Movies Tags Extraction Using Deep Learning

U. A. Khan, N. Ejaz, M. A. Martínez-del-Amor, H. Sparenberg
Moving Picture Technologies, Fraunhofer Institute for Integrated Circuits (IIS)
Am Wolfsmantel 33, 91058 Erlangen, Germany
{khanur,ejaznd,miguel.martinez,heiko.sparenberg}@iis.fraunhofer.de

Abstract

Retrieving information from movies is becoming increasingly demanding due to the enormous amount of multimedia data generated each day. Not only it helps in efficient search, archiving and classification of movies, but is also instrumental in content censorship and recommendation systems. Extracting key information from a movie and summarizing it in a few tags which best describe the movie presents a dedicated challenge and requires an intelligent approach to automatically analyze the movie. In this paper, we formulate movies tags extraction problem as a machine learning classification problem and train a Convolution Neural Network (CNN) on a carefully constructed tag vocabulary. Our proposed technique first extracts key frames from a movie and applies the trained classifier on the key frames. The predictions from the classifier are assigned scores and are filtered based on their relative strengths to generate a compact set of most relevant key tags. We performed a rigorous subjective evaluation of our proposed technique for a wide variety of movies with different experiments. The evaluation results presented in this paper demonstrate that our proposed approach can efficiently extract the key tags of a movie with a good accuracy.

1. Introduction

The sheer volume of movies produced these days poses a huge challenge to their manual processing. Human generated meta data is generally not sufficient to describe the main contents of a movie and/or is not accurate due to the difficulty associated with precise information recall. We also confirm this from our preliminary experiments in which a number of volunteers are asked to watch some movie clips of diverse categories and point out the key information contained therein. By comparing their suggested information with a ground truth collected by a careful analysis of the clips, we discover that human-generated semantic labels lack consistency and present irregularities. Our experiments further show that this seemingly trivial task requires an intelligent video analysis to automatically extract the salient information from movies. This information can be utilized in a number of tasks including search optimization, scene-driven retrieval, object detection, translating movies to natural language, event detection, action recognition, behavior recognition, recommendation systems (to name a few).

The problem addressed in this paper refers to extracting the key information from a movie and summarizing it into a few key tags, representing the overall theme of the movie. The aim is not only to understand the high level semantics in each frame of the movie, but also to identify a compact set of the movie's representative topics. Retrieving this information further helps in movies classification, context-based search, efficient archiving and content-censorship (e.g, violence, sex and nudity in kids movies). Using traditional object detection approaches to understand individual frames of a movie and extracting the salient information will turn out to be highly unproductive as it will generate only the low level information (e.g., the objects in the frame) and not the underlying relationship among the objects and/or the overall context. At the same time, analyzing each movie frame will be inefficient due to the resulting redundancy of the extracted information.

For automatic analyses of movies, a Machine Learning (ML) based algorithm is required which can learn the representative features of movie scenes. Recent developments in ML have brought about a paradigm shift to analyze complex data with an unprecedented efficiency which, in some cases, even outperforms human accuracy. The general-purpose parallel computing offered by Graphical Processing Units (GPUs) has now brought the enormous computing power required for machine learning into a single computer, paving the way for efficient image/video analytics. That being said, using ML to learn the traditional hand-crafted features for our problem is inefficient due to the certain known issues pertaining to these features (scale- and rotation variance, formulation of required mathematical models, lack of generalization, performance degradation under varying conditions, etc).

This problem can be efficiently addressed with Deep Learning (DL) [6] which does not require engineered features to be learned. Instead, it learns representations of data by discovering intricate structure in datasets with multiple levels of abstraction. This motivates us to formulate our problem as a DL based classification problem. We construct a tag vocabulary and build an appropriate dataset. For training a classifier on the constructed vocabulary, we use transfer learning to modify and train the final layer of Inception-V3 [16] Convolution Neural Network (CNN) using Softmax classification. For analyzing a movie for tags extraction, we first extract the key frames of the movie and compute their CNN features to get corresponding predictions from the model's newly added/trained final layer. Subsequently, the predictions pertaining to all the frames are assigned scores and relative strengths based on their prediction probabilities and dominance in the movie. The tags having low relative strengths are filtered out to get a set of few key tags which best describes the overall theme and main contents of the movie. Although we lose motion information by analyzing only the key frames of a movie, our proposed technique still performs efficiently as the recent related work in this domain shows that motion features have little to no impact on such tasks [17][22][1].

The work presented in this paper has the following striking features: (i) our proposed approach works at a higher semantic level by understanding the overall context in the individual movie frames. The context represents the interaction of the objects in a scene and their overall meaning. The examples of context include romance, violence, fight, action, etc, (ii) this work is different from typical event or scene recognition tasks, where each item belongs to a single event or scene, (iii) our proposed technique also stands apart from most object recognition tasks, where the goal is to label everything visible in an image. This will produce thousands of labels for a movie without providing its thematic points, and (iv) this work does include, but it is not limited to, genre classification of movies. A movie typically has 2-3 genres which do not reveal other information in the movie (e.g., violence, nudity, sex, etc). Our carefully designed vocabulary adequately covers the main theme of a movie and is flexible to scalability.

The rest of the paper is organized as follows. Section 2 provides an overview of the related work in this domain. In Section 3, we briefly discuss the theoretical background of CNN and transfer learning. Section 4 gives a detailed insight into our movies tags extraction technique. Section 5 discusses the experimental setup and the evaluation results of the proposed technique. Section 6 concludes the paper.

2. Related Work

To the best of our knowledge, the problem of automatic movies tagging, as formulated in this paper, has still not been studied in the relevant literature. The existing related work is mainly focused on general video tagging on a limited scale. In [11], the authors use multi-label classification to classify the video semantic concepts and models correlations between them for annotating certain concepts in a video. The video annotation technique presented in [14] utilizes the redundancy among YouTube videos to find connection among videos and propagating tags among similar videos. The techniques presented in [13][7] utilize the contextual meta-data acquired from the sensors on smart phones to generate video tags. In [9], the proposed technique recognizes basic objects in images and videos of a digital camera and extracts the meta data including geographical and date/time information to generate tags.

With the advent of deep learning and its growing popularity, the research on video understanding has been directed to use deep networks to learn hierarchical feature representations. The major part of the research on video processing using deep learning is focused on translating videos to natural language [3][19][18][10], video question-answering systems [2][8], and video classification [5][24][23][20]. Translating videos to natural language and video question-answering are different tasks than video tagging as they require more complex architectures such as recurrent neural network [3] in conjunction with CNN to understand the spatio-temporal relationship between successive video frames. Video classification is closely related to video tagging, however, it is primarily focused on predicting the major category a video falls in, rather than extracting the key information from a video.

3. Convolution Neural Network (CNN) & Transfer Learning

CNNs, though similar to traditional neural networks, have much deeper architectures and are best suited to learn underlying patterns in complex data. In the last few years, there has been a growing interest in using CNN for image recognition, classification and other related tasks. A typical CNN architecture includes convolution, activation, pooling and classification layers. Convolution layer extracts image features using multiple filters. The activation layer introduces non-linearity in the learning process by limiting the output of the convolution layer in a certain range. Pooling layer downsamples the data to reduce its size and selects the prominent features. The classification layer is a fully connected layer which computes the final scores of each class. A CNN may have a number of convolution, activation and pooling layers. This deep architecture is useful for extracting and learning general and representative features without human intervention.

Training a CNN from scratch for a new task requires huge computing resources, long training time and a massive amount of training data. Recent research [21] shows that the features learned by a CNN are transferable from one training problem to another. Starting from the lower CNN layers, which extract generic and low level features, the specificity of features increases as we move to the higher CNN layers, making the final layer purely task specific. Therefore, the lower layers of a CNN trained for a large dataset contain learned weights for low level features which can be utilized to train the model for a new task. Using the pre-trained CNN model as a fixed feature extractor, its final layer can be modified and retrained for a new task. This is called transfer learning and we use it to modify and retrain the final layer of Inception-V3 CNN, pre-trained on a large dataset (ImageNet[1]), for the task of movies tags extraction. Although Inception-V3 has been trained for an entirely different problem, its features are effectively transferred to learn this new training problem.

4. Movies Tags Extraction

Following sections explain the various steps involved in our movies tags extraction technique.

4.1. Dataset

We construct a vocabulary of 50 movie tags and collect the dataset for each tag by extracting relevant frames from a number of movie trailers. The tag vocabulary, shown in Table 1, has 700 images corresponding to each tag. Note that the tags in the same color have similar features which makes this training problem harder than the classification problems in which the classes have little or no overlapping. We also frequently increase the vocabulary size as more tags are identified and the relevant dataset is collected.

4.2. Training

We remove the last layer of Inception-V3 pre-trained model and add a new layer for training it on our dataset. We freeze the rest of the layers and use the model as a feature extractor. We further add a dropout layer [15] as a penultimate layer to randomly drop the activations of 50% units during training in order to prevent the emergence of interdependencies among them and making the model more robust. We use a small learning parameter of 0.005 and larger training and validation batch sizes of 500 respectively to get more stable results. The dataset is partitioned as follows: 80% training images, 10% validation images and 10% test images. After applying dropout, the output of the penultimate layer for each input image is calculated as follows,

$$y_i = ReLU[\sum_j W_{i,j} x_j + b_i] \qquad (1)$$

Action	Bomb explosion	Car chase
Destruction	Sword fight	Vehicle crash
Violence	Abduction	Heist
Adventure	Animal	Beach/Sea
Climbing	Desert	Hiking
Forest	Valleys/Hills	Children
Family	Club/Bar	Dance
Music	Wedding	College/Univ.
Hospital	Drinking	Food
Smoking	Exercise	Sports
Swimming	Glamor/Fashion	Nudity
Romance	Sex	Horror
Monster	Murder	Lab Experiment
Sci-fi	Super hero	Technology
Robot	Military	Police
Prison	War	Weapon
Animation	Drama	

Table 1: Movies tag vocabulary

where W_i represents the neurons weights and b_i is the bias for i^{th} tag. The notation x_j represents the j^{th} pixel of the input image. $ReLU$ (Rectified Linear Unit) activation function is used to introduce nonlinearity in the training, so that the network can generalize well for the unseen data.
For converting the output of the penultimate layer into a probability distribution, we use Softmax classification [4] to predict the probabilities of all the tags by the following rule,

$$p_i = \frac{e^{y_i}}{\sum_j^{50} e^{y_j}} \qquad (2)$$

where p_i represents the probability of i^{th} tag in the set of 50 tags. For calculating the error between the estimated distribution p and the true distribution q, we use a cross-entropy error estimate [4] $E(p, q)$ as follows.

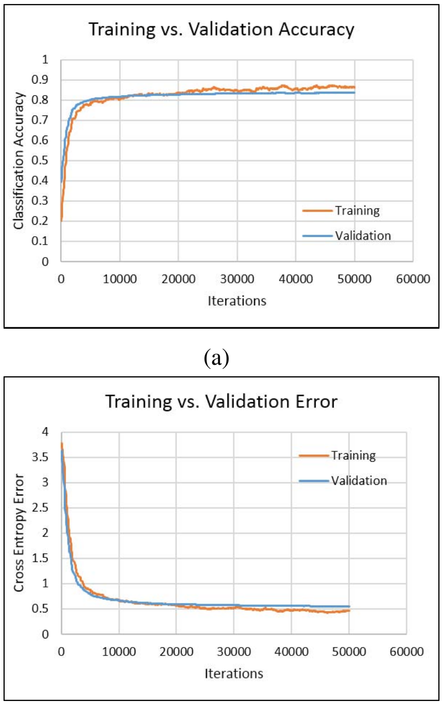
$$E(p, q) = -\sum_x^{50} q(x) \log p(x) \qquad (3)$$

The Softmax classifier minimizes the error between the estimated distribution and the true distribution. We run the training for 50,000 iterations (500 epochs). Figure 1a and Figure 1b show the smoothed curves of training/validation accuracy and cross-entropy error during the training. It is evident that the training/validation margin in both the cases is greatly reduced and the model generalizes well due to the dropout layer and the right selection of training parameters. The overall test accuracy of the model is 85%.

4.3. Tags Extraction for Individual Movie Frames

We test the trained model on individual movie frames for tags extraction. We utilize the overlapping among the

[1] http://www.image-net.org/



(a)

(b)

Figure 1: Smoothed curves of (a) training/validation accuracy, and (b) training/validation cross-entropy error

tags features to consider higher number of predictions in the estimated distribution than only the topmost prediction which represents the most dominating tag in a movie frame. While the topmost tag has the highest probability in the estimated distribution, the other lower probability tags may indicate other relevant information in the movie frame. Table 2 shows the predicted tags for some movies frames. Note that the first tag in each set of the predicted tags represents the topmost tag with the highest probability. The rest of the tags, though predicted with lower probabilities, still reveal the relevant information contained in the movie frames.

4.4. Tags Extraction for Movies

For movie tags extraction, we first extract the key frames of the movie by identifying the boundaries of successive shots. A shot is a series of frames that runs for an uninterrupted period of time. We extract the movie frames at 1 Frame Per Second (FPS) and find the shots boundaries by comparing the HSV histogram of successive frames. If the difference of normalized histogram values of the two successive frames is found to be greater than a certain threshold, it is marked as a shot boundary and we select the median frame of the shot as a key frame. Increasing the threshold makes the shot detection more lenient to change and results in smaller number of key frame, whereas decreasing the threshold makes it more conservative and results in higher number of key frames. We select a threshold value 0.5 which works reasonably well.

Frames	Tags
	Military, action, weapon, war
	Violence, destruction, bomb explosion, action, vehicle crash
	Sex, nudity, romance, Glamor/fashion
	Hiking, adventure, forest, valleys/hills, climbing
	Sci-fi, super hero, robot, action
	Violence, sci-fi, action, horror

Table 2: Predicted tags for individual movies frames

Apart from detecting the shot boundaries and selecting key frames, we also calculate the entropy of each key frame to check if it contains reasonable amount of information to run the tags extraction inference. We convert the key frame to luminance/chrominance color space and calculate the entropy of each channel by the following rule [12],

$$H = -\sum_{i}^{n} p(x_i) \log_2 p(x_i) \tag{4}$$

where $p(x_i)$ represents the probability of a pixel x_i to assume a certain value. The cumulative entropy H serves as a measure to estimate the information contained in a key frame. We only select those key frames whose cumulative entropy is greater than a certain threshold ($H > 0.20$).

After extracting the representative frames of a movie and eliminating redundancy, we run inference on each key frame to get top 3 tags. Subsequently, we find the strength of each tag by the following rule,

$$W_i = \frac{n_i}{N} \sum_{j=0}^{N} P_{ij} \tag{5}$$

where W_i represents the strength of i^{th} tag, n_i denotes the number of occurrences of i^{th} tag, N is the total number of extracted tags, and p_{ij} is the probability of j^{th} occurrence of i^{th} tag.

We further normalize the tags strength in $[0, 1]$ to calculate

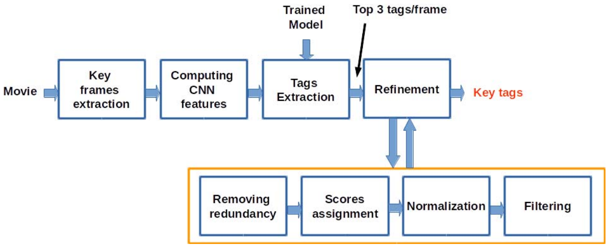


Figure 2: Overall technique of movies tags extraction

Hardware/Software	Specifications
CPU	Intel Xeon(R) E5430, 2.66GHz x 8
RAM	8GB
GPU	GeForce GTX 1050 Ti, 768 cores, 4GB GDDR5
DL framework	Tensorflow 0.12, compiled with GPU support
Operating System	Ubuntu 16.04 (64-bit)
Programming languages	Python 2.7, OpenCV 3.0

Table 3: Experimental setup

the relative strength R_i of each tag by the following rule,

$$R_i = \frac{W_i - W_{min}}{W_{max} - W_{min}} \tag{6}$$

where W_{max} and W_{min} represent the maximum and minimum tags strengths in the set of all the extracted tags. We filter out the tags having strengths less than a certain threshold to get a fewer key tags which best describe the movie. The overall approach is depicted in Figure 2.

5. Experimental Setup and Results

The hardware/software details of our experimental setup are specified in Table 3.

With this experimental setup, the overall average time to process a frame is 50 milliseconds (20 FPS) for a 720p movie. Due to the unavailability of a ground truth, we opt to test the performance of the proposed technique with a subjective evaluation. We perform three different experiments in Fraunhofer IIS digital cinema with different sets of 10 volunteers each. The evaluation is performed on a number of movie trailers of diverse categories, as a movie trailer is a precise representation of a movie's contents. This is also helpful to complete the experiments in a reasonable time.

In the first experiment, the participants are shown a number of movie trailers. At the end of each trailer, the set of the extracted key tags is revealed to the participants and they are asked to rate it between 0 to 10 based on its relevancy and completeness. At the end of the experiment, we calculate the Mean Opinion Score (MOS) from the feedback received from the participants which is found to be 84.3%.

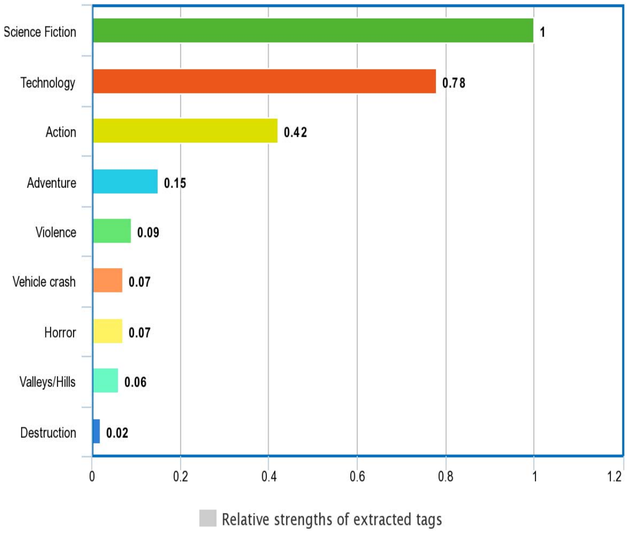


Figure 3: Extracted tags and their relative strengths for the movie trailer *Alien Covenant (2017)*[2]

In the second experiment conducted with a different set of participants, we ask the participants to rate the extracted tags of each movie trailer not only taking into account their relevancy, but also their relative strengths presented to them in the form of a visual chart as depicted in Figure 3. The MOS for this experiment is 78%.

In the third experiment, the participants are provided the tag vocabulary and are asked to suggest the relevant tags from the vocabulary after watching each trailer. Using the participants' feedback as ground truth in this experiment, we calculate the mean average precision P, mean average recall R and F1-score by the following formulas,

$$P = \frac{1}{(MN)^2} \sum_i^N \sum_j^M \frac{T_P(i,j)}{T_P(i,j) + F_P(i,j)} \tag{7a}$$

$$R = \frac{1}{(MN)^2} \sum_i^N \sum_j^M \frac{T_P(i,j)}{T_P(i,j) + F_N(i,j)} \tag{7b}$$

$$F1 = 2(\frac{P \times R}{P + R}) \tag{7c}$$

where $T_P(i,j)$, $F_P(i,j)$ and $F_N(i,j)$ represent the number of true positive, false positive and false negative, respectively, for the i^{th} movie trailer and j^{th} participant. The notations N and M represent the number of participants and the number of movie trailers, respectively. The mean average precision and recall of this experiment are 76% and 74.22%, respectively, which give a F1-score of 0.75.

[2] https://www.youtube.com/watch?v=H0VW6sg50Pk

6. Conclusion

In this paper, we proposed a deep learning based movies tags extraction technique. The proposed technique is primarily focused on retrieving higher level semantics from a movie and representing it in a set of few key tags which best describes the movie. The proposed technique is flexible to increase the size of tag vocabulary while maintaining the performance. The subjective evaluation results of our proposed technique demonstrate its efficacy to retrieve the key tags of a movie with a good accuracy.

In future, we aim to extend the tag vocabulary for more tags and utilize the extracted tags for a number of tasks, including query based scene retrieval, movies classification for efficient archiving and search, and extracting the most relevant key frames for movie summarization (to name a few).

Acknowledgement

This work was carried out during the fellowship tenure of ERCIM *Alain Bensoussan* programme.

References

[1] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, and G. Toderici. Youtube-8m: A large-scale video classification benchmark. *CoRR*, abs/1609.08675, 2016.

[2] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433, 2015.

[3] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[4] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT Press, 2016.

[5] A. Karpathy, G. Toderici, S. Shetty, and T. Leung. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

[6] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[7] X. Liu, M. Corner, and P. Shenoy. Seva: sensor-enhanced video annotation. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 618–627, 2005.

[8] M. Malinowski, M. Rohrbach, and M. Fritz. Ask your neurons: A neural-based approach to answering questions about images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1–9, 2015.

[9] J. Miranda-Steiner. Automatic tag generation based on image content, 2014. EP Patent App. EP20,120,850,387.

[10] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui. Jointly modeling embedding and translation to bridge video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4594–4602, 2016.

[11] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, T. Mei, and H.-J. Zhang. Correlative multi-label video annotation. In *Proceedings of the 15th ACM international conference on Multimedia*, pages 17–26, 2007.

[12] C. E. Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001.

[13] Z. Shen, S. Arslan Ay, S. H. Kim, and R. Zimmermann. Automatic tag generation and ranking for sensor-rich outdoor videos. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 93–102. ACM, 2011.

[14] S. Siersdorfer, J. San Pedro, and M. Sanderson. Automatic video tagging using content redundancy. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 395–402, 2009.

[15] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[16] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567, 2015.

[17] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri. C3d: generic features for video analysis. *CoRR, abs/1412.0767*, 2:7, 2014.

[18] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4534–4542, 2015.

[19] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729*, 2014.

[20] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 461–470, 2015.

[21] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[22] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *The IEEE Conference on Computer Vision and Pattern Recognition*, June 2015.

[23] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702, 2015.

[24] S. Zha, F. Luisier, W. Andrews, N. Srivastava, and R. Salakhutdinov. Exploiting image-trained cnn architectures for unconstrained video classification. *arXiv preprint arXiv:1503.04144*, 2015.