# Human Action Recognition with Still Images and Video

Video URL: https://youtu.be/2-xtDbTRq_w

Wendell Hom
froi@stanford.edu

## Introduction

Human Action Recognition is a challenging task in computer vision that has many important applications including security surveillance, healthcare and elderly care, and pedestrian monitoring in self-driving vehicles. The successful use of systems in these areas could mean the difference between substantial financial losses, or life and death in emergency situations.

With the adoption of Deep Learning, computer vision and HAR has enjoyed significant improvements in the last decade. Here we aim to use deep learning to help us classify the action category from both images and videos.

## Datasets

**The Stanford 40 Actions Dataset** is a collection of 40 different action categories with 180-300 images per category. This dataset was used to fine-tune our 2D CNN image models.

Figure 1a: Shooting an Arrow



Figure 1b: Riding a Horse

**The Kinetics-600 Dataset** is a collection of 600 action categories with at least 600 video clips per category. Each clip was approximately 10 seconds in length. This dataset was used to train our shallow 3D and deep 3D CNNs from scratch.

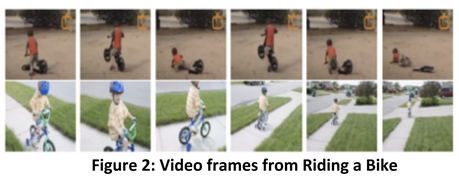

Figure 2: Video frames from Riding a Bike

Initially, we trained our shallow and deep 3D CNNs on a 15-category and then a 27-category subset from the Kinetics-400 dataset, a precursor to the Kinetics-600.

Afterwards we attempted to train the shallow and deep 3D CNNs using the full Kinetics-600 dataset.

## Models

For still image HAR, we used models pre-trained on Imagenet as well as a 2-branch model that included a human localization and an action classification branch as shown in Fig 3.

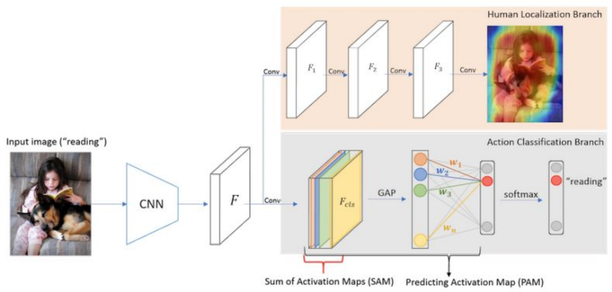


Figure 3: The 2-branch model. The CNN base was a pre-trained Inception-ResNet-v2 model

For video HAR, we used a 2D CNN, a shallow 3D CNN (Fig. 4), and a deep 3D CNN which is a 3D version of ResNeXt-101 with a cardinality of 32.

With the 2D CNN, we classified each frame and then averaged the scores across all frames and return the category with the highest score. With the 3D CNNs, we used 16 frames for the temporal depth of the model, so for videos we would classify non-overlapping windows of 16 frames, average the scores, and return the category with the highest score.

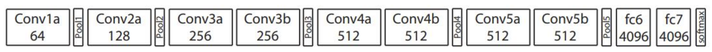


Figure 4: The C3D model depicted above is a shallow 3D CNN model

## Challenges

Data augmentation for the two branch model requires the same augmentation to be applied to the ground truth human mask.

Kinetics-600 dataset is huge, with 350,000 successfully downloaded training videos and 27,000 validation videos.

Video formats are heavily compressed. Extracting frames required sequentially reading each frame. Reading and writing numpy arrays to compressed hdf5 format required ~4TB of disk space.

Long pre-processing time. Use of multi-processing = True and workers = 12 helped significantly.

Long training times. Use of mixed precision 16-bit computations allowed 2x batch size, reducing training time of each epoch from 11+ hrs to 3hrs.

## Results for Images

| Method | Pixels | Parameters | Accuracy |
|---|---|---|---|
| MobileNet-v2 | 224 | 14.1 M | 65.1% |
| VGG-16 | 224 / 500 | 19.5 M | 56.6% / 59.6% |
| VGG-19 | 224 / 500 | 24.8 M | 55.2% / 57.8% |
| ResNet-50 | 224 / 500 | 42.5 M | 2.7% / 1.9% |
| **Inception-ResNet-v2** | **224 / 500** | **68.5 M** | **71.6% / 84.1%** |
| Inception-v3 | 224 / 500 | 40.7 M | 64.5% / 83.7% |
| Loss Guided Act. | 500 | 75.9 M | 84.7% |

Table 1: Results for Image Action Recognition

For images, the MobileNet-v2 model gave us the best bang for the buck in terms of model size and accuracy. It achieved 65.1% accuracy for the 224x224 RGB model with a higher accuracy than the Inception-v3 version while using almost 3 times fewer parameters.

The ResNet-50 model ended up drastically overfitting the training set and the accuracy on the validation set was the same as random guessing. Looking at Fig. 5a, one can see that it essentially classified everything not seen in the training set into one category, in this case "phoning".



(a) ResNet-50 (b) Inception-ResNet-v2

Figure 5: Confusion Matrix. ResNet-50 model failed to generalize

The 2-branch Loss Guided Activation model gave us the highest validation set accuracy, however, it failed to generalize well to images we downloaded from the web.

Increasing the pixel resolution to 500x500 was mostly beneficial to models that greater modeling capability such as the Inception-ResNet-v2 and Inception-v3 models, but had only a minor impact to VGG-16 and VGG-19 models.

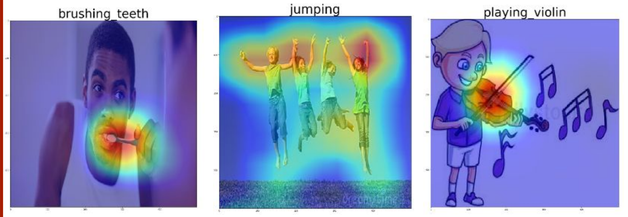


Figure 6: Visualizing the Class Activation Maps

## Results for Videos

| Method | Parameters | Acc. 15-class | Acc. 27-class | Acc. 600-class |
|---|---|---|---|---|
| 2D CNN | 68.5 M | 59.0% | - | - |
| C3D | 63.4 M | 42.5% | 43.7% | 35.2% |
| ResNeXt-101 | 47.8 M | 41.1% | 37.6% | 24.4% |

Table 2: Results for Video Action Recognition

For videos, the 2D Inception-ResNet-v2 CNN we fine-tuned using Stanford 40 action dataset gave us the best performance at 59.0% accuracy for a 15-category subset that was common between Stanford 40 and Kinetics dataset. This gives us a hint that training 3D CNNs is not an easy task.

Contrary to our expectations, the accuracy rate of the 3D CNNs for the 27-category subset and the full 600-category Kinetics dataset was even lower. However, this is mainly because it becomes much more difficult to train these models due to the high computation resources required, making it very difficult to experiment with various hyperparameters.

One other surprising finding was that while the ResNeXt-101 has fewer parameters than the shallow C3D model, it required a lot more memory, largely due to its depth since the model has to cache values from the forward propagation.

## Future Work

Provide a ground truth human mask to the 2-branch Loss Guided Activation model to see if that helps with inference in the wild.

Add Augmentation to 2-branch model.

For training 3D CNNs, it may provide better results to focus the GPU resources on training the 27-category or even Kinetics-400 dataset.

Replace Adam optimizer with the SGD optimizer. It takes longer to converge, but the models generalize better.

Try training on multiple GPUs.

## References

K. Hara, H. Kataoka, Y. Satoh. ***Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet?*** In CVPR, 2018.

W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, A. Zisserman. ***The Kinetics Human Action Video Dataset.*** In arXiv:1705.06950, 2017.

L. Liu, R. T. Tan, S. You. ***Loss Guided Activation for Action Recognition in Still Images.*** In arXiv:1812.04194, 2018.