Variations of a Hough-Voting Action Recognition System

Daniel Waltisberg, Angela Yao, Juergen Gall, and Luc Van Gool

Computer Vision Laboratory, ETH Zurich, Switzerland

Abstract. This paper presents two variations of a Hough-voting framework used for action recognition and shows classification results for low-resolution video and videos depicting human interactions. For low-resolution videos, where people performing actions are around 30 pixels, we adopt low-level features such as gradients and optical flow. For group actions with human-human interactions, we take the probabilistic action labels from the Hough-voting framework for single individuals and combine them into group actions using decision profiles and classifier combination.

Keywords: human action recognition, Hough-voting, video analysis, low-resolution video, group action recognition, activity recognition

1 Introduction

Recognizing human actions from video has received much attention in the computer vision community, though designing algorithms that can detect and classify actions from unconstrained videos and in realistic settings still remains a challenge. One difficulty is scene diversity, i.e. methods designed for sports analysis may not be well suited for surveillance. Furthermore, much of the work in action recognition has focused on single persons. In applications such as intelligent surveillance, where the goal is to detect unusual or dangerous events, however, the classification of group interactions becomes more critical as situations can only be understood by considering the relationship between persons.

We present here variations on a Hough-voting framework for action recognition, previously introduced in [8], as applied to two very different action recognition scenarios from the *ICPR 2010 Contest on Semantic Description of Human Activities*. In the first scenario, the Hough-voting framework is directly applied to classify actions on low-resolution videos, in which people performing actions are around 30 pixels high. In the second scenario, we classify group actions by combining the classification results of single individuals to strengthen the group action response.

The rest of the paper is organized as follows. In Section 2, we give a short summary of the Hough-voting framework described in [8]. In Section 3, we describe the combination of the classifier outputs of multiple people into group actions by using classifier combination rules and extending the model of decision profiles [6]. In Section 4, we show the classification results on low resolution videos and on group action recognition. Finally, Section 5 summarizes the main results.

2 Hough-voting framework

The Hough-voting framework in [8] takes a two-staged approach. In an initial localization stage, the person performing the action is tracked. Then, in a secondary classification stage, 3D feature patches from the track are used to cast votes for the action center in a spatio-temporal action Hough space. In [8], a tracking-by-detection approach was used, though any other tracking method can be used as well since the tracking stage is disjoint from the classification stage. For classifying the action, random trees are trained to learn the mapping between the patches and the corresponding votes in the action Hough space.

2.1 Training

We train a random forest, which we term a “Hough forest”, to learn the mapping between action tracks and a Hough space. Each tree is constructed from a set of patches $\{\mathcal{P}_i = (\mathcal{I}_i, c_i, \boldsymbol{d}_i)\}$, where

$\mathcal{P}_i$ is a 3D patch (e.g. of 16 × 16 × 5 pixels) randomly sampled from the track.
$\mathcal{I}_i$ are extracted features at a patch and can be multi-channeled to accommodate multiple features, i.e. $\mathcal{I}_i = \left(I_i^1, I_i^2, ..., I_i^F\right) \in \mathbb{R}^4$, where each I_i^f is feature channel f at patch i and F is the total number of feature channels.
c_i is the action label.
$\boldsymbol{d}_i$ is a 3D displacement vector from the patch center to the action track center.

From the set of patches, the tree is built from the root by selecting a binary test t, splitting the training patches according to the test results and iterating on the children nodes until either the maximum depth of the tree is reached or there are insufficient patches remaining at a node. Each leaf node stores p_c, the proportion of the patches per class label reaching that leaf, and $D_c = \{\boldsymbol{d}_i\}_{c_i=c}$, the patches' respective displacement vectors.

The binary tests compare two pixels at locations $\boldsymbol{p} \in \mathbb{R}^3$ and $\boldsymbol{q} \in \mathbb{R}^3$ in feature channel f with some offset τ, i.e.

$$t_{f,\boldsymbol{p},\boldsymbol{q},\tau}(\mathcal{I}) = \begin{cases} 0 & \text{if } I^f(\boldsymbol{p}) < I^f(\boldsymbol{q}) + \tau \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

First, a pool of binary tests with random values of f, $\boldsymbol{p}$, $\boldsymbol{q}$ and τ are generated; the test which splits the patches with minimal class or offset uncertainty between the split is chosen. By switching randomly between the two uncertainty measures, the leaves tend to have low variation in both class label and center displacement.

2.2 Classifying and Localizing Actions

During test time, we extract densely sampled patches from the tracks and pass them through the trees in the Hough forest. Each patch arriving at a leaf votes into the action subspace proportional to p_c and into the space-temporal subspace of each class c according to a 3D Gaussian Parzen window estimate of the center offset vectors D_c. Votes from all patches, passed through each of the trees, are integrated into a 4D Hough accumulator. As the track has already been localized in space, we can marginalize the votes into a 2D accumulator in class label and time, with the maxima indicating the class label and temporal center of the track. For a formal description of the voting, we refer the reader to [8].

3 Combining Classifiers for Group Action Recognition

In our setting of group action recognition, we distinguish between symmetric or asymmetric interactions. Symmetric interactions are those in which all individuals perform the same movements, such as shaking hands. Asymmetric interactions, on the other hand, are those in the which the individuals behave differently. For example, when one person pushes another, there is an offender and a victim. We assume for simplicity that victims of all asymmetric actions behave in a similar way and add one generic victim class.

For each individual participating in an action, we get a single-person classification, and then combine them into group classifications using combination rules such as product rule, sum rule, min rule and max rule to strengthen the overall group response. A theoretical framework of these combination rules is given in [5]. A convenient and compact representation of multiple classifier outputs is the decision profile matrix [6] as the combination rules can be applied directly to the matrix. In the following, we review the model of decision profiles and extend them to handle both symmetric actions and asymmetric actions.

3.1 Decision Profiles

We define $c+1$ single action labels, corresponding to c group interactions and an additional victim label v. For each person l in a group interaction of L people, we have a single action classifier D_l, giving for each time instance t

$$D_l(t) = [d_{l,1}, \ldots, d_{l,c}, d_{l,v}] \,, \tag{2}$$

where each d corresponds to the support for a single action class. To combine the single action classifier outputs into group actions, we formulate a decision profile, DP, in matrix notation:

$$DP(t) = \begin{bmatrix} D_1(t) \\ \cdots \\ D_l(t) \\ \cdots \\ D_L(t) \end{bmatrix} = \begin{bmatrix} d_{1,1} \cdots d_{1,c}, \; d_{1,v} \\ \cdots \\ d_{l,1} \cdots d_{l,c}, \; d_{l,v} \\ \cdots \\ d_{L,1} \cdots d_{L,c}, \; d_{L,v} \end{bmatrix} . \tag{3}$$

For the combination of the single actions, the product, sum, min and max rule are directly applied to each column of the decision profile [6].

3.2 Extension for Asymmetric Group Actions

In our case, as we have added a victim class, we extend the above DP by dividing it into a symmetric and asymmetric block:

$$DP(t) = [DP_{sym}(t) \;\mid\; DP_{asym}(t)], \tag{4}$$

with $DP_{sym}(t)$ as defined in Equation (3), but for single action labels belonging to symmetric group interactions only. To handle the asymmetric group interactions, we consider each combination of single actions which could form the interaction. Equation (5) describes the combination for a two-person scenario, but can be easily adapted for more people. Assuming m asymmetric group actions with classifier outputs $d_{l,1}, \ldots, d_{l,m}$ and one victim class v with classifier output $d_{l,v}$, the asymmetric decision profile would be a $2 \times 2 \cdot m$ dimensional matrix defined as follows:

$$DP_{asym}(t) = \left[\begin{array}{cc|cc|c|cc} d_{1,1} & d_{1,v} & d_{1,2} & d_{1,v} & \cdots & d_{1,m} & d_{1,v} \\ d_{2,v} & d_{2,1} & d_{2,v} & d_{2,2} & \cdots & d_{2,v} & d_{2,m} \end{array}\right]. \tag{5}$$

While Equation (5) is a redundant representation of the single action classifications, we choose this formulation as the same classifier combination rules can be directly applied to the each column of the decision profile.

4 Experiments

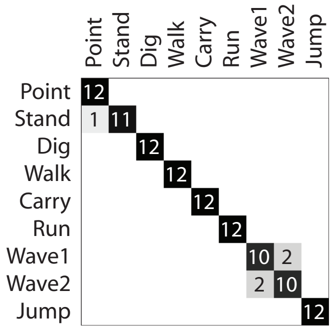
4.1 Action Recognition in Low-Resolution Video

We apply the Hough-voting framework described in Section 2 to classify the actions in the UT Tower Dataset [2]. For building the tracks, we used the provided foreground masks and fit 40 × 40 pixel bounding boxes around the foreground blobs. To handle the low resolution of the video, we chose low-level features robust at lower resolutions [1, 3], and chose greyscale intensity, absolute value of the gradients in x, y and time, and the absolute value of optical flow in x and y.

We achieve an overall classification performance of 95.4%. The confusion matrix is shown in Figure 1. There is some confusion between similar actions, such as *standing* and *pointing*, or *wave1* and *wave2*, but all other actions are classified correctly.

4.2 Group Action Recognition

We demonstrate our approach of group action recognition on the UT-Interaction dataset [7], consisting of six classes of two-person interactions shown in profile view: *shake (hands), hug, kick, point, punch* and *push.* We consider *shake* and *hug* as symmetric and the others as asymmetric interactions. For each class, there are two settings: *set 1* recorded from a parking lot with a stationary background and *set 2*, recorded on a lawn with some slight background movement and camera jitter.



	Point	Stand	Dig	Walk	Carry	Run	Wave1	Wave2	Jump
Point	12								
Stand	1	11							
Dig			12						
Walk				12					
Carry					12				
Run						12			
Wave1							10	2	
Wave2							2	10	
Jump									12

Fig. 1. Confusion matrix for classification on the UT-Tower Dataset.

Single Person Actions. We use the described Hough-voting method to classify the single actions, using the same features as mentioned in Section 4.1. The tracks were built with a Hough forest trained for people detection [4] and a particle filter was used to assemble detections across time.

For simplification, only one classifier was trained for both the left and the right person; during testing, the classifier was applied to both the original and flipped version of the tracks and determined based on the higher response of the classifier if the person in the track stands on the left or right. Classification results for the seven single action classes are shown in Table 1.

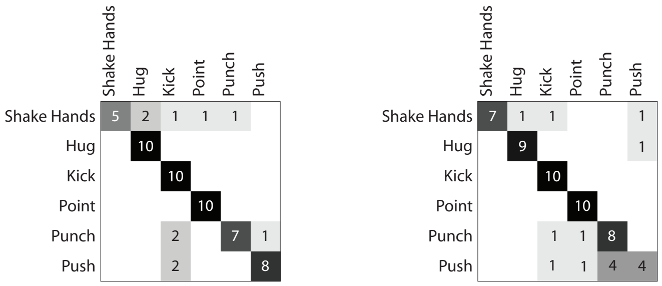
	Set 1		Set 2	
	Left Track	Right Track	Left Track	Right Track
Shake	0.7	0.3	0.3	0.2
Hug	0.9	1.0	0.9	0.9
Kick	1.0	1.0	1.0	1.0
Point	0.8	0.63	0.6	0.6
Push	0.33	0.72	0.8	0.8
Punch	0.66	0.86	0.6	0.2
Victim	0.77	0.73	0.9	0.8
Average	0.74	0.75	0.73	0.64

Table 1. Classification performance of single actions according to track

Group Interactions. For evaluation of the group interactions, we use a leave-one-out cross validation for each set individually. Performance of the different combination rules are compared in Table 2. Confusion matrices of the min-rule for *set 1* and *set 2* are shown in Figures 2*(a)* and *(b)* respectively. Average performance of the best group classifier compared to the best single person classifier was higher by 13% in *set 1* and 7% in *set 2*. The min rule performs well for both sets. The product and sum rule have similar performance in both sets, but are more affected by a weaker individual classifier as is the case in *set 2* for right individual.

	Set 1				Set 2			
	Min	Max	Product	Sum	Min	Max	Product	Sum
Shake	0.5	0.4	0.6	0.7	0.7	0.1	0.5	0.5
Hug	1.0	1.0	1.0	1.0	0.9	0.8	0.9	0.9
Kick	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Point	1.0	0.6	1.0	1.0	1.0	0.5	1.0	1.0
Push	0.7	0.2	0.7	0.7	0.8	0.1	0.8	0.8
Punch	0.8	0.1	0.9	0.9	0.4	0.0	0.4	0.4
Average	0.83	0.55	0.87	0.88	0.8	0.42	0.77	0.77

Table 2. Classification performance of group interactions for different fusion methods.



	Shake Hands	Hug	Kick	Point	Punch	Push
Shake Hands	5	2	1	1	1	
Hug		10				
Kick			10			
Point				10		
Punch			2		7	1
Push			2			8

(a) *set*1

	Shake Hands	Hug	Kick	Point	Punch	Push
Shake Hands	7	1	1			1
Hug		9				1
Kick			10			
Point				10		
Punch			1	1	8	
Push			1	1	4	4

(b) *set*2

Fig. 2. Confusion matrix for classification of group actions for *(a) set 1* and *(b) set 2* using the min rule for classifier fusion.

5 Discussion

The Hough-voting framework for action recognition, previously introduced in [8], was applied to two very different action recognition scenarios and showed flexibility and good results for both tasks. For classifying aerial video, we chose low-level features which were robust at low resolutions. For classifying group interactions, we presented a method for combining the classifiers of single-person actions. Overall performance was increased in comparison to single actions and the method can be easily adapted for scenarios with more than two people. A major advantage of this approach is that no additional training is needed for classifier combination.

Acknowledgements: This work has been supported by funding from the Swiss National Foundation NCCR project IM2 as well as the EC project IURO. Angela Yao was also supported by funding from NSERC Canada.

References

1. Chen, C.C., Aggarwal, J.K.: Recognizing human action from a far field of view. In: IEEE Workshop on Motion and Video Computing (WMVC) (2009)

2. Chen, C.C., Ryoo, M.S., Aggarwal, J.K.: UT-Tower Dataset: Aerial View Activity Classification Challenge. http://cvrc.ece.utexas.edu/SDHA2010/Aerial_View_Activity.html (2010)
3. Efros, A.A., Berg, A.C., Mori, G., Malik, J.: Recognizing action at a distance. In: ICCV (2003)
4. Gall, J., Lempitsky, V.: Class-specific hough forests for object detection. In: CVPR (2009)
5. Kittler, J., Society, I.C., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. In: IEEE Transactions on Pattern Analysis and Machine Intelligence. vol. 20, pp. 226–239 (1998)
6. Kuncheva, L.I., Bezdek, J.C., Duin, R.P.W.: Decision templates for multiple classifier fusion: an experimental comparison. In: Pattern Recognition. vol. 34, pp. 299–314 (2001)
7. Ryoo, M.S., Aggarwal, J.K.: UT-Interaction Dataset, ICPR contest on Semantic Description of Human Activities (SDHA). http://cvrc.ece.utexas.edu/SDHA2010/Human_Interaction.html (2010)
8. Yao, A., Gall, J., van Gool, L.: A hough transform-based voting framework for action recognition. In: CVPR (2010)