Identifying Interpretable Action Concepts in Deep Networks

Kandan Ramakrishnan*[13], Mathew Monfort*[13], Barry A McNamara [1]
Alex Lascelles [1], Dan Gutfreund[23], Rogerio Feris[23], Aude Oliva[13]
[1] MIT CSAIL, [2] IBM Research, [3] MIT-IBM Watson AI Lab

Abstract

A number of recent methods to understand neural networks have focused on quantifying the role of individual features. One such method, NetDissect identifies interpretable features of a model using the Broden dataset of visual semantic labels (colors, materials, textures, objects and scenes). Given the recent rise of a number of action recognition datasets, we propose extending the Broden dataset to include actions to better analyze learned action models. We describe the annotation process and results from interpreting action recognition models on the extended Broden dataset.

1. Introduction

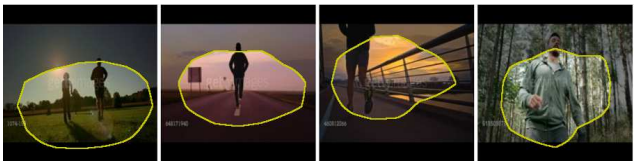
The success of Deep convolutional neural networks (DNNs) is partly due to their ability to learn hidden representations that capture the important factors of variation in the data. Previous works have visualized the units of deep convolutional networks by sampling image patches that maximize the activation of each feature [8] or by generating images that maximize each feature activation. Such visualizations show that individual features act as visual concept detectors. Features at lower layers detect concrete patterns such as textures or shapes while features at higher layers detect more semantically meaningful concepts such as dog heads or bicycle wheels. One tool for network interpretability (NetDissect) [1, 7] uses the Broden dataset (consists of objects, scenes, object parts, textures and materials) to evaluate individual units.

Recently, DNNs have shown significant progress in action recognition with the introduction of large-scale video datasets. However, while NetDissect with the Broden dataset is appropriate for networks trained on object or scene recognition, it does not include the ability to detect learned action concepts.

In this paper, we propose extending the Broden dataset to include actions so that we can more appropriately interpret action recognition networks. We describe our annotation process to collect images across action classes and select regions of importance for identifying each action. We then show results using our Action Region dataset together with the existing Broden set to identify learned interpretable action concepts in deep networks trained for action recognition. The Action Region dataset presented, and the code for integrating with NetDissect, will be made available online.

Sample videos Example frames from a few videos to show intra-class action variation

Action Regions Spatial localization of actions in single frames for network interpretation



Action Concepts Identifying interpretable action features

2. Identifying Action Features

To better analyze action models, we extend the Broden dataset to include actions. This is done by first building an image segmentation dataset for actions.

2.1. Annotation

We begin by collecting bounding box annotations via Amazon Mechanical Turk (AMT) for actions in images selected from videos, for which we use the Moments

*Denotes equal contribution

Category	Action concepts		
	Classes	Source	Samples
actions	210	Moment-Frames	23,244

Table 1: Statistics of action concepts included in the dataset.

in time dataset [5]. We extract a single frame from the center of 500 randomly selected videos for each of the 339 action classes from the dataset and present a binary annotation task to the workers on AMT asking if an action from the source videos label set is visible in the frame shown. This binary interface is very similar to that used for collecting the action labels for the Moments in Time dataset [5] with the main difference being the use of images rather than video. We run this task for at least 2 rounds of annotation to verify that the action is visible in each frame. We then take the set of verified action-frame pairs and pass them to a separate annotation interface on AMT that asks the workers to select the regions most important in the image for identifying the action. Multiple regions can be selected for an image as in the jogging example in Figure 3 and the workers are allowed to skip an image if there are no useful regions for detecting the action (i.e. the action is not visible in the image).

We run this region selection task through multiple rounds and only consider overlapping regions from the different rounds as most important for detecting the actions. After this stage the regions selected are cropped from the original images and passed through the binary annotation task previously described for a final verification that the actions are present and recognizable in the selected regions. After our complete annotation process our total set of verified images with segmented action regions consists of 23,244 images from 210 different classes. Figure 3 displays some examples of the selected regions collected through this process.

(a) Cracked (IoU 0.1)

(b) Typing (IoU 0.34)

Figure 2: Example of the same feature (290) evaluated using the (a) original Broden dataset and (b) the proposed Broden+Action dataset.

Category	Concepts	Interpretable Features
Broden	108	850
Action Regions	144	1971
Broden+Action Regions	193	1978

Table 2: Comparison of the number of concepts and interpretable features identified by NetDissect given the Broden dataset, the Action Region dataset and the combined dataset on block 4 of a ResNet50 trained for action recognition.

2.2. Action concept dataset

To integrate our new action concept dataset into the NetDissect framework, we first consider each selected region to be a mask on the segmented area of the image relating to the action. This is similar to part, material and object masks used for other segmentation datasets [9, 3, 6, 2]. With the data formatted in this manner we extend the Broden dataset to include our action segmentations and extract the set of learned action concepts detected via NetDissect. This process allows us to identify not just object, scene, texture and color concepts learned by our models, but action concepts as well. In Section 3 we show some of the key results from interpreting action networks in this way.

3. Experiments

To score and quantify the unit interpretability of a network we follow the same procedure as outlined in [1]. All experiments use a ResNet50 network [4] trained on the Moments in time dataset [5] for classification performance. We analyze features from the outputs of the residual blocks (referred to as block1, block2, block3 and block4 corresponding to conv2, conv3, conv4 and conv5) of the network.

3.1. Action Dissection

Using the approach described in Section 2 we are able to identify 144 action concepts learned in 1971 different features out of 2048 (Figure 4) units in the final convolutional layer (block4) of a Resnet50 network trained on the Moments in Time dataset. Figure 6 highlights some of the learned concepts. Interestingly the network seems to be recognizing the pattern of a person standing behind a podium as *preaching* which is definitely a common correlation in our dataset. Similarly, the network associates *crawling* with babies as many of our videos of crawling typically depict babies *crawling*. These are the types of data and class biases that are useful to identify via network interpretation that may have gone unnoticed without the ability to identify action concepts.

Table 2 highlights the fact that including actions in

bicycling floating grooming writing jogging

Figure 3: **Visualization of labelled regions**

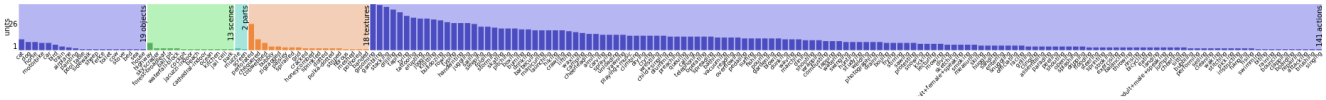
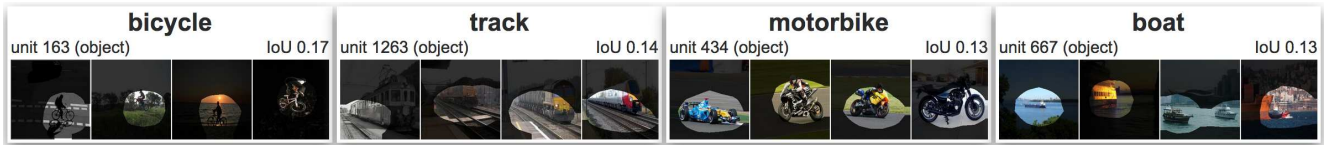


Figure 4: **Graph of learned action concepts ordered by the number of features associated with each concept.**

(a) **scene units**



(b) **object units**

Figure 5: **Visualization of scene and object units in block 4 of a ResNet50 trained for action recognition.**

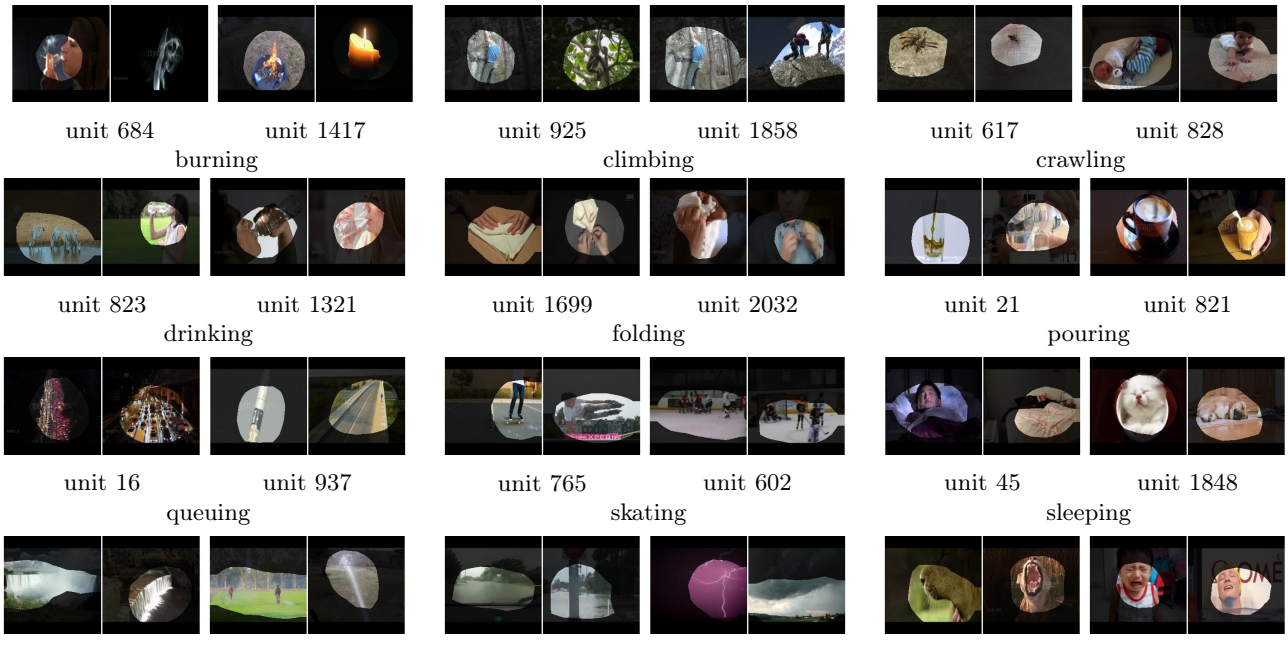


Figure 6: **Visualization of learned action concepts**

the Broden dataset helps to identify a much larger portion of the features in block 4 of a ResNet50 trained for action recognition. Without actions, NetDissect identified 108 concepts in 850/2048 features. If we only consider actions then we were able to identify 144 concepts in 1971 features. This large jump in the number

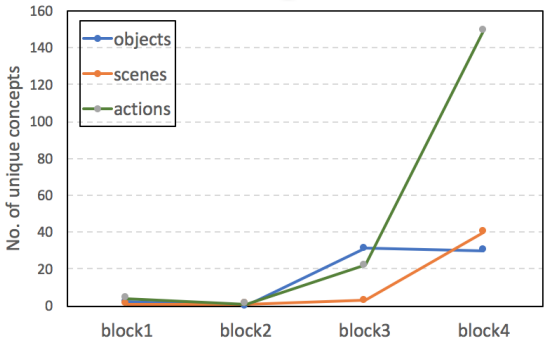


Figure 7: **ResNet block-wise interpretability** Visualize how different semantic concepts - objects, scenes and actions emerge across residual blocks of the ResNet50 network.

of interpretable features makes sense for the final block of a model trained for action classification and suggests that excluding action concepts misses a large amount of useful information each feature represents. Combining the original Broden set with the proposed Action Regions results in identifying a much larger number of concepts, 193 concepts in 1978/2048 features (96.5% of the features). The results from the combined set highlight that some of the features previously interpreted by the original Broden set as object or texture concepts are closely aligned with actions. For example, unit 13 was classified using the Broden set as learning the concept "potted plant" with an IoU of 0.06, but if we include action concepts the unit is found to be more correlated with the action "gardening" with an IoU of 0.15. Similarly, unit 290 was identified by Broden as learning the texture concept "cracked" with an IoU of 0.1 and including actions we found a greater association with the action "typing" with an IoU of 0.34. Features for identifying the ridges between the keys in the keyboards commonly found in actions of "typing" were correctly activating for the texture "cracked", however we can see from Figure 2 that the feature is more correlated with the action "typing".

3.2. Block-wise Interpretability

To understand how individual units evolve over residual blocks we evaluate the interpretability of features from different blocks of a resnet50 network trained for action recognition on the Moments in Time dataset [5] in terms of concepts such as objects, scenes, actions and textures. We observe that action concepts mainly emerge in the last convolutional block (block 4) of the model. It is interesting to note that the network also learns objects and scene concepts even if the model is not explicitly trained to recognize objects or scenes, as seen in Figure 5, suggesting that object and scene recognition aid in action classification.

4. Conclusion

We introduced Action Regions to the Broden dataset to allow for NetDissect to identify action concepts learned by interpretable features in networks trained for action recognition. We showed the resulting increase in identifying interpretable features and learned concepts and highlighted some interesting examples. Future work will focus on expanding feature interpretation for spatio-temporal networks trained for video understanding.

Acknowledgements: We thank David Bau for discussions on the Broden dataset [7]. This work was supported by the MIT-IBM Watson AI Lab, a Google faculty award to A.O, as well as the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/ Interior Business Center (DOI/IBC) contract number D17PC00341. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/IBC, or the U.S. Government.

References

[1] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Computer Vision and Pattern Recognition*, 2017.

[2] S. Bell, P. Upchurch, N. Snavely, and K. Bala. OpenSurfaces: A richly annotated catalog of surface appearance. *ACM Trans. on Graphics (SIGGRAPH)*, 32(4), 2013.

[3] X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] M. Monfort, A. Andonian, B. Zhou, K. Ramakrishnan, S. A. Bargal, T. Yan, L. Brown, Q. Fan, D. Gutfruend, C. Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–8, 2019.

[6] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille. The role of context for object detection and semantic segmentation in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[7] B. Zhou, D. Bau, A. Oliva, and A. Torralba. Interpreting deep visual representations via network dissection. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[8] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2921–2929, 2016.

[9] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ade20k dataset. *arXiv preprint arXiv:1608.05442*, 2016.