Waiting for a Sign: An Unsupervised Pipeline for Sign Language Recognition

Dan Sakaguchi
Stanford University
dsakaguc@stanford.edu

Jacob Waggoner
Stanford University
jacobw1@stanford.edu

Abstract

Our research presents an unsupervised method of retroactively labeling gestures (taken from American Sign Language) in an unlabeled video dataset. Given an estimate for the number of gestures ("temporal motifs") contained in the video, our program attempts to categorize the gestures, as well as estimate their starting times within the video. The model uses low-level graphical features extracted from the video to determine significant information (repetition of full or partial gestures) without temporal dependence, through probabilistic latent semantic analysis (pLSA). pLSA additionally acts to reduce the dimensionality of the large dataset . Gestures and their starting times are then found by probabilistic latent sequential motif (pLSM) analysis on the compressed information determined by the pLSA. Once the gestures are categorized and their starting times are determined by the pLSM analysis, a simple retroactive labeling of each found gesture results in an accurate labeling of the entire video data set.

Introduction

In the field of gesture identification, there exists much work on the recognition and classification of sign language, American Sign Language (ASL), in particular. Several successful models have been proposed for identifying static signs (e.g., signs of the alphabet) [5,6,7]. Despite this, there is relatively little work focused on the recognition and classification of dynamic signs (signs for which the motion is not separable from the meaning). To date, one of the only successful studies produced 65% accuracy after 10 hours of exposure to dynamic, supervised signs [9]. Very few, if any, studies have approached the problem of dynamic sign recognition with an unsupervised model. Initially, we planned to use a supervised model using labeled sign data, however, it was not readily available. This, in combination with the notable absence in the field on unsupervised approaches, motivated our interest in developing a means of identifying repeated signs without supervision in order to learn what a sign 'looks like.' In theory, then, given an internal representation of what a sign looks like, our algorithm would 1) allow the video to be accurately labeled at the substantially reduced cost of labeling each of the representations, and 2) given a metric and probability threshold, allow identification of a new sign as either a particular sign in the built-up vocabulary, or a new sign

In a generalized context, this problem has been referred to as temporal "motif discovery' [8]. Most recently, Emonet et al developed a method for extracting temporal motifs from video of a traffic intersection (e.g., identifying a right turn, a left turn, a crowd crossing the sidewalk, etc.) [2]. Because of its success, as well as the analogy between finding temporal motifs in traffic video and in video of signs, we sought to apply their model to our problem by adapting it to fit the setting of sign language. Our treatment of the problem is thus broken down into four primary

components: feature extraction, feature dimensionality reduction, motif extraction, and validation.

Model Overview

The two primary algorithms utilized in our study were probabilistic latent semantic analysis (pLSA) and probabilistic Latent Sequential Motif (pLSM) analysis. The first was used to reduce the dimensionality of the data (low-level graphical features) passed to the second, while the second incorporates temporal information to determine temporal motifs. Both algorithms operate primarily on data in the form of a "bag of words," a matrix storing counts of "words" in "documents" [1]. These counts are then used to generate probability distributions of latent variables (the number of which is specified by a parameter) over the data with the estimation-maximization algorithm. In the pLSA, the words are the low-level graphic features described below, the documents are the spatio-temporal boxes containing those features in bins, the latent variables are components of signs (segments of motion that are repeated throughout the video, but not necessarily whole signs), and the generated distributions are P(w|z) and P(z|d) where w refers to the words, z to the latent variables, and d to the documents. In the pLSM analysis, the words are frequency counts estimated by the pLSA, the documents are video segments, the latent variables are signs (temporal motifs), and the generated distributions are P(z|d), P(ts|z,d), P(w|z), and P(tr|w,z), where w, z, and d are as before, ts is the starting time of the sign within a document, and tr is the relative time within a sign (ie, tr = 0 with respect to latent variable z refers to the start time ts, of that variable z). These processes are described in greater detail below.

Feature Extraction

In [1,2], Emonet et al proposed the use of low-level graphical features (ie, video frame pixel by pixel) such as optical flow and spatio-temporal location (coordinate (w, h, f) where w and h represent a pixel position in a frame, and f is a frame) to serve as the words for the pLSA [2]. We created documents from these features by dividing the video into 10 x 10 x 6 boxes (with 2 frame of overlap between documents) over which we binned the flows into four motion categories: left, right, up and down. A manually determined threshold was used to eliminate pixels without significant motion. To these bins, we elected to add additional features derived from the edges of our images. Similarly to optical flow, we binned the edges into four categories: horizontal, vertical, and two diagonal categories (45 degrees off of the horizontal and vertical, respectively). These were appended as additional features after testing on artificial data, which indicated that the pLSA might require improved features.

Dimensionality Reduction

The features above contain an enormous amount of data. For a five-minute video of signs, the main video file is a matrix of approximate size 540 x 960 x 3 x 8000. Given our resources, processing matrices of this size was not realistic or time efficient because of the computational expense. Thus, we analyzed the features from above using a pLSA in order to reduce the amount of data that the motif-finding algorithm (pLSM) needed to process. This analysis uses EM maximization to maximize the joint word-document log-likelihood probability distribution in order to generate locally optimal multinomial distributions for word-document co-occurrences. We additionally attempted to perform this reduction step with latent dirichlet analysis, but found that it was too computationally expensive for the

quality of video (540 x 960) and frame sampling rate (10 fps) we desired.

Model

The model that we use for the temporal motif-finding, pLSM analysis, is a generative one. It is summarized as follows [2]:

- Draw a temporal document d with probability P(d)
- Draw a latent motif z from P(z|d)
- Draw the starting time ts from P(ts|z,d)
- Draw a word and relative time pair (w,tr) from P(w,tr|z)
- Set the absolute time in the document to the starting time plus the relative time

This process is also illustrated in Figure 1 below [2]. Generally, as with pLSA, it uses EM maximization to produce locally optimal multinomial distributions on the co-occurrence of words (sign segments found from pLSA) and documents (video segments). Specifically, Tz is a specified maximum sign length (in frames), ta is a time (frame) occurrence, ts represents a starting time (frame) of a sign, and tr is the relative time given a sign (number of frames from ts). The algorithm itself is outlined in Figure 2 [1].

Figure 1 – taken from [2]

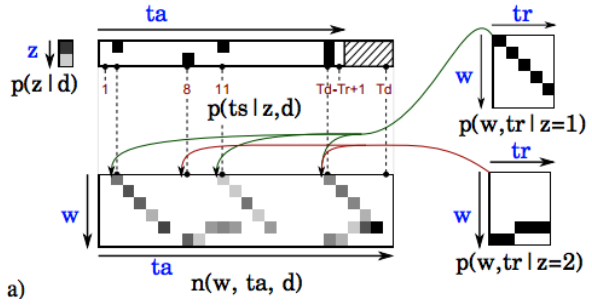


Figure 2 – Equations for pLSM EM-maximization

$$E-Step: P(z,ts \mid w,ta,d) = \frac{P(w,ta,d,z,ts)}{P(w,ta,d)} \; where, P(w,ta,d) = \sum_{z=1}^{Nz}\sum_{ts=1}^{Tds} P(w,ta,d,z,ts)$$

$$M-Step: P(z \mid d) \propto \sum_{ts=1}^{Tds}\sum_{tr=0}^{Tz-1}\sum_{w=1}^{Nw} n(w,ts+tr,d)P(z,ts \mid w,ts+tr,d)$$

$$P(ts \mid z,d) \propto \sum_{w=1}^{Nw}\sum_{tr=0}^{Tz-1} n(w,ts+tr,d)P(z,ts \mid w,ts+tr,d)$$

$$P(w,tr \mid z) \propto \sum_{d=1}^{D}\sum_{ts=1}^{Tds} n(w,ts+tr,d)P(z,ts \mid w,ts+tr,d)$$

$$n(d,ta,w) = \frac{1}{\sum_{\omega \in Wy} n(dta,\omega)} \sum_{\omega} n(dta,\omega)p(\omega \mid y)$$

Synthetic Data

Figure 3 – Synthetic data in a., a. with noise added in b

a. 

b. 

We first demonstrate the temporal motif extraction algorithm on the artificial count data in Figure 3b. Were this data not synthetic, it would represent the counts matrix calculated by equation 5 in Figure 2. Figure 3a shows the data before Gaussian noise (mean 0, standard deviation 1) was added. It contains three different motifs, as shown in Figure 4a-c, comprised of five words (motif components), of different lengths (3, 4, and 5) at random positions in the 200 frame document. The results of pLSM analysis with a maximum motif length of 6 and an estimated number of motifs, 3, are shown in Figures 4-5. Figure 4 d-e represents the estimated motifs, and Figure 5 represents the estimated starting times of the motifs. As is clear from comparing Figures 4a-c and 4d-e, the model yielded estimated motifs that are identical to the synthesized motifs. Moreover, in visually comparing Figure 5a with Figures 5b-d (the ith row in 5a corresponds to the ith image beneath 5a. The rows represent motifs, the columns represent starting times. Also note that some scaling differences arose in formatting the image, and that the times do, in fact, correctly line up), it is clear that the model is most strongly weighted

towards the correct motifs at the correct times. This serves to demonstrate the robustness of the model given noisy data.

Figure 4 – True and learned synthetic motifs

a-c. 

d-f. 

Figure 5 – True and learned motif starting times for the synthetic data

a. 

b. 

c . 

d. 

Real Data Collection

Originally, we intended to use Microsoft Kinect's depth sensor in conjunction with RGB video to collect the data for this study. After observing that the depth information did not have the granularity to substantially contribute to the model, we elected to use only RGB video and extract features using optical flow and edge detection, as described previously. The preliminary results presented here were recorded with the Kinect camera (5 fps). In the video, we used the signs for "help," "want," "hello," and "please," intentionally selecting a diverse vocabulary to strain the model. Although we recorded the data ourselves, we consulted a professor of Sign Language at Stanford University for her expertise in signing prior to collecting the data. Figure 6 shows a still image of the sign for "help."

Figure 6 – Image of the sign for "help"

Real Data

After extracting features from the video of signs, one result of which is shown in Figure 8 from the video (original image, optical flow overlay, and edge overlay), we analyzed the videos with pLSA. Figure 8 shows one component of a sign (the sign "help") found by the pLSA on the second video. Generally, the result of feature extraction and pLSA were qualitatively very similar to these images, in that they were consistently correctly and identifiably associated with a sign.

Results

We only qualitatively compare the estimated starting times for the found motifs and the true starting times. These results are shown in Figure 7 with a. corresponding to the true times, and b. corresponding to the estimated times. As can be seen by comparison of the images, where row represents a sign, and column represents a starting time, our pipeline was able to identify, with 100% accuracy, the signs for "want," "hello," and "please," finding one false positive (bottom left corner) on the sign for want. The sign for "help" was correctly identified 80% of the time. Also note the weak, but significant, probabilities in the first row of each figure. With the exception of two starting times predicted for the sign for "please," all of the estimated start times align with the true start times. This can perhaps be explained by the fact that the sign for "please" is

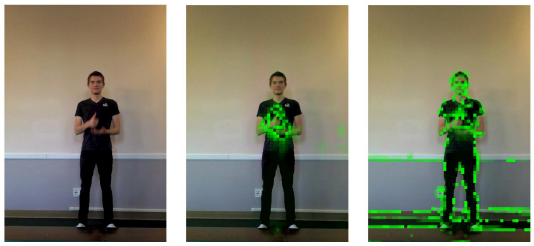
approximately 1.5 times the length of the others, and so may vary more.

Figure 7 – True and learned motif starting times for the real data

a.

b.

Figure 8 – Original image, optical flow and line features for a learned pLSA topic



Discussion

In the preliminary video, comparison of the learned and true labels suggests that our procedure was effective in determining both the identity of the gestures and their starting times throughout the video. Examination of the learned patterns shows that these motifs do indeed correspond with the intended ASL signs with high accuracy. Despite these promising indications though, the algorithm appears to have assigned unacceptably low probabilities to several instances of the 'help' sign, which could have resulted from the low fps of the collected video or the low dynamic content of the sign itself. Future work will take advantage of the non-overlapping nature of ASL signs to increase sign fidelity.

References

[1] Varadarajan, J. (n.d.). Retrieved from http://publications.idiap.ch/downloads/papers/2012/Varadarajan_THESIS_2012.pdf

[2] Varadarajan, J., Emonet, R., & Odobez, J.M. (2010). Probabilistic latent sequential motifs: Discovering temporal activity patterns in video scenes. Idiap Research Institute, Retrieved from http://publications.idiap.ch/downloads.papers/2010/Varadarajan_BMVC2010_2010.pdf

[3] Kharbat, M. (2009). Horn-Schunck Optical Flow Method. Mathworks File Exchange, Retrieved from http://www.mathworks.com/matlabcentral.fileexchange.22756-horn-schunck-optical-flow-method/content/HS.m

[4] Hofman, T., (2001). Probabilistic Latent Semantic Analysis (pLSA). Oxford Visual Geometry Group, Retrieved from http://www.robots.ox.ac.uk/~vgg/software/

[5] Farrell, J., Atwood, J., & Atwood, J. (2012). *American sign language recognition system* . Informally published manuscript, Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, .

[6] Sole, M., & Tsoeu, M. (2011, 09). *Sign language recognition using the extreme learning machine* . Ieee africon 2011 - the falls resort and conference centre, Zambia.

[7] Rehg, J., Essa, I., Hamilton, H., Starner, T., & Yin, P. (2009). *Learning the basic units in american sign language using discriminative segmental feature selection*. Informally published manuscript, School of Interactive Computing, Georgia Institute of Technology, Georgia Institute of Technology, Atlanta, GA.

[8] Sarrahfzadeh, M., Amini, N., & Vahdatpour, A. (2009).*Toward unsupervised activity discovery using multi-dimensional motif detection in time series*. Informally published manuscript, Department of Computer Science University of California, Los Angeles, University of California, Los Angeles, Los Angeles, CA.

[9] Barras, C. (2009, 07 08). Computer learns sign language by watching tv. *New Scientist*. Retrieved from http://www.newscientist.com/article/dn17431-computer-learns-sign-language-by-watching-tv.html