieve a performance that is similar to humans. The procedures will be explained shortly.

To get insight in its generalization power, we split the evaluation for previously seen clips, unseen clips (yet the variations of behavior are similar to the previously seen clips) and unseen variations of behaviors. These cases are increasingly hard and we want to establish where the system degrades.

The first part of the evaluation is to establish whether the RF-TP performs better than straightforward alternatives. We have compared RF-TP against a rule-based system (RBS), a conditional random field with hidden units (HUCRF) and a recurrent temporal restricted Boltzmann machine (RTRBM). The reason for considering the RBS is to have an expert system with manually created rules as opposed to a statistical classifier. The reason for comparing with a HUCRF and RTRBM is to include classifiers from the set of temporal, graphical models. Where the HUCRF is a discriminative model, the RTRBM is generative. Together these four methods span a variety of the types of classifiers. Furthermore we compare to two baselines. The first baseline is the human performance, which we have derived from the annotations and the variation within annotators. The second baseline is a lower-bound baseline. This is the performance that we get when we produce the same fixed response for all clips by reporting the average number of occurrences for all verbs. On average, this is the best fixed response and it serves as a lower bound.

In Table 1, we summarize the recognition performances for the 2 baselines and the 4 classifiers. The reported F1-measure is first computed per verb and then averaged. In this way, we are less sensitive to the prevalence of verbs. HUCRF and RTRBM are similar to the lower baseline and RBS is just above. RF-TP performs significantly better than both the other classifiers. The RF-TP approaches human performance for the entire test set.

Human	*Base-line*	RBS	HU-CRF	RT-RBM	RF-TP
0.57	*0.40*	0.44	0.40	0.40	**0.56**

Table 1. F1-measures of two baselines and four classifiers across entire test set.

The second part of the evaluation is dedicated in establishing how well we are able to recognize

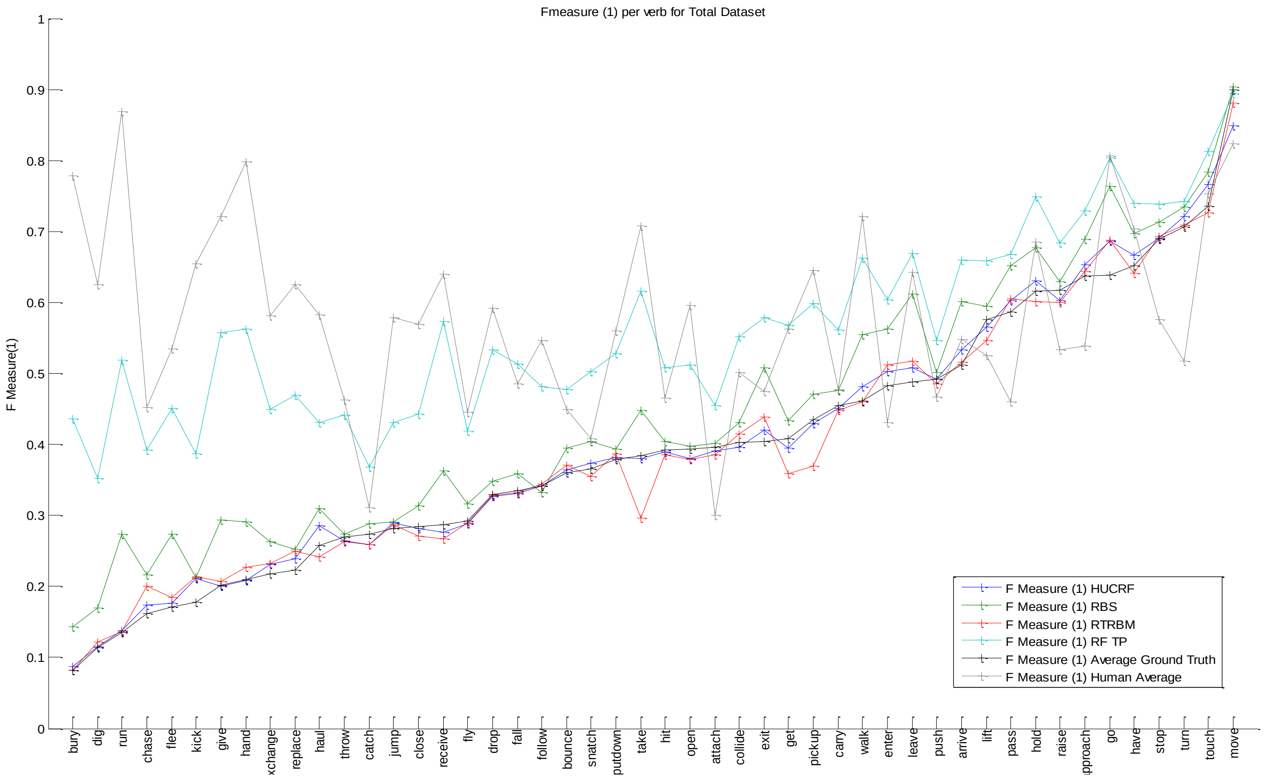

Figure 9. F1-measures for all verbs. The lower baseline (black, bottom) is the ‘average groundtruth’. The upper baseline (gray, top) is the human performance (‘human average’). The color lines represent the classifiers, from which the RF-TP is the only one that exceeds the lower baseline and approached the human performance.

each verb individually. Again we consider the F1-measure, yet now specified per verb. For each verb, the results are shown in Figure 9. The figure is organized as follows. From left to right, the verbs are ordered by increasing prevalence. So, for the verbs on the left, we have far less positive samples to learn from, in some cases only tens of clips (see also the dataset statistics in Figure 1). We point out that the F1-measure is emphasizing errors on the rare verbs, as it is dominated by the amount of true positives. This effect is clear from the figure: the lines are generally lower at the left part than at the right. At the right are verbs like move and go, which are present in respectively 50% and 75% of the videos. Clearly, the classifiers have optimized to perform well on common verbs: all lines towards the right are similar.

The black line on top represents the human performance. This is the line that defines the aimed performance for our RF-TP. The light blue line that is a little below the top line shows the performance of the RF-TP. Indeed, we are approaching the human performance when we consider the entire test set. The verbs on the left are harder and for this subset humans really do better. The black line at the bottom of Figure 9 is the baseline performance. The RTRBM and HUCRF are represented by the colored lines that are just above the baseline. These classifiers fail to discriminate between the 48 verbs. The RBS performs a slightly above these temporal classifiers, yet lacks good selectivity.

The third part of the evaluation is about the ability to generalize from the learned examples to completely new variations of the 48 behaviors. We have distinguished three subsets from the entire set. In increasing order of difficulty: clips we have seen before (these were also contained in the train set), unseen clips yet similar variations of the 48 behaviors, and totally unseen variations of the 48 behaviors. These increasingly hard cases are listed below in Table 2 from top to bottom. The RF-TP does not generalize well to completely unseen variations of behavior. The F1-measure drops to just a little above baseline for these cases. From the high performance for seen clips (compared to humans) we conclude that the the RF-TP is clearly overtraining.

	Human	*Baseline*	RF-TP
Seen clips	*0.57*	*0.39*	**0.65**
Unseen variations	-	*0.43*	**0.50**
Unseen behaviors	-	*0.40*	**0.43**

Table 2. F1-measures of two baselines and the RF-TP for various subsets of the test set.

We aim to discover the reasons for the drop in performance from seen clips to unseen behaviors.

In Figure 10, we have visualized the TP, FP, TN and FN per verb, for seen clips (top) and unseen variations (bottom). Horizontally we have ordered the verbs by prevalence (sum of TP and FN). The RF-TP misses (red) a serious amount of detections in the new cases. Generally, for all verbs, there are about 50% more false positives (orange). This happens especially for a particular set of verbs, given the few orange strikes in the figure. Examples are: leave, pickup, run, hold, etc. We hypothesize that this is due to their ambiguous nature: there are other verbs that are highly similar (e.g. leave/go, pickup/lift, run/walk, hold/carry). We will try to obtain better selectivity between the ambiguous verbs, and better generalization in general. For those verbs that are also indistinctive for humans, we will explore a scoring scheme that does not penalize them if they are confused.

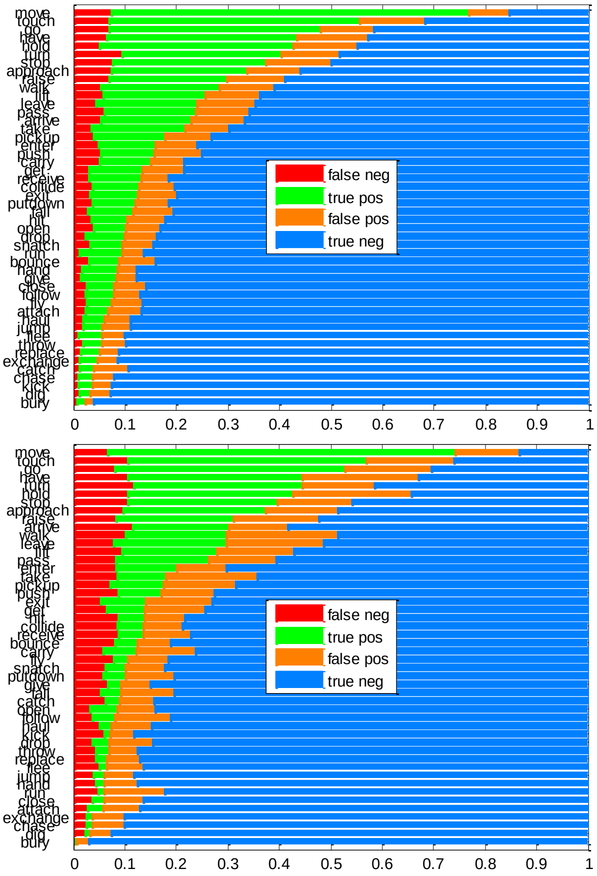

Figure 10. Positives and negatives for all verbs for seen clips (top) and unseen behaviors (bottom).

6. SUMMARY

We have proposed a framework to recognize a set of 48 complex behaviors. The behaviors in this paper include relations between persons, involvement of items, interactions with the environment. That makes this challenge go beyond the typical action recognition approaches where an individual is displaying a limited number of behaviors. Compared to the huge variation in the 48 behaviors, the training set of 3,480 clips is relatively small. To bridge the gap between the low-level features and the recognition, we have considered event properties. They serve as an intermediate level of understanding the set of events that define the behaviors. Our promising recognition results demonstrate that our set of event properties captures discriminative properties of the 48 human behaviors.

The event properties are represented by a bag-of-features model [8] where the codebook is generated by a random-forest [10]. The recognizer consists of a dissimilarity-based classifier. Guided by the human annotations, we retrieve the subsets of clips that are informative about particular behaviors by a tag-propagation model [11]. This RF-TP model proves to approach human performance on the entire test set. However, for the subset of completely unseen variations of behavior, the RF-TP does not yet generalize well.

7. DISCUSSION

Improvements on the RF-TP are expected in both representation and the classifier. For representation, we will optimize the generation of the trees in the random forest. Currently we are investigating the labelling that guides the tree formation process. Parameters that we are exploring are the number of leafs and the regularization in the nodes. The tag-propagation recognizer is exploiting the dissimilarities. We are exploring whether other dissimilarity measures perform better, including metric learning on the random-forest representations.

In parallel, we are extending the set of 78 event properties. We are zooming in on body parts when two persons are close to each other. We are exploring how to incorporate a generic item detector like [12] to detect whether a person is interacting with an item (e.g. give) or with something from the environment (e.g. open).

ACKNOWLEDGEMENTS

This work is supported by DARPA (Mind's Eye program). The content of the information does not necessarily reflect the position or the policy of the US Government, and no official endorsement should be inferred.

The authors acknowledge the CORTEX scientists and engineers for their significant contributions to the overall system: P. Hanckmann, J-W Marck, L. de Penning, J-M ten Hove, S. Landsmeer, C. van Leeuwen, W. Ledegang and R. Wijn.

8. REFERENCES

1. Fei-Fei, L., Fergus, R., Perona, P., "Learning generative visual models from few training examples: an incremental Bayesian approach

tested on 101 object categories", CVPR, 2004.

2. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D., "Object detection with discriminatively trained part based models", IEEE Transactions on Pattern Analysis and Machine Intelligence 32(9), 2010.
3. Stauffer, C., Grimson, W., "Adaptive background mixture models for real-time tracking", CVPR, 1999.
4. Ditzel, M., Kester, L., van den Broek, S., "System Design for Distributed Adaptive Observation Systems," International Conference on Information Fusion, 2011.
5. Bouma, H. Hanckmann, P., Marck, J.-W., Penning, L. de, Hollander, R. den, Hove, J.-M. ten, Broek, S.P. van den, Schutte K., Burghouts, G.J., "Automatic Human Action Recognition in a Scene from Visual Inputs", SPIE (to appear), 2012.
6. Laptev, I., Marszałek, M., Schmid, C. and Rozenfeld, B., "Learning realistic human actions from movies", CVPR, 2008.
7. Ramanan, D., "Learning to parse images of articulated bodies", NIPS, 2006.
8. Sivic, J., Zisserman, A., "Video Google: A text retrieval approach to object matching in videos", ICCV, 2003.
9. Breiman, L., "Random forests", Machine Learning, 45(1), 2001.
10. Moosmann, F., Triggs, B., Jurie, F., "Randomized Clustering Forests for Building Fast and Discriminative Visual Vocabularies", NIPS, 2006.
11. Moosmann, F., Triggs, B., Jurie, F., "Randomized Clustering Forests for Building Fast and Discriminative Visual Vocabularies", NIPS, 2006.
12. Alexe, B., Deselaers, T., Ferrari, V., "What is an object?", CVPR, 2010.