Unsupervised Learning of Procedures from Tutorial Videos

Amlan Sengupta 312217104008
Aparna K 312217104012
Arunima S 312217104016

BE CSE, Semester 7

Dr. Venkata Vara Prasad
Supervisor

Project Review: 1 (31 October 2020)
Department of Computer Science and Engineering
SSN College of Engineering

---

1 Abstract

Online educational platforms and MOOCs (Massive Open Online Courses) have made it so that learning can happen from anywhere across the globe. In addition to theoretical lectures, there are hands-on tutorials being distributed on these platforms as well. While this is extremely beneficial, videos are not accessible by everyone, due to time constraints and lower bandwidths. Textual step-by-step instructions will serve as a good alternative, being less time-consuming to follow than videos, and also requiring lesser bandwidth. In this project, we aim to create a system that extracts and presents a step-by-step tutorial from a tutorial video, which makes it much easier to follow as per the user's convenience

2 Introduction

The video delivery platforms of today host hordes of information in the form of tutorials. These often have a walk through of a concept or example problems. The idea behind this is that the learner can follow along with the video, and hence obtain a holistic learning experience.

Some examples of tutorial videos are: a walk through of math problems, cooking a dish from scratch, an instructional video of certain exercises and so on. These are all instances where the concept in question is explained step by step.

While such videos are extremely useful and can provide a better understanding of a concept, they are not always accessible to the average learner. This may be due to several reasons: no access to the Internet or not having enough bandwidth to stream videos, time constraints that prevent the learner from watching the entire video and so on. Having the content of the video as an easy-to-follow textual algorithm will be of utmost benefit to the learner in these circumstances.

To accomplish this, we aim to use Image Processing and Video Instance Embedding (VIE) that will extract "steps" from a tutorial video and compile it together. The closed captioning of the tutorial videos can also be used wherever applicable. The final goal is to create an efficient, reliable system that can translate a video into a textual set of step-by-step instructions.

3 Literature survey

The authors propose the method of temporal clustering and latent variable modeling, and develop a general pipeline for procedure extraction. Temporal clustering consists of aggregating similar frames or images and removing duplicates from the pipeline. They further evaluate this method and propose improved metrics which are temporal purity and temporal completeness [1].

Another approach called Video Instance Embedding (VIE) framework, which trains deep nonlinear embedding on video sequence inputs is introduced. The authors feel that a two pathway model with both static and dynamic processing pathways is optimal as it provides analyses indicating how the model works [2].

The authors represent the video sequence as a collection of spatial-temporal words by extracting space-time interest points. The probabilistic Latent Semantic Analysis (pLSA) model is the one used, this uses an algorithm that automatically learns the probability distributions of the spatial-temporal words and intermediate topics corresponding to human action categories [3].

The authors use a scalable clustering approach for the unsupervised learning of convolution nets. It iterates between clustering with k-means that features Deep Clustering for Unsupervised Learning of Visual Features produced by the convolution net and updating its weights by predicting the cluster assignments as pseudo-labels in a discriminative loss [4].

The author has used spatio-temporal salient feature detection for motion detection. A salient feature is defined by its center point referred to as spatio-temporal interest/key point. Spatial and temporal scale events require the detection of salient features at multiple scales for which a multi-resolution spatio-temporal representation of the video is required. After this a time-causal multi-resolution representation of a video signal is derived which is then utilized for salient feature detection [5].

When exploring the approach of clustering, there are two methods: textual clustering and video clustering. Textual clustering involves inputting sequences of direct object relations, aligning multiple sequences and discovering steps depending on the value of K (the number of clusters chosen). Video clustering involves localizing the actions shown with text constraints. The assumptions made for both approaches are: each task is composed of an order sequence of steps, and people do what they say roughly when they say it [6].

The authors apply sparse coding to represent natural signals such as images using only a few non-zero coefficients, i.e. as a linear decomposition using a few atoms of a suitable dictionary. In action recognition, the principles of sparse coding are used to aggregate local spatio-temporal features [7].

DPC is used to predict a slowly varying semantic representation based on the recent past. A video clip is partitioned into multiple non-overlapping blocks, with each block containing an equal number of frames. An aggregation function is then applied to the blocks and this is used by the author for creating a context representation which helps identify the actions [8].

The concept of using CompILE, a framework for learning reusable, variable-length segments of hierarchically-structured behavior from demonstration data is also introduced. CompILE uses a novel unsupervised, fully-differentiable sequence segmentation module to learn latent encodings of sequential data that can be re-composed and executed to perform new tasks. Once trained, the model generalizes to sequences of longer length and from environment instances not seen during training. CompILE has been tested in 2D multi-task environments successfully, and will be applicable to our problem statement [9].

4 Proposed system

The proposed system utilises deep learning. It consists of the following steps:

1. The tutorial or demonstration video is taken as input.
2. Frames are extracted from the video at regular intervals.
3. The frames demonstrating similar procedures are grouped together which is known as temporal clustering which also removes the duplicates.
4. Action recognition is performed for each set of frames using deep learning methods.
5. For specific tutorial videos, text recognition is done to identify what's written.
6. The procedural instructions present in the entire video are given as output.

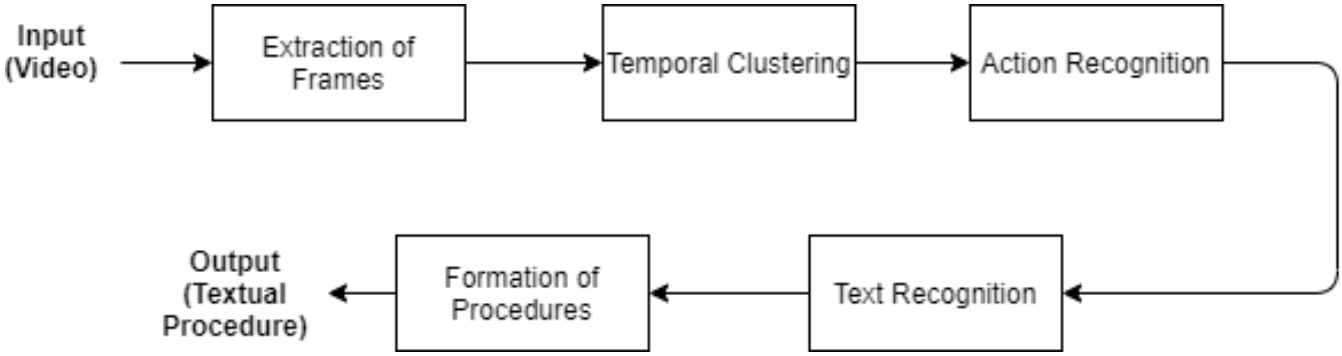

Figure 1: Modules

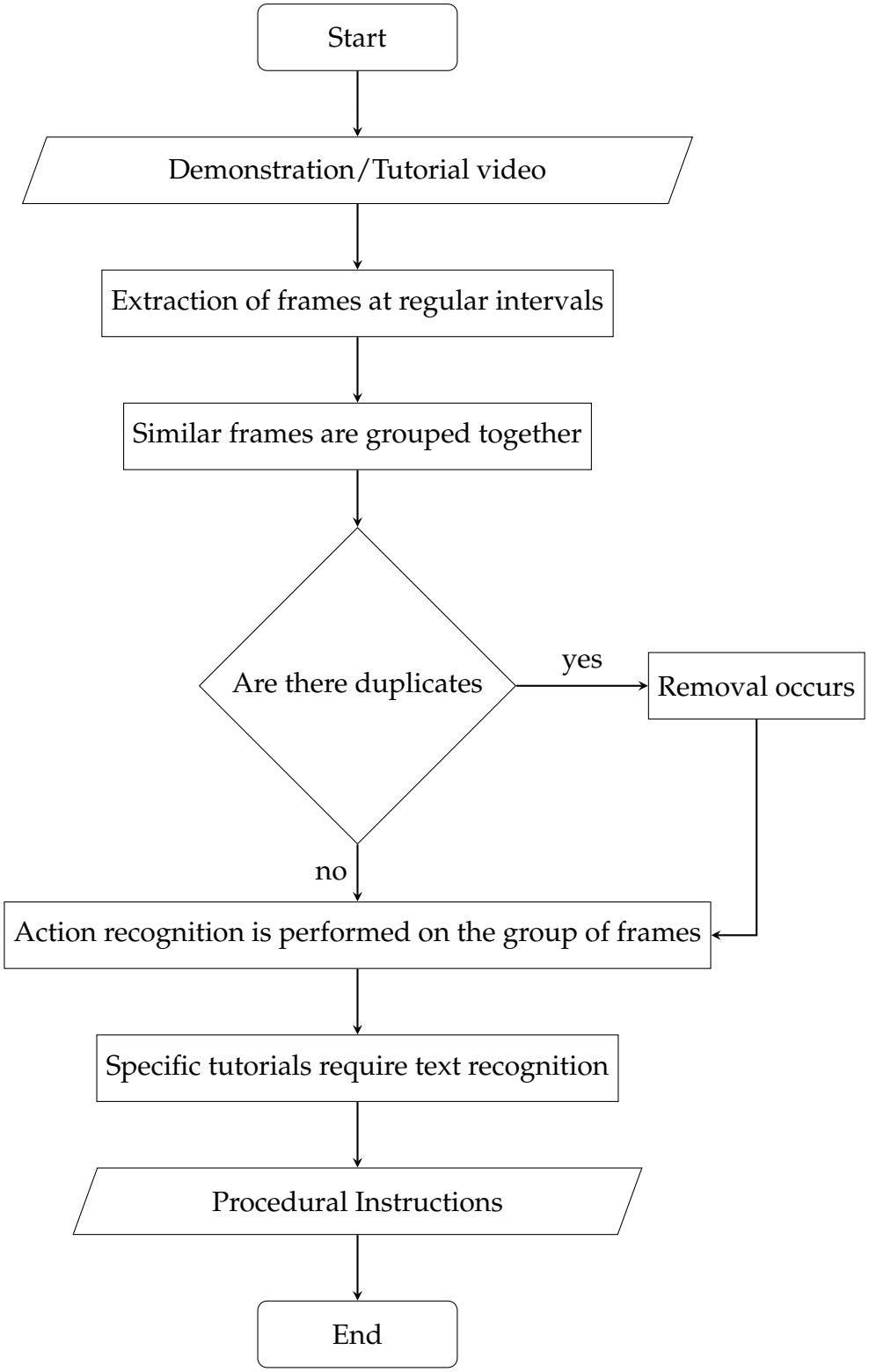

Figure 2: Proposed System Flowchart

5 Modules Split-up

The proposed system consists of 5 modules.

1. Extraction of frames: The demonstration video that is taken has to be split into various images. In-built functions from packages like Skimage and OpenCV which include VideoCapture() and imwrite() are used to take the video into the data frames and split into separate images that are stored.

2. Temporal Clustering: The stored images are then segregated into various groups based on the action denoted by them. Temporal Clustering (TC) refers to the factorization of multiple time series into a set of non-overlapping segments that belong to k temporal clusters. These clusters are used to classify the images into the groups based on their timing and similarity. The duplicates are removed from their respective groups.

3. Action Recognition : The temporal clusters that are formed are processed using Video Instance Embedding (VIE) framework which shows how the idea of deep unsupervised embeddings can be used to learn features from videos.VIE learns powerful representations for transfer learning to action recognition in the videos dataset, as well as for single-frame object classification in the frames dataset.

4. Text Recognition : In cases where there is text present in the frames, we use the OCR package available in python to recognize handwritten and printed text from video frames. These are further added to the procedure for the video.

5. Formation of Procedures: Based on the timing sequence , the instructions for the consecutive frames are put together to form a procedure for the entire video.

References

[1] K. Goel and E. Brunskill, "Unsupervised learning of procedures from demonstration videos," 2018.

[2] C. Zhuang, T. She, A. Andonian, M. S. Mark, and D. Yamins, "Unsupervised learning from video with deep neural embeddings," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9563–9572, 2020.

[3] A. H. Shabani, D. A. Clausi, and J. S. Zelek, "Improved spatio-temporal salient feature detection for action recognition.," in *BMVC*, pp. 1–12, Citeseer, 2011.

[4] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 132–149, 2018.

[5] J. C. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words," *International journal of computer vision*, vol. 79, no. 3, pp. 299–318, 2008.

[6] J.-B. Alayrac, P. Bojanowski, N. Agrawal, J. Sivic, I. Laptev, and S. Lacoste-Julien, "Unsupervised learning from narrated instruction videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4575–4583, 2016.

[7] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *Image and vision computing*, vol. 60, pp. 4–21, 2017.

[8] T. Han, W. Xie, and A. Zisserman, "Video representation learning by dense predictive coding," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 0–0, 2019.

[9] T. Kipf, Y. Li, H. Dai, V. Zambaldi, A. Sanchez-Gonzalez, E. Grefenstette, P. Kohli, and P. Battaglia, "Compile: Compositional imitation learning and execution," in *International Conference on Machine Learning*, pp. 3418–3428, PMLR, 2019.