Modelling and Recognition of Signed Expressions Using Subunits Obtained by Data–Driven Approach

Mariusz Oszust and Marian Wysocki

Department of Computer and Control Engineering,
Rzeszow University of Technology,
2 Wincentego Pola, 35-959 Rzeszow, Poland
{moszust,mwysocki}@prz-rzeszow.pl
http://www.prz-rzeszow.pl

Abstract. The paper considers automatic vision based modelling and recognition of sign language expressions using smaller units than words. Modelling gestures with subunits is similar to modelling speech by means of phonemes. To define the subunits a data–driven procedure is proposed. The procedure consists in partitioning time series of feature vectors obtained from video material into subsequences which form homogeneous clusters. The cut points are determined by an optimisation procedure based on quality assessment of the resulting clusters. Then subunits are selected in two ways: as clusters' representatives or as hidden Markov models of clusters. These two approaches result in differences in classifier design. Details of the solution and results of experiments on a database of 101 Polish words and 35 sentences used at the doctor's and in the post office are given. Our subunit–based classifiers outperform their whole–word–based counterpart, which is particularly evident when new expressions are recognised on the basis of a small number of examples.

Keywords: Sign language recognition; Time series segmentation; Time series clustering; Evolutionary optimisation; Hidden Markov models; Computer vision; Data mining

1 Introduction

Communication disorder such as hearing loss is a significant problem in social contacts. Therefore, there is a justification of research on supporting deaf people with vision–based, automatic sign language recognition system ensuring translation of hand gestures into spoken or written language. In most of such systems

Please cite this paper as follows: Oszust M., Wysocki M.: Modelling and Recognition of Signed Expressions Using Subunits Obtained by Data–Driven Approach. Ramsay A., Agre G., (Eds.), AIMSA, vol. 7557, Lecture Notes in Computer Science, pp. 315–324. Springer, 2012. The final publication is available at http://link.springer.com/chapter/10.1007%2F978-3-642-33185-5_35

(see e.g. [5], [7]) one word model represents one sign. They can achieve good performance only with small sign vocabularies because the training corpus and the training complexity increase with the vocabulary size. Large–vocabulary systems require signed expressions modelling using smaller units than words. Such units are called subunits, and modelling is similar to modelling speech by means of phonemes. The main advantage of this approach is that an enlargement of the vocabulary can be achieved by composing new signs through concatenation of subunit models and by tuning the composite model with only small sets of examples. In the literature different vision–based subunit segmentation algorithms have been reported presenting how to break down signs into subunits. For example Liddell and Johnson developed movement–hold model [15] modelling each word with series of movement and hold segments. In this approach a signer is assumed to make clear and not natural pauses during signing. Kraiss in [5] presents an iterative process of data–driven extraction of subunits using hidden Markov models (HMMs). In all following steps, two state HMMs for subunits determined in prior iteration step are concatenated to models of single signs. The boundaries of subunits for the next step result from the alignment of appropriate feature vector sequence to the states by the Viterbi algorithm. Theodorakis et al. in [11] describe their approach with pronunciation clustering step with respect to each sign. Pronunciation groups are found by HMMs, and then hierarchically clustered at the HMM level. After clustering model–based subunits are obtained. Kong et al. [4], in turn, use a naive Bayesian network to detect subunits' boundaries. They created subunits' transcription and trained HMMs to recognise the subunits in test expressions. Different approach can be found in [2]. The authors define subunits' boundaries using hand motion discontinuity, adapt temporal clustering by dynamic time warping (DTW) to merge similar time series segments, and finally select a representative subunit from each cluster.

In our work, we propose a new approach to find subunits' boundary points as the solution of an optimisation problem. The problem consists in finding subunits which can be grouped in clusters of good quality. Subunits' boundary points are determined by an immune–based, evolutionary algorithm [1], [13]. The approach refers to clustering of time series data [6], [16], and cluster–based time series segmentation [14]. The contribution of the paper lies in (1) formulation of the problem of determining subunits for sign language recognition as time series clusters' representatives or hidden Markov models of clusters, (2) formulation of the problem of modelling signed expressions with subunits, (3) proposition of solution methods, and verification of the approach by experiments on real data. It is worth noticing that we present the results of experiments when new expressions are recognised based on a small number of examples, justifying the need for the use of subunits models. We also present the recognition of sentences on a basis of subunit models obtained for words.

The rest of the paper is organized as follows. Section 2 gives preliminary information concerning Polish sign language (PSL) and the features derived from sign language videos. Section 3 contains description of the subunits extraction

problem. Section 4 gives details on the recognition method. Recognition experiments and their results on 101 isolated PSL words and 35 sentences are given in Sect. 5. Section 6 concludes the paper.

2 Characteristics of the PSL and feature vectors

PSL signs are static or dynamic and mostly two–handed. Hands often touch each other or appear against the background of the face. Their motion can be single or repeated.

Every sign can be analysed by specifying at least three components: (i) the place of the body against which the sign is made, (ii) the shape of a hand or hands, (iii) the movement of a hand or hands [15]. Although in practical sign language communication some additional features (such as lip shape, face expression, etc.) are often used, we do not consider them in this paper.

For detection of the signer's hands and face we used a method based on a chrominance model of human skin. To detect skin–toned regions in a colour image, the image is transformed into a gray–tone form using the skin colour model in the form of a 2D Gaussian distribution in the normalised RGB space, and thresholded. The individual pixel intensity in a new image represents a probability that the pixel belongs to a skin–toned region. The areas of the objects toned in skin colour, their centres of gravity and ranges of motion are analysed to recognise the right hand, the left hand and the face. Comparison of neighbouring frames helps to notice whether the hands (the hand and the face) touch or partially cover each other. In order to ensure correct segmentation there were some restrictions for the background and the clothing of the signer.

We use seven features for each hand: the position of the hand with respect to the face (three spatial coordinates), the area, orientation, compactness and eccentricity of the hand. In our approach shape of the hand is described in a rough manner due to small hand size in respect to the observed size of the signer hindering more accurate modelling of the hand.

3 A data–driven subunits extraction method

3.1 The input data

Let $S = \{X_1, X_2, \ldots, X_n\}$ denote a data set, where a sequence $X_i = \{x_i(1), x_i(2), \ldots, x_i(T_i)\}$ of real valued feature vectors represents a signed expression. All vectors $x_i(t)$, where $i \in I = \{1, 2, \ldots, n\}$, and t is a time sampling point, $t \in \mathcal{T}_i = \{1, 2, \ldots, T_i\}$, have identical structure and contain features (see Sect. 2). Two time sequences X_i and $X_{j \neq i}$ may represent different expressions or different realisations of the same expression.

In signed expressions modelling we take into account that the features appear both sequentially and simultaneously. For example, the hand shape and hand position can change independently at the same time [5]. To model parallel processes

we will distinguish N groups of features (channels). This is based on the assumption that the separate processes evolve independently from one another with independent output. Therefore, we will write $x_i(t) = [x_i^1(t), x_i^2(t), \ldots, x_i^N(t)]$ and, in accordance with it, we will use an upper index to indicate time series related to a group: $X_i^l = \{x_i^l(1), x_i^l(2), \ldots, x_i^l(T_i)\}, S^l = \{X_1^l, X_2^l, \ldots, X_n^l\}, l \in \mathcal{N} = \{1, 2, \ldots, N\}$. During extraction of subunits all elements in a group will be considered jointly, whereas different groups will be considered separately. For instance, one can distinguish two independent channels (N= 2) related with both hands, or 14 independent, features related channels.

3.2 Time series partitioning

Let us consider a time decomposition D^l, which, for each $i \in I$, defines a number $k_i^l = k_i^l(D^l) \geq 1$ and $k_i^l - 1$ cut points $t_{ij}^l = t_{ij}^l(D^l)$, where $1 < t_{i1}^l < t_{i2}^l < \ldots < t_{i,k_i^l-1}^l < T_i$. The decomposition means that X_i^l is partitioned into k_i^l subsequences. The first subsequence $s_{i1}^l(D^l)$ starts at $t = 1$ and ends at $t = t_{i1}^l$, the next subsequence $s_{i2}^l(D^l)$ starts at $t = t_{i1}^l$ and ends at $t = t_{i2}^l$, and so on until the last subsequence $s_{i,k_i^l}^l(D^l)$ which starts at $t = t_{i,k_i^l-1}^l$ and ends at T_i. The resulting data set $S'^l(D^l) = \{s_1'^l(D^l), s_2'^l(D^l), \ldots, s_n'^l(D^l)\}$, where $s_i'^l(D^l) = \{s_{i1}^l(D^l), \ldots, s_{i,k_i^l}^l(D^l)\}, i \in I$, contains $n'^l = n'^l(D^l) = \sum_{i=1}^n k_i^l(D^l)$ subsequences. The length of each subsequence is constrained by the minimal L_{min} and the maximal L_{max} number of points.

We propose determining a good decomposition into subsequences by solving a decision problem, based on the following main steps: (i) partition the set $S'^l(D^l)$ into m^l (a given number) clusters, i.e. $S'^l(D^l) = \{C_1^l(D^l), C_2^l(D^l), \ldots, C_{m^l}^l(D^l)\}$, (ii) evaluation of the decomposition D^l using a criterion (index) $J(D^l)$ which characterises the quality of the resulting clusters. The criterion is the mean distance among the elements and their cluster centres.

3.3 Optimisation method

Our approach to solve the time series partitioning problem (see Subsect. 3.2) adapts the immune–based, evolutionary algorithm CLONALG, which is often used in a wide variety of optimisation tasks [1], [13] especially for solving problems with many local optima and constraints. In the sequel we shortly describe the algorithm, the encoding method, and the mutation operator.

The main loop of the CLONALG algorithm [13] (repeated *gen* times, where *gen* is the number of generations) consists of four main steps: one initial step where all the elements of the population are evaluated computing $J(D^l)$ and three transformation steps: clonal selection, mutation, apoptosis. Elements in the population are often called lymphocytes or antibodies. The antibody represents a decomposition D^l of the set S^l into a set S'^l. It has the form of the integer valued vector $D^l = [t_{11}^l, t_{12}^l, \ldots, t_{1,k_1^l-1}^l, t_{21}^l, t_{22}^l, \ldots, t_{2,k_2^l-1}^l, \ldots, t_{n1}^l, t_{n2}^l, \ldots, t_{1,k_n^l-1}^l]$ composed of the cut points of the original sequences. In the clonal selection

step algorithm chooses a reference set consisting of h elements at the top of the ranking. The mutation process consists of a given number mut of mutations conducted on a population element. The mutation means an operation changing solution maintained in antibody which satisfies the length constraints. In the step of the apoptosis b worst elements in population are replaced by randomly generated elements.

Before clustering we must define a distance between sequences. We used dynamic time warping (DTW) [9], which allows a nonlinear mapping of one sequence to another by minimizing a distance between them. The main motivation for using DTW is its ability to expand or compress the time comparing sequences that are similar but locally out of phase. For example, some related parts of gestures representing the same expression may be performed with different velocities. Length of compared sequences can be different. In experiments we used K-means clustering algorithm [16], [12], which works with vector defined data due to calculation of clusters' centres at each step. Center of the cluster in K-means is a mean of all its elements, and it requires that elements in a cluster should have equal length. Therefore, we considered two approaches based on $n'^l(D^l)$ similarity vectors representing the set $S'^l(D^l)$ of sequences to be clustered. Each of the similarity vectors has $n'^l(D^l)$ elements where the $j-th$ element of the $i-th$ similarity vector is determined as the DTW distance between sequences s'_i and s'_j in the set $S'^l(D^l)$. In second approach sequences were represented by short vectors containing statistical information (i.e. mean and standard deviation). DTW similarity vectors are large then generating shorter vectors by principal component analysis (PCA) [12] may be a solution. In our experiments we used short vectors.

The optimisation results in obtaining a good decomposition D^l_{opt}. We can use it to transform each sequence X^l_i to a string of labels $X^{ls}_i = \{e^l_{i1}, e^l_{i2}, \ldots, e^l_{i,k^l_i}\}$, where $e^l_{ik} \in E^l = \{\alpha^l_1, \alpha^l_2, \ldots, \alpha^l_{m^l}\}$, α^l_k denotes the label assigned to the cluster $C^l_k(D^l_{opt})$, and e^l_{ik} is a label of the cluster the subsequence $s^l_{ik}(D^l_{opt})$ belongs to. Let us denote by X^s_i the string–based counterpart of X_i, i.e. $X_i = \{X^{1s}_i, X^{2s}_i, \ldots, X^{Ns}_i\}$ and, consequently, by S^s the counterpart of S.

4 Recognition

The subunits can be selected in two ways: as clusters' representatives or HMMs of clusters. Two types of subunits result in differences in a classifier design. Let us assume that an expression to be classified is represented by a sequence $Y = \{y(1), y(2), \ldots, y(T_y)\}$. The feature vectors $y(.)$ have the same structure as $x(.)$ and therefore the sequences $Y^l = \{y^l(1), y^l(2), \ldots, y^l(T_y)\}$, where $l \in \mathcal{N}$, will be considered separately. Two problems have to be solved. The first problem consists in finding an appropriate string representation of Y^l, i.e. $Y^{ls} = \{e^l_{y1}, e^l_{y2}, \ldots, e^l_{y,k^l_y}\}$, where $e^l_{yk} \in E^l$ and, consequently, the string representation Y^s of Y (according to the first, aforementioned representation of signed expressions). The second problem is to find $NN(Y^s)$ – the nearest neighbour of Y^s in the set S^s. Then the unknown expression is assigned to the class which

$NN(Y^s)$ belongs to. The string representation can be found by solving an optimisation problem with respect to cut points of Y^l for each $l \in \mathcal{N}$. Let $D_y^l = [t_{y1}^l, t_{y2}^l, \ldots, t_{y,k_y^l-1}^l]$ characterises a decomposition. As opposed to the previous optimisation, now the criterion to be minimized is $J(D_y^l) = \sum_{k=1}^{k_y^l} d_{DTW}(k)$, where $d_{DTW}(k)$ denotes the DTW distance between the $k-th$ subsequence $s_{y,k}^l(D_y^l)$ of Y^l and its nearest neighbour $NN(s_{y,k}^l(D_y^l))$ in the set $S'^l(D_{opt}^l)$.

The optimisation task is solved by CLONALG. Then e_{yk}^l is a label of the cluster the $NN(s_{y,k}^l(D_{y,opt}^l))$ belongs to. The procedure is repeated for each $l \in \mathcal{N}$. The second problem is also an optimisation task. Here the so called edit distance [16] is used as a measure of the difference between two strings. The method resembles DTW. It uses dynamic programming to find a minimum number of operations (insert, delete, replace) required to transform one string into the other. The sequence Y becomes assigned to the class X_j belongs to, where edit distance is minimal.

In the second representation of signed expressions sequences which belong to a cluster were used to train HMMs [8]. A HMM is a model used to characterize the statistical properties of a signal, consists of two stochastic processes: one is unobservable Markov chain with a finite number of states, an initial state probability distribution and a state transition probability matrix; the other is characterized by a set of probability density functions associated with observations generated by each state. We assumed that each cluster can be represented by one state HMM with Gaussian output. We used HTK software [17] to design the HMM–based models. Words (sentences) were recognized using a composite model built as a network of isolated subsequence models. The scheme used a statistical information about the transition probability between two successive subunits, calculated for any subunit in relation to each possible preceding subunit from the training corpus (bigram language model [5], [17]). The parsing were performed by a Viterbi algorithm based on token passing [17]. The modelling was proceeded in two steps. First, isolated subunit models were trained using the Viterbi algorithm and appropriate training data. Then parameters of these models were tuned on the basis of whole words/sentences. The HTK offers so–called embedded training that makes it possible. Embedded training uses the same procedure as for the isolated subunit case, but rather than training each model individually, all models are trained simultaneously. The location of subunit boundaries in the training data is not required for this procedure, but the symbolic transcription of each training sequence is needed [17]. This transcription is obtained during the subunit determining process described earlier. Networks of elementary HMMs representing whole words are automatically created.

5 Experiments

In order to examine usability of designed subunit-based classifiers in recognition task we have performed set of experiments on real sequences obtained for signed

Polish words and sentences. The sequences represent 101 words and 35 sentences which can be used at the doctor's and in the post office. Each expression was performed 20 times by two signers (resulting in 4040 words' realisations and 1400 sentences' realisations). One signer is a PSL teacher, the other has learnt PSL for purposes of this research. The data have been registered with the rate 25 frames/s. The following Subsections present results of recognition. First we use cross–validation to estimate performance of the subunit–based classifiers of isolated words. Next we consider recognition of new words, i.e. not included in the vocabulary used to determine the subunits, on the basis of a small number of examples. Last experiment concerns recognition of PSL sentences on the basis of subunits determined from words.

5.1 Cross–validation

All word's realisations were divided into ten disjoint subsets in order to perform cross-validation tests. Each subset consisted of data corresponding to four repetitions of each word (two repetitions performed by each signer). We performed ten experiments using nine subsets as the training set S and the remaining subset as the test set. Because of the stochastic nature of the optimisation method each experiment has been repeated ten times. Subunits for each feature were extracted independently ($N = 14$). Parameters used by immune algorithm were as follows: $B = h = 20$; $c = 5$; $b = 2$; $mut = 2$; $gen = 100$; $L_{min} = 4$; $L_{max} = 8$. The optimisation task was solved for $m^l = 10$ clusters.

Table 1 shows the recognition performance during cross–validation for two classifiers (see Subsect. 4). Determining subunits during the learning stage took about 11 minutes, on the processing unit with 3.3 GHz, 4 cores, 16GB RAM, Windows 7 64 bit, and Java. The average time needed for recognition of one gesture from the test set was approximately 1 s. Representing the clusters of subunits obtained during the training by its medoids, and using the medoids instead of the whole clusters, speeds up the optimisation step described in Section 4, which makes the recognition about seven times faster (approx. 0.15 s) and with HMMs models of clusters approx. 0.26 s, but at the cost of recognition rate (approx. 2 percent). We also performed an experiment showing the impact of the number of clusters on recognition rate, good results (in the sense of ten–fold cross–validation) were obtained for more than five clusters.

5.2 Recognition of New Words and Sentences

This Subsection considers a situation when some new words are recognised on the basis of small sets of examples presenting usability of our subunit–based approach in extending vocabulary of recognised expressions. Experiments presented below are motivation of the use of subunits in large–vocabulary systems. We used the same data–set as before. Instead of preparing new data we randomly chose ten words (since then called new) from the data set and omitted them from the process of determining subunits. So, models of subunits were determined on the basis of data related to remaining 91 words. A small number w of examples

Table 1. Sample results of the cross validation test for two classifiers based on different subunits' models. Because of the stochastic nature of the optimisation method each experiment has been repeated ten times. Mean recognition rate and standard deviation in %.

Testing subset	1	2	3	4	5	6	7	8	9	10
Subunits as subsequences in clusters										
Mean	96.88	99.11	99.38	99.11	99.58	99.01	98.32	99.41	99.08	98.94
StDev	0.50	0.42	0.46	0.35	0.17	0.44	0.33	0.42	0.35	0.57
Subunits as HMMs of clusters										
Mean	89.70	95.22	84.93	97.00	95.17	96.51	94.03	93.29	94.23	96.66
StDev	0.97	1.04	1.58	0.77	1.66	0.69	1.10	2.16	1.57	0.82

of each new word was used to tune subunit-based models of these words. Remaining $40-w$ examples of the new words were used for testing. This experiment has been repeated 20 times, each time with different group of new words. We repeated the experiment with whole word models and nearest neighbour classifier based on DTW distance. Figure 1 shows mean values of recognition rates in relation to the number w of examples. As we can see, a relatively small number of examples enables good recognition. Better recognition rate obtained by classifiers based on subunits can be explained as follows: the whole word models were represented by small sets of examples, whereas the subunit–based models additionally used information accumulated in subunits that have been modelled on the basis of large sets.

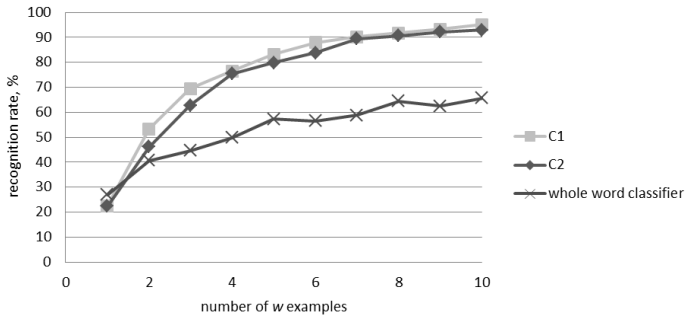

Fig. 1. Recognition rate in % of new words vs. the number w of examples. *C1* denotes classifier using subunits taken from subsequences' clusters, *C2* denotes classifier using HMMs based subunits.

Each of 40 realisations of 35 sentences was optimally transcribed using the subsequences obtained for isolated words. The transcriptions were performed in the way described in Section 4, but here the sequences represented sentences

instead of words. The resulting models will be called A–models. Another model of each sentence (B–model) was created directly from concatenated transcriptions of constitutive words. It represents idealized training examples, which do not take into account the coarticulation phenomenon [10]. We considered the recognition rate in dependence on the number of examples. So for each sentence we took its B–model and a small number α of its drawn A–models, as the training set, and the remaining $40 - \alpha$ A–models for testing. Experiment was repeated 20 times. The recognition rates are shown in Fig. 2. As we can see, a relatively small number of examples enables good recognition. Note that the quite acceptable result of 68.4% for $\alpha = 0$ corresponds to the case when only B–models are present in the learning set. It is the common situation when a word in a sentence differs from its isolated realisation. Results obtained for classifier based on subunits modelled by HMMs are weaker for small number of sentences examples because some resulted HMMs of sentences were too long (sometimes more than 20 states).

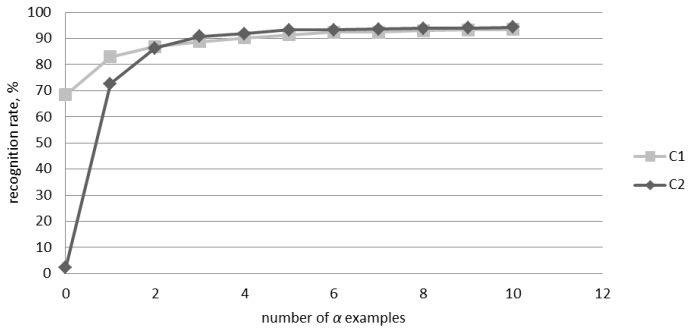

Fig. 2. Recognition rate in % of sentences vs. the number α of examples. *C1* denotes classifier using subunits taken from subsequences' clusters, *C2* denotes classifier using HMMs based subunits.

6 Conclusions

Large–vocabulary sign language recognition require the modelling of signed expressions using smaller units than words. However, an additional knowledge of how to break down signs into subunits is needed. In vision–based systems the subunits are related to visual information. As linguistic knowledge about the useful partition of signs in regard of sign recognition is not available, the construction of an accordant partition is based on a data–driven process when signs are divided into segments that have no semantic meaning - then similar segments can be grouped and labelled as a subunit or an HMM model using cluster of such segments can be trained. In this paper we propose a new approach to determining the subunits, which boundaries are considered as decision variables in

an optimisation problem. Our approach has been successfully verified on a data base of 101 Polish sign language expressions. In future research we are considering more advanced experimentation including recognition words and sentences of PSL.

Acknowledgment. This research was partially supported by Polish Ministry of Science and Higher Education under the grant no. U-8608/DS/M.

References

1. De Castro, L.N., Von Zuben, F. J.: Learning and optimization using the clonal selection principle. In: IEEE Trans. on Evolutionary Computation, vol. 6, pp. 239–251 (2002)
2. Han, J., Awad, G.,A. Sutherland, A.: Modelling and segmenting subunits for sign language recognition based on hand motion analysis. Pattern Recognition Letters, vol. 30, pp. 623–633 (2009)
3. Hendzel, J.K.: Dictionary of Polish Sign Language (in Polish). OFFER, Olsztyn (1985)
4. Kong, W.W., Ranganath, S.: Automatic Hand Trajectory Segmentation and Phoneme Transcription for Sign Language. In 8th IEEE International Conference on Automatic Face and Gesture Recognition FG '08, pp. 1–6 (2008)
5. Kraiss, K.F.: Advanced man–machine interaction. Springer, Berlin (2006)
6. Liao, T.W.: Clustering of time series data – a survey. Pattern Recognition, vol. 38, pp. 1857–1874 (2005)
7. Ong, S.C.W., Ranganath, S.: Automatic sign language analysis: a survey and the future beyond lexical meaning. In: IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 6, pp. 873–891 (2005)
8. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. In: IEEE Proceedings, vol. 77, pp. 257–286 (1989)
9. Ratanamahatana, C.A., Keogh, E.: Three myths about dynamic time warping data mining. In: SIAM Int. Conf. on Data Mining, pp. 506–510 (2005)
10. Segouat, J., Braffort, A.: Toward Modeling Sign Language Coarticulation, Gesture in Embodied Communication and Human-Computer Interaction. In: Kopp S., Wachsmuth I., Eds., Lecture Notes in Computer Science, vol. 5934, pp. 325–336. Springer Berlin/Heidelberg (2010)
11. Theodorakis, S., Pitsikalis, V., Maragos, P.: Model–level data-driven sub–units for signs in videos of continuous sign language. In: IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), pp. 2262–2265 (2010)
12. Theodoridis, A., Kontroumbas, K.: Pattern Recognition. Acad. Press, London (1999)
13. Trojanowski, K., Wierzchon, S.: Immune-based algorithms for dynamic optimization. Information Sciences, vol. 179, pp. 1495–1515 (2009)
14. Tseng, V.S., Chen, C.H., Huang, P.C.,Hong, T.P.: Cluster-based genetic segmentation of time series with DWT. Pattern Recognition Letters, vol. 30, no. 13, pp. 1190–1197 (2009)
15. Vogler, C., Metaxas, D.A.: Framework for recognizing the simultaneous aspects of American sign language. Computer Vision and Image Understanding, vol. 81, no. 3, pp. 358–384 (2001)

16. Xu, R., Wunsch, D.C.: Clustering. J. Wiley and Sons, Inc., Hoboken, New Jersey (2009)
17. Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P.: The HTK Book. Cambridge University (2006)