Continuous Gesture Recognition using a Sparse Bayesian Classifier

Shu-Fai Wong and Roberto Cipolla
Department of Engineering, University of Cambridge
{sfw26,cipolla}@eng.cam.ac.uk

Abstract

An approach to recognise and segment 9 elementary gestures from a video input is proposed and it can be applied to continuous sign recognition. An isolated gesture is recognised by first converting a portion of video into a motion gradient orientation image and then classifying it into one of the 9 gestures by a sparse Bayesian classifier. The portion of video used is decided by using a sampling technique based on CONDENSATION framework. By doing so, gestures can be segmented from the video in a probabilistic manner. Experiments show that the proposed method can achieve accuracy around 90% in both isolated and continuous gesture recognition without using special equipment such as glove devices and the system can run in real-time.

1 Introduction

Sign language systems have structured sets of gestures and they can provide a test bed for gesture recognition algorithms. Research in sign recognition is thus useful in developing gesture-based human-computer interfaces.

Sign recognition is a complex problem, which requires a divide-and-conquer approach. Complex sign recognition can be considered as recognition of a *sequence* of *elementary gestures*. It is, however, usually difficult to *segment* elementary gestures from raw images (i.e. gesture spotting problem). Apart from this problem, large inter- and intra-personal variations in gesture result in poor performance of *isolated gesture recognition*. Several attempts have been made to recognise continuous sign or gesture using Hidden Markov Models (HMMs), but they usually requires accurate tracking and complicated classification models. Hence, the applicability of the system is often compromised.

In this paper, an approach to segment and recognise 9 elementary gestures from an image sequence is proposed. Isolated gesture recognition is done by exploiting motion gradient orientation (MGO) images [3] to form motion features and using a sparse Bayesian classifier [9] to map the features into their corresponding gesture classes. Gesture segmentation is done by evaluating each hypothesis on location of gesture boundary under CONDENSATION framework [5]. This paper presents two main contributions. Firstly, the classifier for isolated gesture recognition maintains a *sparse* model, which facilitates an efficient use of computational resources and a *real-time* performance especially under the framework involving multiple hypotheses. Secondly, this approach provides a *probabilistic* solution (i.e. embedding Bayesian classifiers in CONDENSATION framework) to continuous gesture recognition and therefore ensures *reliable* performance.

Previous work: Sign recognition problem was first tackled by borrowing techniques from speech recognition, e.g. Hidden Markov Models (HMMs). HMMs (e.g. [8]) were applied widely to sign recognition in the past decade. Afterwards, HMMs were modified to tackle continuous sign recognition (e.g. [6]). Recently, the use of HMMs has been criticised for its requirement of large training sets (e.g. [2]) and its inflexible classification model (e.g. [4]).

A few suggestions have been proposed to avoid the use of HMMs. Derpanis et al. [4] suggested recognising elementary gestures or primitive movements, which are the most primitive components of American sign language, by trajectory matching. They also proposed converting continuous sign recognition problem into several elementary gesture recognition problems, although they haven't described explicitly how to do it. Bowden et al. [2] adopted a similar approach where motion data was first transformed into a linguistic feature vector (i.e. signature of an elementary gesture) and explicitly tackled the continuous sign recognition problem by using a bank of Markov chains. Their approach, however, inherits some weaknesses of Markov models such as inflexible classification model. In order to avoid the use of Markov models, Shan et al. [7] suggested the use of a trajectory-based feature and a tree-based classifier to recognise 7 elementary gestures. Their work, however, relies heavily on an accurate tracking result and haven't tackled the continuous gesture recognition problem explicitly. This paper extends the above work to perform continuous gesture recognition explicitly without using complicated classification models and relying on accurate tracking.

2 Continuous Gesture Recognition

Problem Definition:

In [4], Derpanis et al. introduced the idea of breaking down signs into constituent elementary gestures. They generalised a set of 14 elementary gestures and also a set of basic hand shapes based on linguistic theory. Sign language recognition can be therefore done by recognising the hand shapes, the elementary gestures and the corresponding sequence. To simplify the problem, we will focus on continuous sign recognition by analysing hand motion only (i.e. ignoring the effect due to the change in hand shape). Under this context, continuous gesture recognition can be broken down into two processes: isolated gesture recognition and temporal segmentation (or gesture spotting). This is to say, while passing through a video stream, a segment of video will be extracted and tested against several classification models to determine the likelihood of existence of a certain gesture. Once the likelihood exceeds a certain threshold, the corresponding gesture is spotted out.

In this paper, we adopt a similar gesture decomposition approach as Derpanis et al.'s work. Instead of having 14 elementary gestures as in their work, we have only 9 (shown in Figure 1): (1) upward, (2) downward, (3) rightward, (4) leftward, (5) toward signer, (6) away signer, (7) nod, (8) supinate, and (9) pronate, by eliminating complex gestures such as 'Up and down'. The research problem in this paper is to spot these nine gestures from a given video sequence of hand motion.

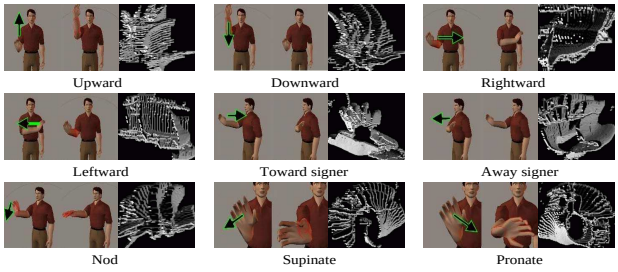

Figure 1: The 9 elementary gestures and their MGO images.

Mathematical Formulation:

As explained previously, continuous gesture recognition consists of isolated gesture recognition and temporal segmentation. Given a video sequence, $V = \{I_0, I_1, ..., I_t\}$ where I_t is an image captured at time t, continuous gesture recognition means obtaining the most probable sequence of gestures ($C = c_1, c_2, ..., c_n$) from the observation, V. In other words, we aim at finding C that maximises a probabilistic term $P(C \mid V)$. Since each gesture associates with a certain set of consecutive images, we introduce a sequence of segmentation functions, $\theta = \{\theta_1, \theta_2, ..., \theta_n\}$, to mark the gesture boundaries and each function is used to extract a set of consecutive images, $\{I_k, I_{k+1}, ..., I_{k+m}\}$, from the video sequence, V. Using this, the probabilistic term can then be rewritten as $\int P(C, \theta \mid V) d\theta$ and can be approximately expanded as $\int \prod_i P(c_i, \theta_i \mid V) d\theta_i$. It can be further approximated as $\prod_i (\int P(c_i \mid V, \theta_i) P(\theta_i \mid V) d\theta_i)$, and maximising this term can now be done by maximising (1) $P(c_i \mid V, \theta_i)$ which corresponds to the isolated gesture recognition problem, and (2) $P(\theta_i \mid V)$ which corresponds to the temporal segmentation problem.

The *isolated gesture recognition* problem can be considered as a classification problem where a motion feature is extracted from a portion (defined by θ_i) of the video sequence, V, and is classified into its corresponding gesture (i.e. c_i) by maximising the probabilistic term $P(c_i \mid V, \theta_i)$. Through experiments, Motion Gradient Orientation image is chosen as the motion feature and sparse Bayesian classifier is chosen as the classifier.

The *temporal segmentation* problem involves finding the most probable segmentation hypothesis, θ_i, via $P(\theta_i \mid V)$. The likelihood of a certain hypothesis can be evaluated using $P(c_i \mid V, \theta_i)$. In order to support multiple hypotheses on segmentation, a sampling technique based on CONDENSATION framework [5] is used in this work. The overview of the framework is illustrated in Figure 2. The implementation details of feature extraction, classification and hypotheses generation will be described in next section.

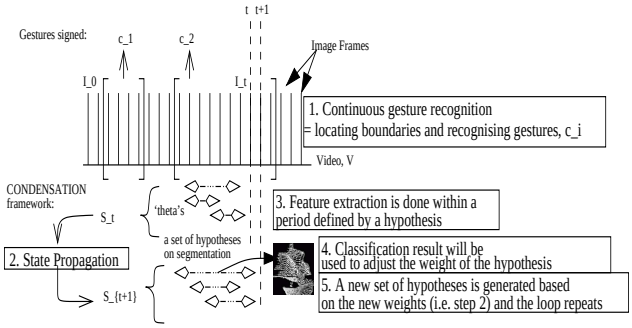

Figure 2: An overview of the framework used in this work.

3 Implementation Details

Process 1: Feature extraction

Motion gradient orientation (MGO) was proposed by Bradski and Davis [3] to explicitly *encode changes* in an image introduced by motion events. Given a video sequence, $V = \{I_0, I_1, ..., I_t\}$ where I_t is an image captured at time t (and the precise form is $I(x, y, t)$), a binary mask $D(x, y, t)$ can be obtained from image differencing and this mask can be used to indicate regions of motion at time t.

The motion history image (MHI) can be obtained as [1]:

$$H_\tau(x, y, t) = \begin{cases} \tau & \text{if } D(x, y, t) = 1 \\ \max(0, H_\tau(x, y, t-1) - 1) & \text{otherwise.} \end{cases} \quad (1)$$

where τ is set to 255, which is the highest intensity of a greyscale image. The motion gradient orientation (MGO) can then be computed as [3]:

$$\phi(x,y) = \arctan \frac{F_y(x,y)}{F_x(x,y)} \quad (2)$$

where $F_x(x,y)$ and $F_y(x,y)$ are the spatial derivatives along x and y direction of the MHI. After performing normalisation (e.g. $\phi(x,y)$ is rescaled between 1 and 255), MGO features can be obtained (illustrated in Figure 1).

Process 2: Classification

Variation in MGO features due to inter- and intra- personal variation can be huge and thus a strong classifier is needed. In this work, a sparse Bayesian classifier (or a Relevance Vector Machine (RVM) [9]), is used. RVM is used instead of Support Vector Machine because RVM gives a *sparse* classification model and a *probabilistic* output.

Given a training set $\{\mathbf{x_n}, t_n\}_{n=1}^{N}$, the problem of learning a binary classifier can be expressed as that of learning a function f so that the input feature $\mathbf{x_n}$ will map onto their correct classification label t_n and the probability of $\mathbf{x_n}$ is classified as the target class (where $t_n = 1$) equals to $\sigma(y_n) = 1/(1+e^{-y_n})$ where $y_n = f(\mathbf{x_n})$. The function f can be written as a classification model building from a sparse set (with size $M \ll N$) of prototypes (or relevance vectors). Readers may refer to [9] and our previous work [10] for more details on the learning algorithm. In this work, the RVM classifier is extended to handle multi-class classification using "one-versus-others" training scheme.

Process 3: Hypotheses generation

In order to perform temporal segmentation reliably, several *segmentation hypotheses* are generated at each time frame and each hypothesis is evaluated using the likelihood of existence of a certain gesture. The algorithm on hypotheses generation and evaluation is shown in Algorithm 1.

Algorithm 1 Hypothesis Generation Framework

- Train RVM classifiers, $f_c(x)$, for each gesture, c
- Initialise a sample set, $S_0 = \{\theta_0^{(j)}, \gamma_0^{(j)} = \frac{1}{J}\}_{j=1}^{J}$

for $t = 1$ to T **do**
  for $j = 1$ to J **do**
    - Obtain a sample state, $\theta_{t-1}^{(j)}$, from a sample set S_{t-1}
    - Obtain a propagated state, $\theta_t^{(j)}$ based on $P(\theta_t^{(j)} \mid \theta_{t-1}^{(j)})$
    - Obtain a motion feature, $x_t^{(j)}$, from the video V using $\theta_t^{(j)}$
    - Evaluate the likelihood of state $\theta_t^{(j)}$ using the RVMs, $f_c(x_t^{(j)})$
    - Update the weight, $\gamma_t^{(j)}$, of the state using $\sigma(\max_c(f_c(x_t^{(j)})))$
  end for
  if $\sigma(\max_{c,j}(f_c(x_t^{(j)})))$ larger than threshold **then**
    - Report the detected gesture and reinitialise the sample set
  end if
  - Normalise the weights, $\gamma_t^{(j)}$, and then form a new sample set S_t
end for

In this work, each hypothesis on segmentation, $\theta_{t-1}^{(j)}$, consists of 2 parameters, namely offset position and duration. Under the proposed framework[1], state propagation is done according to $P(\theta_t^{(j)} \mid \theta_{t-1}^{(j)})$, which is a normal distribution with mean $\theta_{t-1}^{(j)}$, and state evaluation is performed using $P(c \mid \theta_t^{(j)})$, where isolated gesture classification (refer to Process 1 and 2) is done on video segmented by $\theta_t^{(j)}$. At the end of each iteration, likelihood of each hypothesis is updated according to the state evaluation result and a new sample set of hypotheses is generated as in [5]. The algorithm can be implemented to support real-time application.

4 Experimental Result

The proposed method was implemented using unoptimised C++ code and the OpenCV library. All the experiments described were executed on a P4 2.4GHz PC with 1G memory. Both training and testing data are video captured under arbitrary room conditions. The video was captured by a webcam with a resolution of 320×240 pixels at 15 frames per second (fps). In each video clip, the signer signs the elementary gestures described in Section 2. On average, each gesture lasts around 2 seconds.

Test 1: Isolated Gesture Recognition

In this test, a training set was obtained by inviting a signer to sign all nine gestures described in Section 2 and a testing set was obtained by inviting 5 other signers to sign those gestures under different capturing environment. Gestures presented in the video clips were manually segmented. Given the training set of size 628 and using the "one-versus-others" classification scheme, each RVM classifier maintains a sparse model with around 30 relevance vectors eventually (and at most 40). The overall accuracy on 1025 test cases is 89.7%. Among these 1025 test cases, 73 of them cannot be mapped into any class (i.e. 7.1% of test cases). A detailed report of this test can be found in our previous work [10].

Test 2: Locating Transition between Gestures

In this test, a testing set was obtained by inviting a signer to sign 17 combinations of gestures (which are shown in Table 1). Each combination consists of 2 gestures and contributes to 30 samples (or video clips) in the testing set. The proposed algorithm (using the classifiers trained in Test 1) was used to segment and classify the gestures presented. The classification result is shown in Table 1.

Test 3: Continuous Gesture Recognition

A testing set was obtained by asking a signer to sign 30 sequences of gestures. Each sequence forms a video clip

[1] Observation term, V, is omitted here for simplicity.

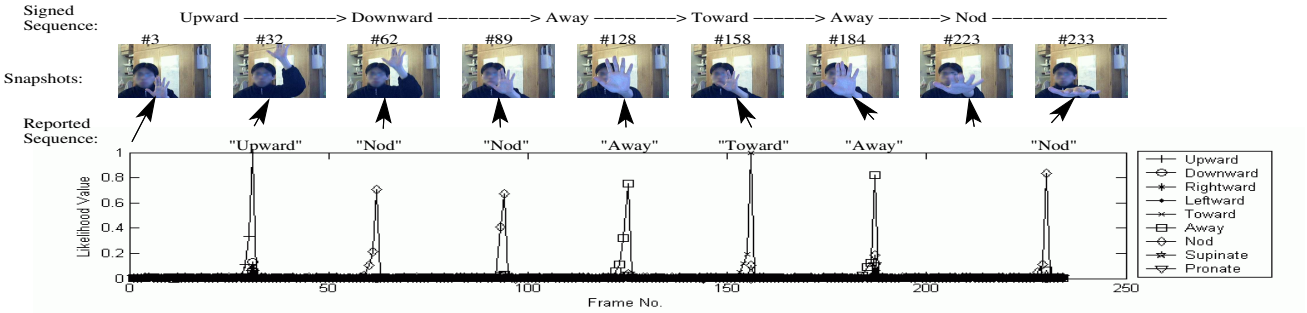

Figure 3: This figure shows the recognised gestures and the change in likelihood values reported by the proposed algorithm given a typical testing sequence used in Test 3.

Transition	Accuracy	Transition	Accuracy
Up → Down	83.3%	Up → Left	93.3%
Up → Away	86.7%	Down → Up	83.3%
Down → Left	86.7%	Down → Away	83.3%
Right → Up	93.3%	Right → Left	90.0%
Right → Away	83.3%	Left → Right	90.0%
Toward → Up	83.3%	Toward → Left	86.7%
Toward → Away	83.3%	Away → Toward	83.3%
Away → Nod	80.0%	Away → Supinate	80.0%
Away → Pronate	83.3%		

Table 1: Classification result on segmenting and recognising two consecutive gestures. In this experiment, correct recognition means successful recognition of the two gestures presented in each testing video clip.

and consists of more than 2 gestures combined in a random order. The proposed algorithm was used to segment and recognise the gestures presented and the result is shown in Table 2. The number of hypotheses generated per frame is 30 and the time taken per iteration is 49.5 ms (i.e. 20.2 fps). Figure 3 illustrates the recognition process on a typical testing clip. The proposed algorithm can also be implemented to work on a live video feed from camera at the mentioned frame rate. If more constraints (such as average duration) are added, the accuracy can be boosted to around 90%.

N	I	D	S	C	Accuracy
127	12	7	9	111	77.95%

Table 2: Classification result on segmenting and recognising continuous gestures. Notation used (see [8]): Accuracy = (N-I-D-S)/N, N is the total number of gestures appeared in the testing set (30 video clips), I is the number of insertions, D is the number of deletions, S is the number of substitutions and C is the number of correctly recognised gestures.

5 Conclusion

A new method is proposed to recognise continuous elementary gestures in this paper and it can be applied to sign language recognition. The proposed method performs better than recently used methods in two ways. Firstly, by using a sparse Bayesian classifier, the classification result can be obtained with a minimum amount of online computational resources and thus the system can run in *real-time*. Secondly, the probabilistic nature of the Bayesian classifier and CONDENSATION framework realises a pure *probabilistic solution* towards the continuous gesture recognition problem and ensures robustness. The main disadvantage of the proposed method is the difficulty in handling ambiguous gestures (e.g. 'downward' was incorrectly recognised as 'nod' in Figure 3) and therefore further investigation will be done in this aspect.

Acknowledgements. SW is funded by the Croucher Foundation Scholarships (Hong Kong).

References

[1] A. F. Bobick and J. W. Davis. The recognition of human movement using temporal templates. *PAMI*, 23(3):257–267, 2001.

[2] R. Bowden, D. Windridge, T. Kadir, A. Zisserman, and M. Brady. A linguistic feature vector for the visual interpretation of sign language. In *Proc. ECCV*, 2004.

[3] G. R. Bradski and J. W. Davis. Motion segmentation and pose recognition with motion history gradients. *Machine Vision and Applications*, 13(3):174–184, 2002.

[4] K. G. Derpanis, R. P. Wildes, and J. K. Tsotsos. Hand gesture recognition within a linguistics-based framework. In *Proc. ECCV*, 2004.

[5] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. ECCV*, 1996.

[6] R. H. Liang and M. Ouhyoung. A real-time continuous gesture recognition system for sign language. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, 1998.

[7] C. Shan, Y. Wei, X. Qiu, and T. Tan. Gesture recognition using temporal template based trajectories. In *Proc. ICPR*, pages 954–957, 2004.

[8] T. Starner, J. Weaver, and A. Pentland. Real-time american sign language recognition using desk and wearable computer based video. *PAMI*, 20(12):1371–1375, 1998.

[9] M. E. Tipping. Sparse bayesian learning and the relevance vector machine. *The Journal of Machine Learning Research*, 1:211–244, 2001.

[10] S.-F. Wong and R. Cipolla. Real-time interpretation of hand motions using a sparse bayesian classifier on motion gradient orientation images. In *Proc. BMVC*, 2005.