# A search engine for retrieval and inspection of events with 48 human actions in realistic videos

G.J. Burghouts, L. de Penning, M. Kruithof, P. Hanckmann, J-M ten Hove, S. Landsmeer, S.P. van den Broek, R. den Hollander, C. van Leeuwen, S. Korzec, H. Bouma, K. Schutte

*TNO, The Hague, The Netherlands*
gertjan.burghouts@tno.nl

Keywords: Human behavior understanding, search engine, video retrieval, action recognition, textual description, meta data, indexing, 48 human actions.

Abstract: The contribution of this paper is a search engine that recognizes and describes 48 human actions in realistic videos. The core algorithms have been published recently, from the early visual processing (Bouma, 2012), discriminative recognition (Burghouts, 2012) and textual description (Hankmann, 2012) of 48 human actions. We summarize the key algorithms and specify their performance. The novelty of this paper is that we integrate these algorithms into a search engine. In this paper, we add an algorithm that finds the relevant spatio-temporal regions in the video, which is the input for the early visual processing. As a result, meta-data is produced by the recognition and description algorithms. The meta-data is filtered by a novel algorithm that selects only the most informative parts of the video. We demonstrate the power of our search engine by retrieving relevant parts of the video based on three different queries. The search results indicate where specific events occurred, and which actors and objects were involved. We show that events can be successfully retrieved and inspected by usage of the proposed search engine.

## 1 INTRODUCTION

Recognizing and inspecting human activities is essential to security (robbery, vandalism), public safety (aggression, riots), health care (incidents with elderly people), commerce (shopping behavior) and also to users of the internet (searching YouTube videos). In this paper, we provide a solution for detecting and retrieving 48 human behaviors in realistic videos, like CCTV feeds or archives. We adopt algorithms from the state-of-the-art to perform the video processing and recognition of actions and textual description of events. Our goal is to demonstrate the power of a search engine that combines action recognition algorithm with an algorithm that described the recognized actions. We will show that the proposed search engine enables the user to find relevant events in realistic videos. The demonstrated examples are the retrieval of a rare event (a person who is digging), an event that involves the meeting of multiple people (persons that approach each other), and a complex event where two persons exchange an item.

The key algorithms are summarized in Sections 2 to 7, from early visual processing to producing the meta-data that the search engine uses to retrieve relevant events. For extensive discussion of state-of-the-art and related work, we refer to our previous papers on which the current paper is based, please see the references. The search engine is demonstrated in Section 9. In Section 10 we outline our final conclusions.

## 2 48 HUMAN ACTIONS

Our search engine has been trained on the visint.org dataset, which includes 4,774 videos of a wide range of clips that involve 48 human actions, in various forms including humans inside a car or on a motor bike or bicycle, where multiple human actions may happen at the same time, or just before or after each other. The actions vary from single person (e.g. walk) to two or more persons (e.g. follow). Some of these actions are defined by the involvement of some object (e.g. give), or an interaction with the

G.J. Burghouts, L. de Penning, J-M ten Hove, S. Landsmeer, S.P. van den Broek, R. den Hollander, P. Hanckmann, M. Kruithof, C. van Leeuwen, S. Korzec, H. Bouma, K. Schutte,
"A search engine for retrieval and inspection of events with 48 human actions in realistic videos",
Int. Conf. Pattern Recognition Applications and Methods ICPRAM, (2013).

environment (e.g. leave). The most complex actions involve two persons and an object (e.g. exchange, throw-catch combination).

# 3 EARLY VISUAL PROCESSING

Since the actions involve humans, and sometimes also vehicles, we use dedicated person, car and bike detectors (Felzenszwalb, 2010) and a generic moving object detector (Stauffer, 1999, Withagen, 2004). These methods deliver the bounding boxes, which are tracked by applying a local search for the best frame to frame match. World knowledge is included to improve the tracking. We know that typical trajectories are mostly horizontal and have typical velocities of a few pixels per frame. Non-moving but shaky objects are usually false. Such prior knowledge is included in order to merge object detections and to reduce false detections. The detected objects in the scene are referred to as 'entities'.

# 4 SPATIAL AND TEMPORAL EXTENT OF HUMAN ACTIONS

The next step is to find the parts in the video where particular events take place. We do so by segmenting the video into spatio-temporally confined regions. We refer to the initiative-taking entity as an 'agent'. The goal is to find the relevant agents and their relations with other entities and items.

Based on the detected entities and their bounding box and classification type (i.e., person, car, bike or other) our search engine determines the spatial and temporal extent of possible human actions. The engine calculates for each agent (an object classified as person, car or bike) possible spatio-temporal relations with other objects.

To determine whether these regions indeed contain informative human actions, each agent and its relations are passed on to the next level where features are computed, actions are recognized, and unlikely hypotheses are filtered..

# 5 ACTION FEATURES

In our search engine, we consider two complementary types of human action features, EP features which are based on bounding-boxes, and STIP features which are local features in space-time.

## 5.1 Localized motion features

The STIP features (Laptev, 2005) are regionally computed at spatio-temporal interest points, i.e. a 3D Harris detector that is an extension of the well-known 2D corner detector. The features comprise of histograms of gradients (HOG) and optical flow (HOF). Together these two feature types capture qualities about local shape and motion. The STIP features are computed with Laptev's implementation (Laptev, 2005), version 1.1, with default parameters. Our STIP based feature vector are the 162 STIP HOG-HOF features.

## 5.2 Event properties

The rationale is to design features that capture the semantically meaningful properties of the person and his/her actions, including kinematics, trajectory, interactions with other persons and items. These Event Properties (EP) features are a set of event-related attributes of entities, interactions and involved items. A distinction is made between single-entity EP features (e.g. type of entity; an entity moves horizontal; a person moves his arm, etc.), multiple-entity and relational properties (e.g. one entity approaches another entity; etc.) and global properties (e.g. there is more than one entity in the scene; etc.). In some cases, direct implementation is not possible, for instance with a person that holds an item, as the item that is carried by the person is not detectable. Instead, 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 the 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 handed over to another person. Pose estimation (Ramanan, 2006) is projected onto a set of 7 pose types that are relevant for the 48 behaviors. In total, 86 EP features are collected, of which 65 are single-entity, 13 are multi-entity, and 8 are global properties. An EP based feature vector used in this work lists the changes of these 86 EP features per entity.

# 6 DETECTION OF ACTIONS

Our search engine consists of a recognizer that detects 48 human actions in videos, and a descriptor

that provides textual descriptions. Our recognizer consists of 48 detectors, one for each human action.

We create action detectors from a pipeline of local spatio-temporal STIP features (Laptev, 2005), a random forest to quantize the features into action histograms (Moosmann, 2006), and a SVM classifier with a $\chi^2$ kernel (Zhang, 2007) serving as a detector for each action. For the random forest we use Breiman and Cutler's implementation (Breiman, 2001), with the M-parameter equal to the total number of features values. For the SVM we use the libSVM implementation (Chang, 2001), where the $\chi^2$ kernel is normalized by the mean distance across the full training set (Zhang, 2007), with the SVM's slack parameter default C=1. The weight of the positive class is set to (#pos+#neg)/#pos and the weight of the negative class to (#pos+#neg)/#neg, where #pos is the size of the positive class and #neg of the negative class (van de Sande, 2010).
The novelties with respect to the above pipeline are: (1) We have improved the selection of negative examples during training (Burghouts, 2012). The rationale is to select negatives that are semantically similar to the positive class. This gives an average improvement of approx. 20%. (2) We have improved the detection of each action, by fusion of all actions in a second stage classification. For each action, we create a second stage SVN classifier that takes the first stage classifiers' outputs, i.e. the posterior probability of each action detector, as a new feature vector (Burghouts, 2012). The improvement is approx. 40%. The combination of both improvements yields an overall improvement is 50% for the detection of the 48 human actions.
The recognizer's performance is measured by the Matthews Correlation Coefficient, MCC = ( TP·TN - FP·FN ) / sqrt( (TP+FP) · (TP+FN) · (TN+FP) · (TN+FN)), where T=true, F=false, P=positive and N=negative. This performance measure is independent of the sizes of the positive and negative classes. This is important for our evaluation purpose, as there are +1,000 positive samples for "move", to 61 samples for "bury". . The actions that go well (MCC > 0.2) are: Dig, Hold, Throw, Receive, Carry, Bounce, Raise, Replace, Exchange, Bury, Lift, Hand, Open, Haul. Fair performance $0.1 \leq MCC \leq 0.2$ is achieved for: Touch, Give, Kick, Take, Pickup, Fly, Drop, Snatch. Actions that do not go well (MCC < 0.1) are: Hit, Catch, Putdown, Push, Attach, Close. The average MCC = 0.23.

# 7 DESCRIPTION

Based on the actions classified by the action recognizer a textual description of the scene is generated. The description is generated by a Rule Based System (RBS). The RBS (Hanckmann, 2012) encodes world knowledge about the actions and encodes these as rules. There are 73 rules describing 48 actions. The rules specify a set of conditions. The conditions are based on the properties and relations as generated by the Event Properties (see 5.2).

The RBS connects the action with the entity or entities involved in the action. It determines which actor is the sentence subject and, if present, which object or actors are involved as direct or indirect objects. Subsequently, the description sentence is constructed using the action as an action combined with the subjects and objects, and a number of templates. A sentence is considered to at least contain a subject and an action.

## 7.1 RBS Algorithm

Based on the rules, a multi hypotheses tree is constructed. Each hypotheses is a combination of possible entities and/or object connected with the action. The hypothesis score is higher when more conditions are met.

There are a three condition types: entity/object properties (event properties that are expected to be valid for an entity/object in combination with an action) entity/object relations (event properties that are expected to be valid describing the relation two entities/objects have), temporal ordering (temporal properties of the previous two condition types, e.g. the order of actions in time).

The description describes the actions with the highest probabilities (maximum of seven actions with a minimum probability of 0.7). From these actions, the hypothesis with the highest score is selected and used in the sentence construction.

## 7.2 Description Performance

The description generator is evaluated on 241 short videos (visint.org) with ground truth. The ground truth consist of 10 sentences per video, written by 10 different people. Per video the number of different annotated actions is approximately 5.
For each ground truth (GT) sentence we extract, using The Stanford Parser (see last reference), the action, subject, and object(s) and compare these with

the system response (SR) of the RBS.
We calculate two scores: a union and a percentage score. The clip's union score is the best match for all sentence pairs (the percentage of clips with at least one agreement between GT and SR); its percentage score is the mean match corrected for the minimum number of the amount of ground truth sentences and the amount of generated sentences (the agreement between the sets of GT and SR).

Table 1. Scores of the textual description of actions.

| Description Score | Overall | Action | Subject | Objects |
|---|---|---|---|---|
| union | 68.3% | 92.3% | 62.0% | 67.8% |
| percentage | 40.4% | 59.3% | 30.5% | 31.5% |

# 8 SELECTION CRITERIA

In our pipeline, the spatio-temporal regions in the video are defined at the front-end. In this section, we call these regions 'actions'. Often, the actions are not informative: e.g. there are many 'walk' instances that are not relevant to the end-user. We need a filter on the meta-data that has been produced by the recognition and description algorithms.
To determine which actions should be selected, we present the following model. Each detected entity is assigned a confidence $P(E)$, an estimate for the chance that entity is indeed an entity. Each action consists of an agent, an entity which is the subject of the action and zero or more entities and multiple detected verbs. Each verb also has a confidence $P(v|E)$, which is an estimate for the chance that that verb is detected given the fact that the entity is detected. Furthermore each verb has a relevance $R(v)$. The relevance scales between 1 and 0 with 1 being false negatives are much more costly than false positives and 0 being false positives are much more costly than false positives.
For each action the total entity confidence, $P_t(E)$, is calculated from the confidences of each entity in the action, $P_t(E)$, as:

$$P_t(E) = \prod_i P(E_i).$$

The confidence of each verb $P(v)$ in the action is now calculated from the total entity confidence of the action and the verb confidence

$$P(v) = P(v|E) \cdot P_t(E).$$

The goodness of a verb in an action is defined as follows:

$$G(v) = R(v) \cdot P(v).$$

In a certain time window $T$ we take for each verb in a set of overlapping actions $a(T)$, for a single agent $A$ the maximum goodness:

$$G_A(v,T) = \max\left(G_{a(T)}(v)\right)$$

and report the $N$ verbs with the highest goodness larger than $X$ for all actions in the time window

$$V(N,X,T) = \max_N(G_A(v,T) \text{ if } G_A(v,T) > X)$$

The advantages of this simple model are that there are only two intuitive settings that determine the general output namely $N$ and $X$.

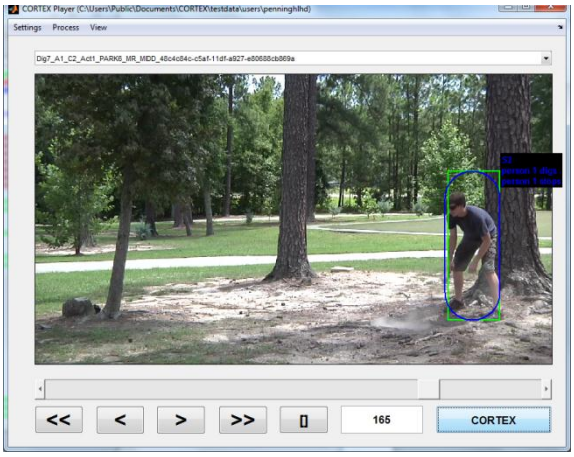
# 9 SEARCH ENGINE

A GUI enables interactive exploration of tracks, entities and actions in a video, that visualizes their temporal extent as segments in a segment viewer and their spatial extents as bounding boxes in a video player. When a segment is selected in the explorer (see Figure 1) the player displays the related frame (within the segment) and bounding boxes of the related tracks (yellow), entities (green), and actions (red). And when a bounding box is selected in the player, the explorer automatically jumps and zooms into the related segment

## 9.1 Person who is digging

Figure 1 shows a screenshot with a video of some person digging. As can be seen the search engine has detected the actions dig n and stop. The segment viewer shows the temporal extents of the detected person (i.e. agent 1), its actions (i.e. action 2) and relevant stories (i.e. stories 1, 2 and 3). It also shows the detected verbs (in light red) and related event properties (in light green). One can see that dig and stop are in the list, but also a number of other verbs. The reason why these verbs are not reported in the description of the stories (depicted in Figure 1) is because they had a low confidence or relevance.

Also depicted are the related event properties that support the detected verbs. For example one can see that in the beginning of the action the person was not moving and then started moving slowly in a vertical and downward direction, which is typical for a dig action.



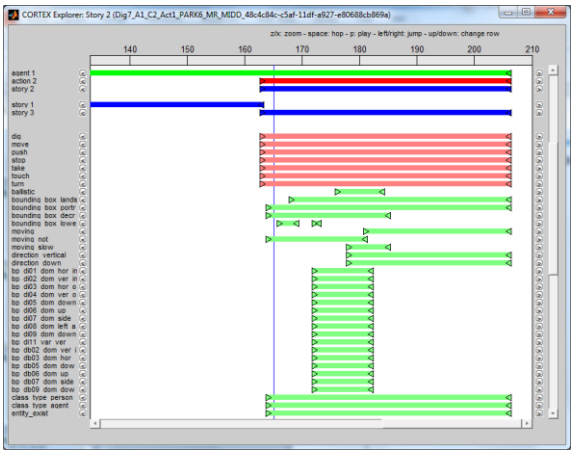
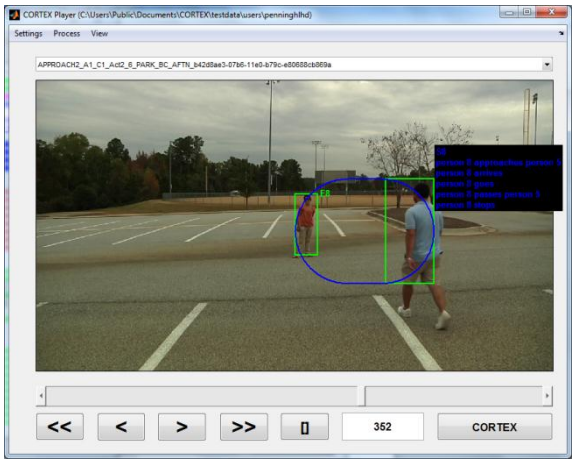




Figure 1: A person digging

## 9.2 People who meet each other

Figure 2 shows a screenshot with a video of a person approaching another person. As can be seen the search engine has detected the actions correctly (e.g. arrives, goes, approaches, stops). It also detected a "pass", which is incorrect. We can see the actions and related entities for the depicted scene. It also depicts an interaction (purple) segment that denotes the temporal extent of the two persons (i.e. agent 8 and entity 5) being involved in the interaction.

As can be seen in the segment viewer, the interaction has been reported as story in the system response for which several verbs have been recognized, but not all have been reported (due to filtering on relevance as described in section 8). An explanation for the reporting of "pass" is noise in the track detection which has been propagated to the event properties. As can be seen in the explorer, the property bounding_box_landscape is active during the second part of the interaction, although both persons are standing up. Because the bounding box of one person is sometimes detected as landscape, which easily overlaps the portrait bounding box of the other person, the action has been detected as a "pass".



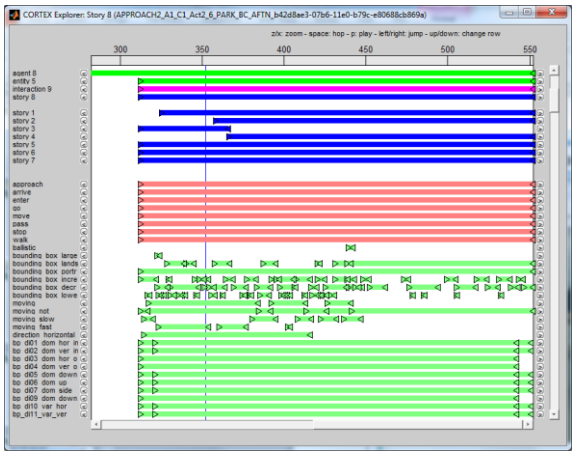




Figure 2: One person approaching another

## 9.3 Two persons who exchange an item

Figure 3 shows a screenshot with a video of two persons exchanging something. As can be seen the search engine has detected the actions correctly (e.g. has, gives and exchanges). But it also reported "fall" and "attach". The latter can be easily explained because of the small difference in moving parts for giving an item to someone or attaching an item to someone. But to explain the detection of "fall" we need to look at the segment viewer, where we can see the actions and related entities for the depicted scene. It also depicts an interaction (purple) segment that denotes the temporal extent of the two persons (i.e. agent 1 and entity 4) being involved in the exchange.

In the segment viewer one can see that more incorrect verbs have been recognized that all have to

do with vertical moving bounding boxes (e.g. bounce, drop, lift, raise). This typically the result of noise in the detected tracks and related STIP features (depicted in the explorer as many active STIP related event properties; "bp_diN_dom_..."). Due to the interplay of the recognizer and descriptor, only the verbs that have enough supporting evidence from the detected event properties are selected to be reported by the descriptor in the system response.

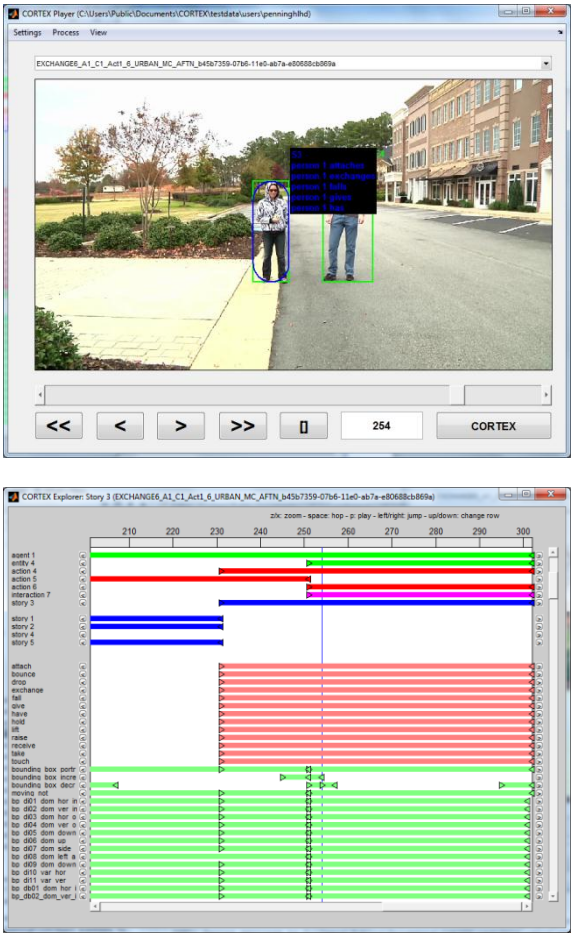



Figure 3: Two persons exchanging an item

## 10 CONCLUSIONS

We have showed that events related to 48 human behaviors can be successfully retrieved and inspected by usage of the proposed search engine. The search engine combines algorithms for early visual processing, spatio-temporal segmentation of the initiative-taking person and other related persons and items, extraction of data-driven and semantic action features, action detection, and description of the detected actions including the subject and object. Several examples of retrieved events in realistic videos demonstrate the power of the combined algorithms. We have shown successful searches for a rare event, an event that involves a group of persons, and a detailed action of two persons that exchange something. The search engine enables both the retrieval and inspection of human action related events.

## 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.

## REFERENCES

Bouma, H., Hanckmann, P., Marck, J-W., de Penning, L., den Hollander, R., ten Hove, J-M., van den Broek, S.P., Schutte, K., Burghouts, G.J., 2012. Automatic human action recognition in a scene from visual inputs, *SPIE*.

Breiman, L., 2001, Random forests, *Machine Learning*.

Burghouts, G.J., Schutte, K., 2012, Correlations between 48 human actions improve their detection, *ICPR*.

Chang, C.-C., Lin, C.-J., 2001, LIBSVM: a library for support vector machines, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D., 2010, Object detection with discriminatively trained part based models, *PAMI*.

Hankmann, P., Schutte, K., Burghouts, G.J., 2012, Automated Textual Descriptions for a Wide Range of Video Events with 48 Human Actions, *ECCV*.

Laptev, I., 2005, On Space-Time Interest Points, *IJCV*.

Moosmann, F., Triggs, B., Jurie, F., 2006, Randomized Clustering Forests for Building Fast and Discriminative Visual Vocabularies, *NIPS*.

Ramanan, D., 2006, Learning to parse images of articulated bodies", *NIPS*.

van de Sande, K.E.A., Gevers, T., Snoek, C.G.M., 2010, Evaluating Color Descriptors for Object and Scene Recognition, *PAMI*.

Stauffer, C., Grimson, W., 1999, Adaptive background mixture models for real-time tracking, *CVPR*.

Withagen, P.J., Schutte, K., Groen, F.C.A., 2004, Probabilistic classification between foreground objects and background, *ICPR*

Zhang, J., Marszalek, M., Lazebnik, S., Schmid, C., 2007, Local features and kernels for classification of texture and object categories: A comprehensive study, *IJCV*.

Stanford Parser: http://nlp.stanford.edu/software/lexparser.shtml