# Automatic Collection of Web Video Shots Corresponding to Specific Actions using Web Images

Do Hang Nga Keiji Yanai

Department of Informatics, The University of Electro-Communications, Tokyo
1-5-1 Chogugaoka, Chofu-shi Tokyo 182-8585 Japan
dohang@mm.cs.uec.ac.jp, yanai@cs.uec.ac.jp

## Abstract

*In this paper, we apply Web images to the problem of automatically extracting video shots corresponding to specific actions from Web videos. Our framework modifies the unsupervised method on automatic collecting of Web video shots corresponding to the given actions which we proposed last year [9]. For each action, following that work, we first exploit tag relevance to gather 200 most relevant videos of the given action and segment each video into several video shots. Shots are then converted into bags of spatio-temporal features and ranked by the VisualRank method. We refine the approach by introducing the use of Web action images into shot ranking step. We select images by applying Poselets [2] to detect human in the case of human actions. We test our framework on 28 human action categories whose precision values were 20% or below and 8 non-human action categories whose precision values were less than 15% in [9]. The results show that our model can improve the precision approximately 6% over 28 human action categories and 16% over 8 non-human action categories.*

## 1. Introduction

In the research community of action recognition, as classification rates on small and fairly controlled datasets like KTH [12] or Weizmann [1] has reached nearly perfect rates [8], there is a need for a large and unconstrained dataset. The largest and latest proposed action dataset is quite large with 51 action categories [6]. However, so far constructing action training database is known as a very time-consuming process, since video sequences corresponding to a specific action are usually recorded or collected from video sources like movies manually. On the other hand, more and more videos are being uploaded to the Web through video sharing web sites such as YouTube and DailyMotion. Even though when we search for relevant videos of a specific action the retrieved results generally contain noise, some of them may actually hold relevant scenes that correspond to the action. Here we consider a video shot as a set of consecutive frames which represent a scene. If video shots corresponding to any action can be obtained automatically from Web source, so that building action database will become easier than before.

The problem of automatically extracting video shots related to specific human actions from Web videos in the unsupervised manner was first proposed in our paper last year [9]. We aimed to collect most relevant video shots to given action keywords from a large number of tagged YouTube videos by an unsupervised ranking method. The top ranked shots are supposed to be shots corresponding to the actions. We tried as much as 100 human action categories including sport activities like "jog" or "swim butterfly" and activities of daily living such as "drink coffee" and "wash dishes". Our previous approach achieved 50% or over in precision over 24 categories. Note that the precision here is defined as the proportion of relevant shots over 100 top ranked shots (Precision@100). However, in case of some other actions, due to the extremely noisy tags which have been tagged subjectively by YouTube users, we did not succeed in selecting relevant videos, and our previous approach did not obtain as much relevant shots as desired.

Then, in this paper, we modify our previous method by taking Web still images corresponding to given actions into account, with an intuition that the shots which are more similar to Web action images may be more likely relevant shots so thus they should be ranked higher. In fact, recent works [11, 17, 13, 16] show that action recognition for still images is possible. Then, we collect images related to the given actions automatically via Web image search engines by only provided keywords, and evaluate the similarities between video shots and selected images by feature matching. That means our modification also does not require any supervision, so that the automaticity of the whole framework can

be preserved.

In our framework, with each action keyword, we first download 200 most action related videos from YouTube; segment each downloaded video into several shots and represent each shot as a bag of spatio-temporal features following the method described in [9]. At the same time, we download hundreds of images using Bing API and apply Poselets [2] to detect human in those images. Next, we extract SURF features [5] from video shots and human detected images, then calculate similarity between each shot and those images by counting matched local features. Note that we apply human-detection-based image selection on human actions only. In case of non-human actions, we simply use images directly retrieved by Bing API. Finally, we apply VisualRank ranking method [7], to rank video shots by taking account of spatio-temporal features of video shots and shots-images similarities so that we can obtain relevant shots to the given keywords as highly ranked shots.

We test our framework on 28 human action categories whose precision values are less than 20% in our previous work [9]. The results show that our modification enhances the performance on tested categories by approximately 6% in average. Especially, for 8 categories whose precision is below 10%, the performance is improved significantly from 5.7% to 21.6%. We also verify the efficiency of the proposed framework to non-human actions whose precision values is less than 15% in [9]. These results demonstrate that by introducing Web images into shot ranking, we can boost the precision from 2% to 16% in average. That means even in the case where tags are too noisy so that they can lead to the selection of mostly irrelevant videos, our proposed method still can extract from those videos quite a number of action related video shots.

## 2. Related Work

In this section, we refer to some related works on action recognition with uncontrolled video datasets.

Recently, works which are dealing with video categorization for YouTube videos [15, 14] and Kodak consumer video dataset [4] has been increasing. All of them employed supervised learning which requires training samples, and their objective is categorizing videos into one of the predefined categories. On the other hand, our work do not require any training samples and detecting video shots associated with the given keywords for a large number of Web videos.

As the most similar work to ours, Cinbis *et al*. [3]'s method learn action models automatically from Web images gathered via Web image search engines. While we use both Web videos and Web images as training source and spatio-temporal features as action descriptors, they use only Web images and static features. In addition, they concentrate on only human actions, while we aim to all kinds of action including non-human actions such as "airplane+flying" and "flower+blooming".

## 3. Proposed Framework

The framework presented in this paper is built on the framework of automatic construction of an action video shot database using Web videos [9], which aims to extract most relevant video shots to given keywords from a large number of tagged YouTube videos in an unsupervised manner. The introduction of Web images into video shot ranking process make it more possible to obtain relevant shots in the case that tag noisy causes the failure on collecting relevant videos.

### 3.1. Previous Framework

Our approach bases on our previous work [9], which first introduced the problem of constructing automatically a large scale action video shot database using Web videos in the unsupervised way. Given the action keywords such as "walking" or "surfing+wave", we proposed to extract from tagged YouTube videos video shots corresponding to those actions by following three steps: (1) tag-based video selection, (2) segmenting videos into shots and extracting features from all the shots, (3) visual feature-based video shot selection with tag-based scores taken into account.

In the first step, for each given action keyword, we rank 1000 YouTube videos which have been tagged with the keyword regarding the relevance of each video to the action by using tag co-occurrence statistics and select 200 most relevant videos to download. Note that a list of tags and video IDs corresponding to the given query word can be obtained via YouTube API, so it is not necessary to download all 1000 videos for tag-based video ranking and just 200 top ranked ones will be downloaded using their IDs. In the second step, each of downloaded video is divided into video shots and visual features are extracted from each of all the shots. As visual features, we use the spatio-temporal (ST) features proposed by Noguchi and Yanai [10]. In the third step, they rank video shots by applying graph-based ranking method, VisualRank [7], with a visual-feature-based similarity matrix and a bias damping vector which is calculated based on tag relevance scores of videos.

This framework works quite well in many action categories. However, in the case where the tags are too noisy, most of results of tag based video selection are irrelevant videos and shot selection step obtains just a very few relevant video shots.

### 3.2. Improvements

In this paper, we enhance our previous work [9] by introducing Web images into video shot selection so that even when noisy tags caused failures on video selection, our ap-

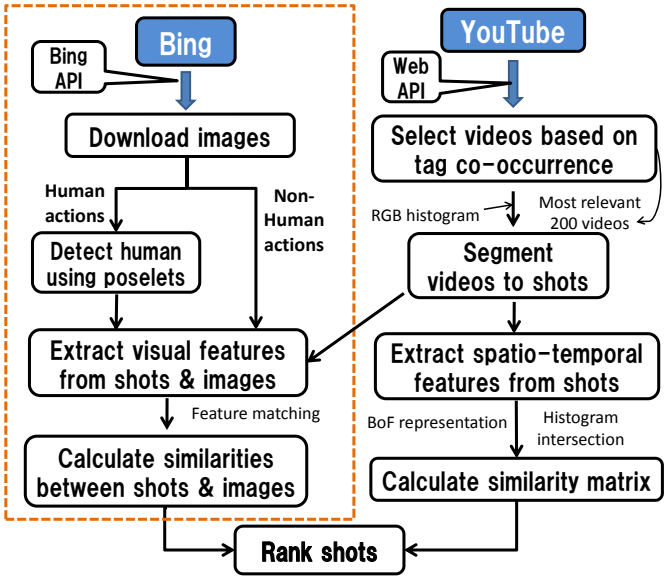


Figure 1. Overview of the proposed framework. The half left part with a dotted red box shows our modifications.

proach still can detect quite a number of video shots corresponding to the given actions.

The proposed method consists of four processing steps (Figure 1):

(1) tag-based video selection; video shot segmentation; spatio-temporal feature extraction from shots; bag of features representation; similarity matrix between shots calculation,

(2) for only human actions: Web images selection using Poselets-based human detection [2],

(3) visual feature extraction from shots and images; feature matching based similarity calculation, and

(4) spatio-temporal-feature-based video shot selection with shots-images similarities taken into account.

For each action, following [9], we first evaluate the relevance of each video in 1000 videos which have action keyword as one of their tags to the action using tag co-occurrence; download 200 most relevant videos; segment each video into shots based on its RGB histogram; extract spatio-temporal features from all the shots and represent them as bags of features; calculate similarity matrix for shot ranking using histogram intersection. The next steps are our modification to the previous work and represented as the left part of Figure 1 with a dotted red box. This is how we automatically select action related images and use selected images to improve relevant shots selection. We download hundreds of images from the Web via a Web image search API, Bing API, which is officially provided by Microsoft. We then apply Poselets proposed by [2] to detect human in searched images. The human detected images will be selected to go to the next step. The appropriate number of images to select will be discussed in the experiment section. In the next step, we extract visual features from selected shots and images and evaluate the similarities between shots and images based on local image feature matching. Here we use SURF [5] as visual feature. Finally, with the similarity matrix between shots obtained in the first step and a damping vector calculated by using the similarities between shots and images, we rank video shots applying VisualRank method [7].


Figure 2. The top six Web images after Poselets-based filtering.

## 4. Methods

In this section, we describe in detail Web action image selection using Poselets-based human detection; feature matching based shots-images similarities calculation and shot ranking with shots-images similarities taken into account.

### 4.1. Web action image selection

When one queries an action keyword on Web image search engine, thousands of images might be returned. However, in general, even the top returned images may be not relevant images of the queried action because of a wide variety of meaning of the keyword as well as the action itself, especially in the case of human action. On the other hand, we also want to preserve the automaticity of the original framework, since manual selection is not preferred here. We postulate two assumptions: (1) the set of retrieved images contains relevant images of the queried action and (2) human or body part should be seen in human action images.

It is reasonable to consider that images which contain human poses are likely related to that human action. Based on these assumptions, we select a collection of action images by applying Poselets-based human detection [2] on Web images. Poselets are demonstrated as effective body part detectors trained by 3D human pose annotations. We apply Poselets detector tools which are officially offered by the authors [1] on the set of retrieved Web images using default parameters. Figure 2 illustrates some examples of selected Web images using Poselets-based human detection.

The appropriate number of images to select then become a question. Let $N$ be the number of action images that will

[1] http://www.cs.berkeley.edu/%7Elbourdev/poselets/

be used to improve relevant shot detection, is it true that the larger $N$ is, the more relevant shots will be detected? We actually try several values for $N$: 10, 20, 30, and 50, and discuss more about this issue in the experiment section.

### 4.2. Shots-images similarity calculation

To evaluate the similarity between a video shot and given set of action images, we first extract SURF local features from all action images of selected set and each one frame per five consecutive frames of all the shots. For each shot, we count matching points between SURF local features extracted from each frame and each Web image by thresholding Euclidean distances between SURF feature vectors. The similarity $\mathrm{SI}(S_i)$ between a shot $S_i$ which has $M$ frame images ($F_j(j = 1..M)$) and an image set $\mathcal{I}$ which has $N$ images ($I_k(k = 1..N)$) is calculated by the following equations:

$$\mathrm{SI}(S_i) = \sum_{k=1}^{N} \max_{j=1:M} \mathrm{SI}(F_j|I_k), \quad (1)$$

$$\text{where } S_i(F|I_k) = \frac{2 * \mathrm{MatchPoint}(F_j, I_k)}{(\mathrm{Point}(F_j) + \mathrm{Point}(I_k))}, \quad (2)$$

$\mathrm{MatchPoint}(F_j, I_k)$, $\mathrm{Point}(F_j)$ and $\mathrm{Point}(I_k)$ represent the number of matched points between a frame image $F_j$ and a Web image $I_k$, the number of extracted SURF features from $F_j$ and the number of extracted SURF features from $I_k$, respectively.

### 4.3. Relevant shot selection

As a method on visual-feature-based shot ranking, we also employ VisualRank method [7] following [9]. While in [9] shots from videos which have larger tag relevance scores were assigned higher probability of being ranked to the top, in our approach we assign higher probability to the shots that are most similar to the selected set of action Web images. We believe that when tags are too noisy, tag relevance based shot ranking will not help to rank relevant shots to the top.

Eq.(3) represents an equation to compute VisualRank.

$$\boldsymbol{r} = \alpha S\boldsymbol{r} + (1 - \alpha)\boldsymbol{p} \quad (0 \leq \alpha \leq 1) \quad (3)$$

where $S$ is the column-normalized similarity matrix of images, $\boldsymbol{p}$ is a damping vector, and $\boldsymbol{r}$ is the ranking vector with each element which represents a ranking score of a corresponding image. $\alpha$ plays a role to control the extent of effect of $\boldsymbol{p}$. Commonly, $\alpha$ is set as 0.85. The final value of $\boldsymbol{r}$ is estimated by updating $\boldsymbol{r}$ iteratively with Eq.(3). Note that $S$ is calculated as similarity matrix of shots using their histograms of spatio-temporal features [10].

Although $\boldsymbol{p}$ is set as a uniform vector in original VisualRank equation [7], $\boldsymbol{p}$ is known as a bias vector which affects the final value of $\boldsymbol{r}$ since the rank scores of the biased images would become higher than unbiased ones. In our previous work [9], damping vector is proposed to be calculated based on tag relevance scores of videos. We defined two kinds of bias vectors as follows:

$$p_i^{(1)} = \begin{cases} 1/k & (i \leq k) \\ 0 & (i > k) \end{cases} \quad (4)$$

$$p_i^{(2)} = \begin{cases} Sc(V_i)/C & (i \leq k) \\ 0 & (i > k) \end{cases} \quad (5)$$

$$\text{where } C = \sum_{j=1}^{k} Sc(V_j)$$

where $Sc(j)$ represents the tag relevance score of the video from which shot $j$ was extracted.

The bias vector (1) represented in Eq.(4) is defined by giving uniform bias values to the elements corresponding to the top $k$ shots regarding tag-based video scores, while the bias vector (2) represented in Eq.(5) is defined by setting the bias vector as the normalized values proportional to the tag-based scores within the top $k$ shots. These method of calculating damping vector may work effectively in the case where tags information is reliable. In reverse, when tags are too noisy, tag relevance score will be noisy as well.

In this paper, to bias the shots we propose to give larger values on the elements more similar to the selected action images with the following equation:

$$p_i^{(3)} = \frac{\exp(\gamma \mathrm{SI}(S_i))}{\sum_{j=1}^{n} \exp(\gamma \mathrm{SI}(S_j))} \quad (6)$$

In the experiments, we set a constant value $\gamma$ as $\log 3$.

## 5. Experiments

We examine the effectiveness of our approach on 28 human action categories and 8 non-human action categories whose precision is quite low in [9]. We use the same video dataset and the same evaluation method as [9]. So that precision here means the precision rate over 100 top ranked shots (Precision@100). We try several values for number of selected images $N = 10, 20, 30$ and 50. $N = 0$ refers to the precision obtained by [9]. The results are shown in Figure 3 and 4 for human actions and non-human actions, respectively. Each result is the average precision over tested action categories. The results of all categories are shown in Table 1 (human actions) and Table 2 (non-human actions).

The results show that for both human actions and non-human actions, using $N$ = 20 or $N$ = 30 images brings best performance by enhancing the precision approximately 6% and 16% in average in the case of human actions and non-human actions, respectively. Especially, by referring Table 1 and 2, we can see that our approach could improve significantly the results of 10 human action categories including "bake+bread", "jog", "squat", "swim+breaststroke",

"serve+volleyball", "smile", "cook+rice", "grill+fish", "swim+butterfly", "tie+tie" and 6 non-human action categories including "falling+leaves", "snow+falling", "typhoon", "airplane+flying", "earthquake", "waterfall".

However, our proposed framework did not achieve good results on some categories such as "slap+face" and "wash+clothes". To explain for this, we think of the following two cases:

(1) human-detection-based image selection selects very few relevant images

(2) shots-images similarity calculation method is not effective

The first case corresponds to categories like "slap+face". Figure 5 shows some first results of image selection step for "slap+face". We can see that most of selected images are irrelevant to the action. That can explain why introducing these images causes a decrease in precision on "slap+face" category. The second case corresponds to categories like "wash+clothes". In this case, even image selection actually succeeds in selecting relevant images of the action (Figure 6), applying these action images cannot help to select more shots corresponding to the action. The reason is that, not the selected action images, but the variety of the action itself causes failure to shots-images similarity calculation so that similarity-with-images-based shot ranking cannot detect relevant shots correctly. While selected images contain mainly "outdoor washing clothes using hands" (Figure 6), most of the downloaded videos are about "indoor washing clothes using washing machine" (Figure 7).

Discussing about the appropriate number of images to use, as results shown in Figure 3 and 4, the precision increases while number of images is going up from 10 to 20 but decreases while number of images is rising from 30 to 50. That means it is not always true that the larger number of images is, the more relevant shots will be detected. Our intuition is that, here we use Web images as training source, and it is well known that in general, top retrieved Web images are more related to the query keyword, so using just the top images should obtain better results. To verify this intuition, we evaluate the precision rates of image selection by counting number of relevant images over all selected images and show the results in Figure 8 (human actions) and Figure 9 (non-human actions). The results show that the proportion of relevant images over $N$ = 10, 20 and 30 selected images are quite high while it is comparatively low when $N$ = 50.

## 6. Conclusions

In this paper, we apply Web images to the problem of automatically extracting from Web video shots corresponding to specific actions. In case of human actions, we apply Poselets [2] to detect human, and then use human detected images to improve that relevant video shots extrac-

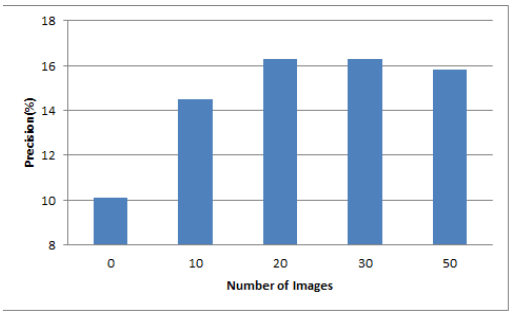


Figure 3. The mean precision of the top 100 ranked video shots over 28 human action categories.

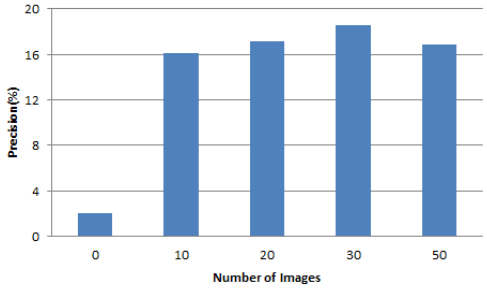


Figure 4. The mean precision of the top 100 ranked video shots over 8 non-human action categories.

Table 1. Results of 28 human action categories depending on number of selected images

| Action | [9] | N=10 | N=20 | N=30 | N=50 |
|---|---|---|---|---|---|
| slap+face | 20 | 16 | 14 | 13 | 17 |
| read+book | 19 | 21 | 22 | 23 | 24 |
| squat | 19 | 34 | 38 | 32 | 35 |
| row+dumbbell | 16 | 23 | 23 | 24 | 22 |
| wash+clothes | 15 | 9 | 10 | 10 | 9 |
| wash+dishes | 15 | 23 | 21 | 25 | 24 |
| comb+hair | 14 | 12 | 15 | 12 | 23 |
| drink+coffee | 14 | 8 | 10 | 9 | 16 |
| swim+breaststroke | 13 | 23 | 27 | 31 | 24 |
| cry | 12 | 4 | 6 | 5 | 4 |
| eat+sushi | 12 | 13 | 13 | 11 | 10 |
| serve+tennis | 11 | 14 | 18 | 15 | 14 |
| tie+tie | 11 | 18 | 17 | 23 | 30 |
| boil+egg | 9 | 4 | 8 | 6 | 6 |
| head+ball | 9 | 5 | 9 | 7 | 6 |
| swim+backstroke | 9 | 10 | 12 | 14 | 12 |
| take+medicine | 8 | 5 | 8 | 7 | 6 |
| serve+volleyball | 7 | 20 | 24 | 31 | 23 |
| swim+butterfly | 7 | 29 | 33 | 31 | 36 |
| bake+bread | 6 | 18 | 19 | 18 | 14 |
| cook+rice | 6 | 15 | 16 | 15 | 13 |
| grill+fish | 5 | 21 | 23 | 26 | 17 |
| jog | 5 | 15 | 19 | 21 | 20 |
| pick+apple | 5 | 8 | 10 | 9 | 10 |
| slice+apple | 5 | 2 | 4 | 2 | 6 |
| bowl+ball | 4 | 18 | 17 | 15 | 5 |
| smile | 4 | 16 | 17 | 18 | 15 |
| kiss | 2 | 2 | 4 | 3 | 2 |
| Average | 10.1 | 14.5 | 16.3 | 16.3 | 15.8 |

Table 2. Results of 8 non-human action categories depending on number of images

| Action | [9] | N=10 | N=20 | N=30 | N=50 |
|---|---|---|---|---|---|
| explosion | 0 | 4 | 5 | 5 | 1 |
| falling+leaves | 3 | 12 | 14 | 16 | 9 |
| snow+falling | 0 | 18 | 21 | 22 | 24 |
| typhoon | 4 | 21 | 25 | 29 | 24 |
| airplane+flying | 2 | 29 | 30 | 32 | 27 |
| earthquake | 7 | 26 | 24 | 25 | 23 |
| heavy+rain | 0 | 4 | 3 | 3 | 4 |
| waterfall | 0 | 15 | 15 | 17 | 15 |
| Average | 2 | 16.1 | 17.1 | 18.6 | 15.9 |

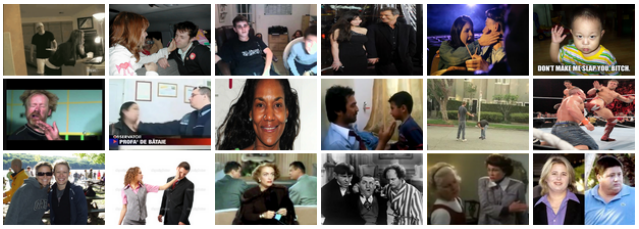

Figure 5. The top 18 Web images selected by Poselets for "slap+face" action. We can see that most of them are irrelevant to the action.

Figure 6. The top 18 Web images selected by Poselets for "wash+clothes" action. We can see that most of them represent "outdoor washing clothes using hands"

Figure 7. Some video shots of "wash+clothes" automatically extracted from the Web video dataset. We can see that most of them present "indoor washing clothes using washing machine"

tion. For non-human actions, we simply use top retrieved images. The results demonstrated the effectiveness of our proposed framework on categories whose precision is quite low in [9].

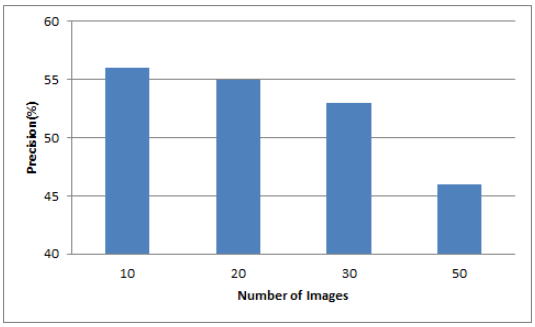


Figure 8. The mean precision of the top $N$ relevant Web images over all the 28 human actions

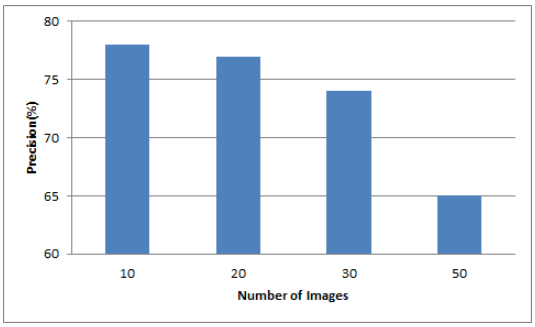


Figure 9. The mean precision of the top $N$ relevant Web images over all the 10 non-human actions

## References


[1] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, 2005. 1

[2] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, 2009. 1, 2, 3, 5

[3] N. I. Cinbins, R. G. Cinbins, and S. Sclaroff. Learning actions from the web. In *ICCV*, pp. 995–1002, 2009. 2

[4] L. Duan, D. Xu, I. W. Tsang, and J. Luo. Visual event recognition in videos by learning from web data. In *CVPR*, 2010. 2

[5] B. Herbert, E. Andreas, T. Tinne, and G. Luc. Surf: Speeded up robust features. *CVIU*, pp. 346–359, 2008. 2, 3

[6] H. A. Jhuang, H. A. Garrote, E. A. Poggio, T. A. Serre, and T. . Hmdb: A large video database for human motion recognition. In *ICCV*, 2011. 1

[7] Y. Jing and S. Baluja. Visualrank: Applying pagerank to large-scale image search. *PAMI*, 30(11):1870–1890, 2008. 2, 3, 4

[8] J. Liu, J. Luo, and M. Shah. Recognizing realistic action from videos "in the wild". In *CVPR*, 2009. 1

[9] D. H. Nga and K. Yanai. Automatic construction of an action video shot database using web videos. In *ICCV*, 2011. 1, 2, 3, 4, 5, 6

[10] A. Noguchi and K. Yanai. A surf-based spatio-temporal feature for feature-fusion-based action recognition. In *ECCV WS on Human Motion: Understanding, Modeling, Capture and Animation*, 2010. 2, 4

[11] K. Schindler and L. van Gool. Action snippets: How many frames does human action recognition require? In *CVPR*, 2008. 1

[12] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *ICPR*, pp. 32–36, 2004. 1

[13] C. Thurau and V. Hlavac. Pose primitive based human action recognition in videos or still images. In *CVPR*, 2008. 1

[14] G. Toderici, H. Aradhye, M. Pasca, L. Sbaiz, and J. Yagnik. Finding meaning on YouTube: Tag recommendation and category discovery. In *CVPR*, pp. 3447–3454, 2010. 2

[15] Z. Wang, M. Zhao, Y. Song, S. Kumar, and B. Li. YouTubeCat: Learning to categorize wild web videos. In *CVPR*, 2010. 2

[16] D. Weinland and E. Boyer. Action recognition using exemplar-based embedding. In *CVPR*, 2008. 1

[17] W. Yang, Y. Wang, and G. Mori. Recognizing human actions from still images with latent poses. In *CVPR*, 2010. 1