Look for the Change: Learning Object States and State-Modifying Actions from Untrimmed Web Videos Supplementary Material

Tomáš Souček[1] Jean-Baptiste Alayrac[2] Antoine Miech[2]
Ivan Laptev[3] Josef Sivic[1]
[1]CIIRC CTU [2]DeepMind [3]ENS/Inria
tomas.soucek@cvut.cz
https://data.ciirc.cvut.cz/public/projects/2022LookForTheChange/

In this Supplementary Material, we start by providing additional details on the selection of hyper-parameters for the video relevance weight $\omega(v)$ in Section A. In Section B, we describe training details, hyper-parameters, and data preprocessing steps. We then provide details on the action selection and annotation process for our new **ChangeIt** dataset together with additional statistics in Section C. Section D shows the effect of different feature extractors on the performance. Further, we report per-class quantitative results and show qualitative results in Section E. Lastly, we discuss broader impact of our work in Section F. On the project website[1], we also provide a video showing our model's predictions on handful of dataset videos.

A. Video relevance weight $\omega(v)$

Video relevance weight $\omega(v)$ (Equation (8) in the main paper) contains a temperature hyper-parameter τ and a centering hyper-parameter θ. We choose τ globally by grid search. However, θ needs to be chosen individually per each category as the distribution of relevance scores r_v varies greatly between the categories. Possible value for θ would be a median or other fixed quantile of the score distribution. Instead, we opt for solution that does not require manual choice of a quantile and follows from our observation that the score distributions for different categories are often bimodal with the two modes corresponding to the relevant and irrelevant videos. We compute θ for each category $\mathcal{C}$ by minimizing the intra-class variance of the category relevance scores r_v as:

$$\arg\min_{\theta} \text{var}_{v\in\mathcal{C}}\{r_v : r_v < \theta\} + \text{var}_{v\in\mathcal{C}}\{r_v : r_v > \theta\}. \quad (1)$$

We validate our approach by computing the number of annotated videos with $r_v > \theta$, as the annotated (test) videos contain the object of interest with certainty. Using this method, we retrieve 80.7% of the annotated videos while retrieving only 59.5% of all dataset videos. The fixed quantile method retrieves 77.8% of the annotated videos while retrieving the same total number of videos.

[1]https://data.ciirc.cvut.cz/public/projects/2022LookForTheChange/

B. Training details

Hyper-parameters. We use a batch size of 48 randomly sampled videos. We optimize the classifiers using stochastic gradient descent with a momentum of 0.9 and L_2 penalty of 0.001. We sample five ($\delta = 2$) positive examples for both the action and states and use temperature $\tau = 0.001$ for the relevance weight $\omega(v)$. In order to compute the features used for the noise adaptive video relevance score r_v we use the 2D ResNeXT backbone only. The distance parameter for action negatives is fixed to $\kappa = 60$. The action loss is weighted by $\lambda = 0.2$ and the action positives are weighed by $\mu = 10$.

Data preprocessing. We apply data augmentation to the inputs as follows: each video is randomly rotated by up to five degrees and horizontally flipped with probability 50%. Then each video's sides are randomly cropped by 16% and with 80% chance one change of brightness, color, or contrast is applied. The same augmentation is applied on all frames of the video to ensure temporal consistency. The importance of data augmentation is shown in the ablation section of the main paper. The visual features are extracted by running an image feature extractor on one frame per second and a video feature extractor on 25 frames per second of the original video. The output of the video extractor is temporally downsampled to match the 1fps sampling rate of the image features.

C. ChangeIt dataset details

Action selection. The 44 manipulating actions of the **ChangeIt** dataset were selected as follows: **(i)** A list of candidate verb-object pairs was constructed by combining top verbs and objects sorted by the sum of their concreteness score [3]. **(ii)** Verb-object pairs corresponding to a visual change of an object state were manually selected. **(iii)** Too general pairs were manually removed from the list, *e.g. open a door*. **(iv)** Similar or consecutive actions were joined into a single verb-object category, *e.g. cut* and *peel an avocado*. **(v)** YouTube API was queried for videos corresponding to a search term "How to *verb* an *object*?", "*verb* an *object*" or similar. **(vi)** Categories with a small number of videos were removed.

Action-state annotation. We hold out a small fraction of videos and manually annotate them for evaluation. For each state-changing action, 30 videos are randomly sampled and annotated. As the videos are uncurated, some of them do not contain the object nor the action of interest. Thus, not all the held-out videos are exhaustively annotated with states and actions.

Each video is divided into one second time intervals and each interval is assigned one of the following labels: *background*, *initial state*, *action*, *end state*. We assign the *initial* (resp. *end*) *states* to frames containing an object of interest that is visually similar to its appearance right at the beginning (resp. end) of the manipulating action. The *background* label is used when the object of interest is not clearly visible within the time interval. In total, 667 videos with combined duration of 48 hours were annotated, yielding 15 videos per state-changing action on average. Given the ratio between annotated and deliberately unannotated videos processed by our annotators, we estimate that approximately half of the videos in the dataset are *noisy* in the sense that either the object, action or both are not clearly visible in the video. The proportion of annotated labels in the test set is the following: *initial state* 5%, *end state* 12%, *action* 42% with the rest being labelled as *background*.

Video statistics. Figure 1 shows the number of videos in our **ChangeIt** dataset for given video lengths. Only less than 15% of the videos are shorter than one minute, and almost 60% of the videos are longer than three minutes. On average, *cherry pitting* has the shortest videos with the average duration of 2.6 minutes, the longest videos on average are in *outlet installing* class with the mean duration of 7.2 minutes. The number of videos in each class varies from 265 in *juice pouring* to 1914 in *tortilla wrapping*. The list of all dataset classes is shown in Table 2.

D. Different feature extractors

In our model we use 2D ResNeXT pre-trained on ImageNet-21K [4] and 3D TSM ResNet50 pre-trained on HowTo100M and AudioSet [2] as feature extractors. Besides those, we also tested CLIP [6] model and S3D trained using MIL-NCE loss on HowTo100M dataset [5]. We show results of the tested models in Table 1. In case of CLIP, we observe substantial improvements in our metrics can be made when the last projection layer is removed (†). We can also see that TSM-ResNet features contain additional information and improve action precision when used jointly with both ResNeXT and CLIP, even though their individual performance is low. On the contrary, using jointly only image extractors does not yield any benefit in action detection.

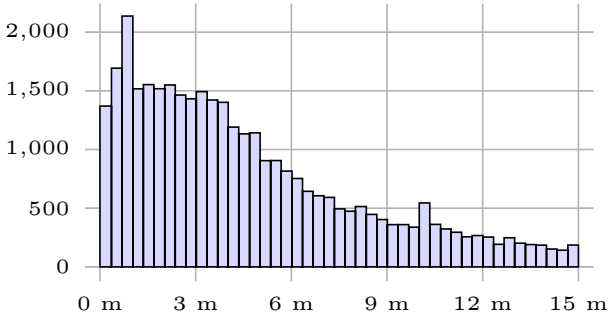

Figure 1. Histogram of video lengths in our **ChangeIt** dataset.

Type	Feature extractor	St prec.	Ac prec.
Video	MIL-NCE S3D [5]	0.22	0.49
	TSM-ResNet [2]	0.25	0.56
Image	CLIP ViT-B/16 [6]	0.33	0.61
	CLIP ViT-B/16† [6]	**0.35**	0.63
	ResNeXT [4]	0.33	0.66
	ResNeXT + CLIP ViT-B/16†	0.34	0.66
Both	CLIP ViT-B/16† + TSM-ResNet	0.34	0.66
	ResNeXT + MIL-NCE S3D	0.34	0.66
	ResNeXT + TSM-ResNet	**0.35**	**0.68**

† Without the last projection layer.

Table 1. Comparison of different feature extractors in our method.

E. Additional results

In this section we report per-class results, show additional examples and provide a qualitative analysis.

Per-class results. In Table 2 we report action and state precision for all dataset classes individually. We compare results of our approach to: (i) the state-of-the-art method for learning object states and state-modifyng actions by Alayrac *et al.* [1] as well as (ii) the zero-shot CLIP baseline [6]. Details of the experimental set-up for both these baseline methods are in Section 5.4 in the main paper.

Qualitative results. Figures 2, 3, 4 and 5 show qualitative results for a set of selected classes. Figures 2, 3 and 4 il-

lustrate predictions for classes where the learning of object state and action classifiers is successful. Figure 5 shows the main limitations of our method, as described in Limitations in Section 5.4 in the main paper. Example videos for each class were chosen as those with the highest prediction scores (within their class), where the prediction score is defined as:

$$\max_{l \in \mathcal{D}_v} h_1(x_{l_{s_1}}) \cdot g(x_{l_a}) \cdot h_2(x_{l_{s_2}}), \tag{2}$$

where $h_1(x_{l_{s_1}})$ is the classifier score of the initial state, $g(x_{l_a})$ is the classifier score of the action, and $h_2(x_{l_{s_2}})$ is the classifier score of the end state. We do not show videos with clearly visible faces, videos of poor quality or uninteresting static videos. Therefore, the shown videos may not be the highest ranked but (up to) fourth or fifth in the list of highest scoring videos for a given class.

Analysis of the results. As shown in Table 2 as well as Figures 2, 3, 4 and 5, there can be large differences in performance between individual classes. We attribute these differences to the fact that some actions and object states are visually clearly defined whereas some can be visually ambiguous. Here are some of our qualitative findings:

(a) Peeling, slicing, chopping, cutting. We observe strong performance for peeling or cutting any type of food. The actions and the states are visually distinct. The action has clear start and end, it is accompanied by visually distinct objects such as a knife. Also, there are many how-to videos for these actions.

(b) Frying, grinding, melting. The initial and the end states of objects manipulated by actions such as frying or melting are visually distinct but the action itself is long without clearly defined start and end. For example the causal start of the melting action is turning on the heat source, which requires deep context understanding and even may not be shown in a video.

(c) Whisking, rolling, cleaning, tying. For whisking or rolling, the action is clearly defined by an accompanying object such as a whisk or a roller but the visual difference between the object states can be rather small. For example, a cream is still white no matter the action, a dough is only a bit thinner, *etc*.

(d) Opening, pouring. There are not that many YouTube instructional videos for primitive actions such as pouring, thus the videos of these categories contain large variety of sub-actions, advertisement clips, and other noise making it much harder to correctly discover the states and the action.

(e) Drilling, wrapping, cutting. These classes often have clearly defined actions and visually distinct states, but there is a large variance of appearance of the initial and the end state across different videos. For example, in some videos only a single hole is drilled but in other videos five new holes are drilled next to a handful of existing ones.

(f) Inflating, pitting, starting, removing. For some of these classes, it is difficult to pinpoint the exact location of the action in the video. Also, the visual difference between the initial and the end state can be quite small. For example, a partially deflated ball can be often recognized only by touch, not by vision.

We believe that additional forms of supervision, such as incorporating the audio signal or the language narration, may be needed to learn some of the hardest object changes.

F. Potential negative societal impact

Our work is based on models trained without human annotation. However, the models are still subject to biases in the training data. We gather our training dataset from the YouTube platform, which makes our results dependent on the availability and the quality of the videos uploaded to the site. In addition, the content on the platform is potentially not uniformly distributed across countries, ages, ethnic groups, *etc*. Thus our models can under-perform for some actions, for example, conducted only by minorities. Also, if an action has multiple possible realizations, there is a risk of learning only the realization that is the most prevalent in the data.

References

[1] Jean-Baptiste Alayrac, Ivan Laptev, Josef Sivic, and Simon Lacoste-Julien. Joint discovery of object states and manipulation actions. In *ICCV*, 2017. 2, 4

[2] Jean-Baptiste Alayrac, Adria Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *NeurIPS*, 2020. 2

[3] M. Brysbaert et al. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 2014. 2

[4] Pascal Mettes, Dennis C. Koelma, and Cees G. M. Snoek. Shuffled imagenet banks for video event detection and search. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2020. 2

[5] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-End Learning of Visual Representations from Uncurated Instructional Videos. In *CVPR*, 2019. 2

[6] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 2, 4

State-modifying action	St prec. Ours	[1]	[6]	Ac prec. Ours	[1]	[6]
(a) *Visually distinct states and actions*						
Apple Peeling/Cutting	**0.46**	0.41	0.41	0.79	0.69	**0.81**
Avocado Peeling/Slicing	**0.44**	0.38	0.28	**0.92**	0.88	0.88
Beer Pouring	**0.37**	0.22	0.25	**0.79**	0.62	0.44
Corn Peeling	**0.51**	0.45	0.19	**0.68**	0.57	0.48
Dragon Fruit Peeling/Cutting	0.47	**0.48**	**0.48**	**0.94**	0.90	**0.94**
Eggs Peeling	0.36	0.25	**0.38**	**0.60**	0.25	0.50
Garlic Peeling/Chopping	**0.44**	0.39	0.28	**1.00**	0.89	**1.00**
Onions Peeling/Chopping	**0.49**	0.40	0.23	**0.91**	0.73	0.80
Paper Plane Folding	0.37	**0.38**	0.21	**1.00**	**1.00**	**1.00**
Pineapple Peeling/Slicing	0.28	**0.33**	0.12	**1.00**	0.90	**1.00**
T-shirt Dyeing	**0.48**	0.47	0.35	0.86	**1.00**	0.82
Tortilla Wrapping	0.39	0.27	**0.54**	**0.71**	0.53	0.53
(b) *Visually distinct states, actions with unclear boundaries*						
Bacon Frying	**0.40**	0.22	0.25	0.20	0.39	**0.61**
Chocolate Melting	**0.54**	0.32	0.47	**0.65**	0.35	0.59
Coffee Grinding	0.47	0.29	**0.50**	0.25	**0.50**	0.17
Potatoes Frying	**0.32**	0.26	0.27	**0.98**	0.96	0.95
(c) *Visually distinct actions, small visual changes between states*						
Cake Frosting	0.22	**0.25**	0.19	0.50	0.42	**0.79**
Cream Whipping	0.36	**0.39**	0.30	0.42	**0.50**	0.32
Dough Rolling	0.27	0.27	**0.37**	0.62	0.60	**0.73**
Eggs Whisking	**0.30**	0.29	0.25	**0.86**	0.64	0.64
Fish Filleting	0.21	0.22	**0.23**	0.87	0.90	**0.95**
Pan Cleaning	0.46	**0.53**	0.36	**0.96**	0.78	0.94
Rubik's Cube Solving	**0.16**	0.14	0.03	0.91	0.67	**1.00**
Shoes Cleaning	0.18	0.21	**0.26**	**0.90**	0.84	0.89
Tie Tying	0.46	**0.50**	0.12	**1.00**	**1.00**	**1.00**
Ribbon/Bow Tying	**0.28**	0.17	0.19	**0.98**	0.94	0.89
Rope/Knot Tying	**0.36**	0.25	0.29	**0.83**	**0.83**	0.58
(d) *Not many how-to videos available*						
Butter Melting	0.22	**0.25**	0.17	**0.67**	0.50	0.50
Candle Lighting	0.50	0.31	**0.62**	**0.29**	0.12	0.00
Champagne Opening	**0.45**	0.39	0.36	**0.24**	0.14	0.14
Juice Pouring	0.32	0.32	**0.45**	0.12	0.00	**0.36**
Milk Boiling	0.28	0.26	**0.35**	0.21	**0.29**	0.24
Milk Pouring	0.20	**0.40**	0.20	0.50	0.30	**0.70**
Tea Pouring	**0.17**	0.08	**0.17**	**0.39**	0.00	0.17
(e) *Distinct states and actions but high variance in appearance*						
Gift/Box Wrapping	0.20	0.21	**0.26**	0.84	**0.95**	**0.95**
Outlet Installing	0.23	0.15	**0.35**	**0.87**	0.77	0.69
Pancake Flipping	**0.36**	0.31	0.29	**0.19**	**0.19**	0.14
Tile Cutting	0.28	0.35	**0.45**	**0.63**	0.50	0.60
Tree Cutting	**0.40**	0.19	0.22	**0.70**	0.61	0.56
Wood Drilling	0.29	0.14	**0.41**	0.45	0.36	**0.73**
(f) *Minimal visual change of states, actions with unclear boundaries*						
Ball Inflating	**0.22**	0.20	0.05	0.40	**0.60**	0.30
Cherries Pitting	0.31	0.25	**0.38**	**0.50**	0.12	**0.50**
Grill Starting	**0.33**	0.12	**0.33**	**0.83**	0.58	0.58
Weed Removing	0.33	0.41	**0.45**	**0.70**	0.64	0.55
Average	**0.35**	0.30	0.30	**0.68**	0.59	0.63

Table 2. **Per-class state and action precision** on our **ChangeIt** dataset. Our approach improves on average over the state-of-the-art approach of [1] and CLIP ViT-L/14 [6].

Figure 2. **Additional example results for four different DIY classes, "Paper plane folding", "Ribbon tying", "T-shirt dyeing" and "Outlet installing".** For each class we show the temporal localization of the initial state, state-modifying action, and the end state in three different example videos. Note how our model is able to learn the object states and the state-modifying action despite the large appearance variation in the videos (viewpoint, environment, intra-class variation for both the object states and the action).

Figure 3. **Additional example results for four different meal preparation classes, "Chocolate melting", "Fish filleting", "Dragon fruit peeling" and "Bacon frying".** For each class we show the temporal localization of the initial state, state-modifying action, and the end state in three different example videos. Note how our model is able to learn the object states and the state-modifying action despite the large appearance variation in the videos (viewpoint, environment, intra-class variation for both the object states and the action).

Figure 4. **Additional example results for four different classes, "Cream whipping", "Beer pouring", "Pan cleaning" and "Rubik's cube solving".** For each class we show the temporal localization of the initial state, state-modifying action, and the end state in three different example videos. Note how our model is able to learn the object states and the state-modifying action despite the large appearance variation in the videos (viewpoint, environment, intra-class variation for both the object states and the action).

Figure 5. **Examples of typical failure modes.** (a) The model learns a different action than the expected action (here *holding a piece of onion* instead of *chopping onion*). (b) The model discovers consistent visual appearance in the videos, which is just an artefact of the editing process (here single-colored frame at the and of a video as an end state). (c) Some categories can have a large variance in the appearance of the initial and the end state across different videos (here removing a single plant or clearing the whole path). (d) Some categories are not well represented on YouTube (here videos of *making butter cookies* dominate the search results for query *melting butter*).