# Fingerspelling Recognition with Support Vector Machines and Hidden Conditional Random Fields

## A comparison with Neural Networks and Hidden Markov Models

César Roberto de Souza, Ednaldo Brigante Pizzolato, and Mauro dos Santos Anjo

Universidade Federal de São Carlos, São Carlos, Brasil
{cesar.souza, ednaldo, mauro_anjo}@dc.ufscar.br

**Abstract.** In this paper, we describe our experiments with Hidden Conditional Random Fields and Support Vector Machines in the problem of fingerspelling recognition of the Brazilian Sign Language (LIBRAS). We also provide a comparison against more common approaches based on Artificial Neural Networks and Hidden Markov Models, reporting statistically significant results in k-fold cross-validation. We also explore specific behaviors of the Gaussian kernel affecting performance and sparseness. To perform multi-class classification with SVMs, we use large-margin Directed Acyclic Graphs, achieving faster evaluation rates. Both ANNs and HCRFs have been trained using the Resilient Backpropagation algorithm. In this work, we validate our results using Cohen's Kappa tests for contingency tables.

**Keywords:** Gesture Recognition, Fingerspelling, Sign Languages, LIBRAS, Support Vector Machines, Hidden Conditional Random Fields, Neural Networks, Hidden Markov Models, Discriminative Models.

## 1 Introduction

Human communication goes much further than the commonplace speaking, hearing, writing and reading activities. Whenever there is visual contact between a speaker and listeners, a whole continuum of information becomes available through the visual-gestural system. Moreover, when the phonological system is unavailable or damaged, the need for sign languages arises.

This work investigates the automatic recognition of sign languages. More specifically, we provide a comparison between recent advances and past approaches in pattern recognition to deal with the Brazilian Sign Language's manual alphabet. In the Brazilian Sign Language, henceforth LIBRAS, the use of the manual alphabet is only needed in specific occasions. Those occasions include, for example, explicitly spelling the name of a person or a location.

An example of a fingerspelling recognition system was given by [1], in which the authors considered an Artificial Neural Network (ANN) classification scheme based on a two stage architecture to deal with static gesture signs, followed by a bank of Hidden Markov Models (HMMs) to provide dynamic gesture classification. In this paper, we investigate the use of Support Vector Machines (SVMs) disposed in large-margin Decision Directed Acyclic Graphs (DDAGs) [2] to achieve static gesture classification, while at the same time providing comparisons against the ANN approach. Furthermore, we investigate the discriminative counterpart of the HMM-based classifiers given by Hidden Conditional Random Fields (HCRFs). We further explore the implications of those new models, their characteristics, advantages and drawbacks.

This paper is organized as follows. After this introduction, section 2 gives a list of related works, raising some points of interest and discussions. Section 3 gives an overview of the gesture recognition field, its motivation and a brief literature review. Section 4 presents the methods, models and tools used in this work. In section 5 we detail our experiments with fingerspelling recognition, presenting their results in section 6. We then conclude our work, giving final considerations in section 7.

## 2 Related Works

One of the earlier works on sign language recognition using HMMs were conducted by Starner [3] and colleagues [4]. His work with the American Sign Language (ASL) using a single camera had shown up to 99.2% accuracy. The success of his approach could be partially explained by a restricted vocabulary and the use of a grammar to make the problem tractable. Other correlated works were given by [5] in the recognition of the German Sign Language (*Deutsche Gebärdensprache*, DGS), and by [6] for the British Sign Language (BSL).

Besides [1], one of the most directly related works was developed by Dias et al. [7], who focused specifically on the movement aspect of the LIBRAS. By stating the problem in a convenient mathematical formulation, the authors provided a more tractable formalization of the movement recognition problem amendable to be solved by SOM networks, Learning Vector Quantization (LVQ) and fuzzy variants, attaining overall good results.

Moreover, other papers have already explored HCRFs [8] and other variants for gesture recognition. Morency et al. used LD-CRFs [9] to perform gesture recognition in continuous image streams, with excellent results. Elmezain et al. [10] also studied CRFs, HCRFs and LD-CRFs in the recognition of alphabet characters and numbers drawn in mid-air using hand trajectories, obtaining 91.52%, 95.28% and 98.05% for each model, respectively.

## 3 Gesture Recognition

Following a comprehensive survey conducted by Mitra and Acharya [11], one can say that gesture recognition methods have been traditionally divided into two main categories: Device-based and vision-based [5,7,10,12]. Device-based approaches often constrain the users to wear a tracking device, such as a tracking glove, resulting in less natural interaction. In this work we will focus only on vision-based approaches.

Gestures can be either static or dynamic. Static gestures, often known as *poses*, are individual still configurations performed by the user. They can often be registered in a single still image. On the other hand, dynamic gestures vary on time, and have to be captured as a sequence of still images (such as in the form of an image stream). Often, gestures have both elements, such as in the case of sign languages [11]. In this paper, we will be covering both gesture types.

Sign languages are natural languages, and therefore have their own structure and grammatical system. Unfortunately, sign languages (and gestures in general) are often ambiguous, in the sense that a specific sign can be used to denote different things depending on context. It is inevitable to make a parallel with the field of speech recognition: Juan and Rabiner reported how the paradigm shift — from generative to discriminative models — played a fundamental role in the field. The change was highly motivated by the fact that probability distributions governing acoustic speech signals could not be modeled accurately, turning Bayes decision theory "inapplicable under these circumstances" [13].

Discriminative models seem to show better performance than generative ones (although this claim is somewhat disputed, e.g. [14]). Nevertheless, there is an increasing literature interest in the application of CRFs, including applications in computer vision [10,15] and sign language recognition [9,12]. A comprehensive description of CRFs and HCRFs is given in [16].

## 4 Models and Tools

### 4.1 Artificial Neural Networks

As the name implies, at their creation ANNs had a strong biologic inspiration. However, despite their biological origins, they can be seen as simple functions $f: \mathbb{R}^n \rightarrow \mathbb{Y}$ mapping a given input $\boldsymbol{x} \in \mathbb{R}^n$ to a corresponding output $\boldsymbol{y} \in \mathbb{Y}$. The output vectors $\boldsymbol{y} = \langle y_1, \ldots, y_m \rangle$ are also restricted to a specific subset of $\mathbb{R}^n$. Each $y_i \in \boldsymbol{y}$ is restricted to a particular range according to the choice of activation function for the output neurons. In the case of a sigmoid activation function, this range is $[0; 1]$; in case of a bipolar sigmoid function, it is $[-1; 1]$.

The learning problem can be cast as a standard optimization problem, in which we would like to minimize divergence (such as measured by the error gradient) between produced outputs $\hat{\boldsymbol{y}}$ and desired answers $\boldsymbol{y}$. A promising method to minimize the error gradient is the Resilient Back-propagation algorithm (Rprop) [17], which is one of the fastest methods for gradient learning restricted solely to first-order information. Unlike other gradient based methods, such as Gradient Descent, in which the step size is always proportional to the gradient vector, Rprop takes into account only the direction of the gradient.

### 4.2 Support Vector Machines

Unlike ANNs, SVMs seem not to suffer from the curse of dimensionality (although the validity of this claim is again sometimes disputed, e.g. [18]). Nevertheless, SVMs have shown great performance in many real-world problems [12,19,20], including high dimensionality and large-scale ones. Its hyperplane decision function is given by

$$h(\boldsymbol{x}) = sgn(\boldsymbol{\theta} \cdot \boldsymbol{x} + b) \underset{\omega_2}{\overset{\omega_1}{\gtrless}} 0 \tag{1}$$

with $sgn(0) = 1$. Cortes and Vapnik [21] proposed finding a separating hyperplane using an approximate version of the Structural Risk Minimization principle: minimizing the structural risk through maximization of the classification margin, while at same time enforcing capacity control by controlling the margin's width. This problem could then be stated as a constrained optimization problem in the form

$$\min_{w,b,\xi} \frac{1}{2} \|\boldsymbol{\theta}\|^2 + C\left(\sum_{i=1}^{n} \xi_i\right) \tag{2}$$

subject to $y_i(\boldsymbol{\theta} \cdot \boldsymbol{x_i} + b) \geq 1 - \xi_i$ in which $\xi_i \geq 0$ are slack variables and $C$ is a regularization term imposing a weight to the training set error minimization in contraposition to minimizing model complexity. A large value for $C$ would increase the variance of the model, risking overtting. A small $C$ would, in turn, lead to possible underfitting. Considering the dual form of the optimization problem, by adding a nonlinear transformation $\varphi(\cdot): \mathbb{R}^n \rightarrow \mathcal{F}$ such that, when applied to the input vectors $\boldsymbol{x_i} \in \mathbb{R}^n$, creates a projection in a high-dimensionality feature space $\mathcal{F}$, one can create a non-linear version of the SVM classifier as

$$h(\boldsymbol{x}) = sgn\left(\sum_{x_j \in SV} \alpha_j \, y_j \, \langle \varphi(\boldsymbol{x_j}), \varphi(\boldsymbol{x}) \rangle + b\right). \tag{3}$$

In this formulation, one can observe that the decision function can be expressed solely in terms of inner products in feature space. Those inner products can then be computed through a Mercer's kernel $k(\boldsymbol{x}, \boldsymbol{z}) = \langle \varphi(\boldsymbol{x}), \varphi(\boldsymbol{z}) \rangle$. Since $\varphi$ does not have to be computed, the feature space $\mathcal{F}$ can have an arbitrarily high dimensionality.

### 4.3 Multiclass Classification Approaches

Considering its separating hyperplane formulation, the SVM is only a binary classifier, implying it can only decide between two classes at a time. Many approaches have been proposed to generalize SVMs to multiclass problems; one of the most promising being the Large-Margin Decision Directed Acyclic Graph (DDAG).

The most common approaches to multi-class classification in SVMs are the *one-vs-one* and *one-vs-all* strategies. For a decision problem over *c* classes, *one-vs-all* demands the creation of *c* classifiers, each trained to distinguish one class from the others. In the *one-vs-one* strategy, the problem is divided into $c(c-1)/2$ sub-problems considering only two classes at a time. This leaves the problem of evaluating an increased number of machines for every new instance undergoing classification – which could easily become troublesome or prohibitive in time sensitive applications. The use of DDAGs allows one to conciliate the faster training times of the *one-vs-one* strategy with evaluation speed linear to the number of classes. For $c$ classes, only $(c-1)$ machines need to be evaluated [2].

### 4.4 Hidden Markov Models

Hidden Markov Models (HMMs) attempt to model the joint probability distribution of a sequence observations $\boldsymbol{x}$ and their relationship with time through a sequence of hidden states $\boldsymbol{y}$. A HMM is described by a tuple $\lambda = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\pi})$ in which $\boldsymbol{A}$ denotes a matrix of possible state transition probabilities, $\boldsymbol{B}$ is a vector of probability distributions governing the observations and $\boldsymbol{\pi}$ is a vector of initial states probabilities. In the literature, HMMs are often described alongside with three associated canonical problems: evaluation, learning and decoding. Although we will not be discussing those in detail, a very comprehensive explanation is due to Rabiner [22].

Exploring the fact that an HMM is able to provide the likelihood for a given sequence $\boldsymbol{x}$, it is possible to create a classifier by creating a model $\lambda_i$ for each sequence label $\omega_i \in \Omega$. Treating each model $\lambda_i$ as a density model conditioned to an associated class label $\omega_i$, one can apply the Bayes' rule to obtain the a posteriori probability and then decide for the class with *maximum a posteriori.*

### 4.5 Hidden Conditional Random Fields

Conditional Random Fields (CRFs), first proposed in [23], attempt to model the conditional probability $p(\boldsymbol{x}|\boldsymbol{y})$ directly. In a general definition [16], one can consider a factor graph $G$ partitioned in a set of clique templates $\mathcal{C} = \{C_1, C_2, \dots, C_P\}$. Each clique templates $C_p$ specifies a set of sufficient statistics $\{f_{pk}(\boldsymbol{x_p}, \boldsymbol{y_p})\}$ and parameters $\boldsymbol{\theta_p} \in \Re^{K(p)}$, such that the general model for a CRF can then be written as

$$p(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{x})} \prod_{C_p \in \mathcal{C}} \prod_{\Psi_c \in C_p} \Psi_c(\boldsymbol{x_c}, \boldsymbol{y_c}; \theta_p) \tag{4}$$

with $\Psi_c(\boldsymbol{x_c}, \boldsymbol{y_c}; \theta_p) = exp\{\sum_{k=1}^{K(p)} \theta_{pk} f_{pk}(\boldsymbol{x_c}, \boldsymbol{y_c})\}$ and in which $Z(\boldsymbol{x})$ is a normalization function to keep results as probabilities. It can be seen that a CRF denotes a family of Markov Random Fields defined over $\boldsymbol{y}$ for each new observation $\boldsymbol{x}$. By choosing a specific set of features and initial values for the parameter vector, it is also possible to replicate the exact model of any given HMM, discrete or continuous. To replicate a discrete model, one could choose members from the two families of functions

$$\boldsymbol{f}_{ij}^{edge}(\boldsymbol{y_t}, \boldsymbol{y_{t-1}}, \boldsymbol{x}) = \mathbf{1}_{\{\boldsymbol{y_t}=\boldsymbol{i}\}} \mathbf{1}_{\{\boldsymbol{y_{t-1}}=\boldsymbol{j}\}} \qquad \boldsymbol{f}_{io}^{node}(\boldsymbol{y_t}, \boldsymbol{y_{t-1}}, \boldsymbol{x}) = \mathbf{1}_{\{\boldsymbol{y_t}=\boldsymbol{i}\}} \mathbf{1}_{\{\boldsymbol{x_t}=\boldsymbol{o}\}}$$

to form the feature vector and then initialize the parameter vector $\boldsymbol{\theta} = \{\boldsymbol{\lambda}, \boldsymbol{\mu}\}$ with $\lambda_{ij} = \ln a_{ij}$ and $\mu_{io} = \ln b_{io}$, in which $a_{ij}$ and $b_{io}$ are elements from the $\boldsymbol{A}$ and $\boldsymbol{B}$ matrices of discrete HMMs, respectively.

However, unlike HMMs, CRFs assume the label sequence $\boldsymbol{y}$ is known during training. One possible solution to this problem is to handle $\boldsymbol{y}$ as latent variables. By adding a variable $\omega$ to designate class labels, and setting $\boldsymbol{y}$ to be hidden, one arrives at the Hidden Conditional Random Field (HCRF) formulation, given by

$$p(\omega|\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{x})} \sum_{\boldsymbol{y}} \prod_{C_p \in \mathcal{C}} \prod_{\Psi_c \in C_p} \Psi_c(\boldsymbol{x_c}, \boldsymbol{y_c}, \omega_c;\ \theta_p) \tag{5}$$

which can be computed by the same exact algorithms used to compute $Z(\boldsymbol{x})$ in the CRF case. The penalty paid for this extra flexibility is that the optimization problem is not convex anymore, so one has to deal with the problems of local minima. Nevertheless, one can also use the same Rprop algorithm as mentioned in section 4.1 to mitigate those problems.

# 5 Experiments

## 5.1 Dataset

The data used in this study had been gathered as part of a previous work detailed in [1]. The whole dataset contains static gestures gathered from 45 subjects who articulated 27 static signs from the LIBRAS manual alphabet. It also contained dynamic gestures covering fingerspelling of 15 words. Static and dynamic gestures were stored as sequences of still images registered by a single camera in a controlled environment.

For the specific purpose of this experiment, a subset of 16,200 static gesture samples had been randomly selected from the original static sign set, with spurious or corrupted samples removed. Half of those samples were separated for testing purposes and the other half for training and validation. For the dynamic gesture set, we had 540 words containing 63,703 static signs. Hands were located from the still images using Otsu threshold with subsequent cropping and centering. The images were then dimensioned to 32x32 grayscale windows, forming vectors in $\mathbb{R}^{1024}$. Albeit not an optimal representation, this high-dimensionality approach has been done on purpose as part of the study, as it should be explained in the next section.

## 5.2 Static Gesture Recognition

The goal of this experiment is to perform a direct comparison between ANNs and SVMs without incorporating prior information into the problem (such as in the form of more elaborated features). Thus, we considered the same input dimensionality and output codes for both classifiers. For ANNs, we have created and tested multi-layer feed-forward networks with a single hidden layer and a varying number of hidden nodes using Rprop. All networks were created using bipolar sigmoid activation functions, with initial values given by Nguyen-Widrow's method. For SVMs, we used the Sequential Minimal Optimization (SMO) algorithm [24], together with the DDAG decision scheme [2] to achieve faster multi-class classification times. Parameter tuning for SVMs was performed using a coarse-to-fine grid-search (GS). We also investigated the effectiveness of using heuristic values for the Gaussian kernel based on the inter-quartile range of the input data norm statistics, as proposed by [25].

### 5.3 Dynamic gesture recognition

For the dynamic gesture recognition problem, we considered the same set of discrete features for both classifiers. After completing the previous experiment on static gesture recognition, we gathered the best static gesture classifiers to label all frames from all sequences in the dynamic gesture set. This procedure transformed our image stream data set in a more manageable set of discrete symbol sequences. We designed models containing one hidden state after each symbol in a given word. All models have been created considering the same forward-only topology for state transitions. All HMMs have been trained using Baum-Welch, and HCRFs have been trained using Rprop. Fig. 1 shows a schematic diagram of the system's architecture.

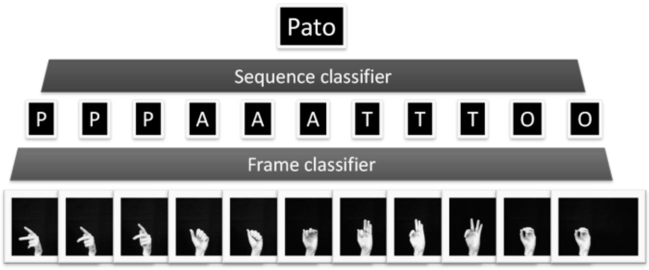


**Fig. 1.** Schematic representation for the recognition of the finger-spelled word "Pato" (Portuguese for "duck") using a two-stage classification architecture.

## 6 Results and Discussion

We start describing our results for the static gesture recognition experiment with SVMs. For the Gaussian kernel, we found a behavior similar to the one described in [26]. We found out that $C$ did not influence the performance of the classifier as much as would a proper choice for $\sigma^2$. Both the Kappa ($\kappa$) statistic and the average number of support vectors (SVs) for each classifier were mostly dependent on $\sigma^2$ rather than $C$, as shown in fig. 2. It should be worth to point out that the heuristic choice for $\sigma^2$ not only resulted in overall good performance, but also resulted in less SVs.

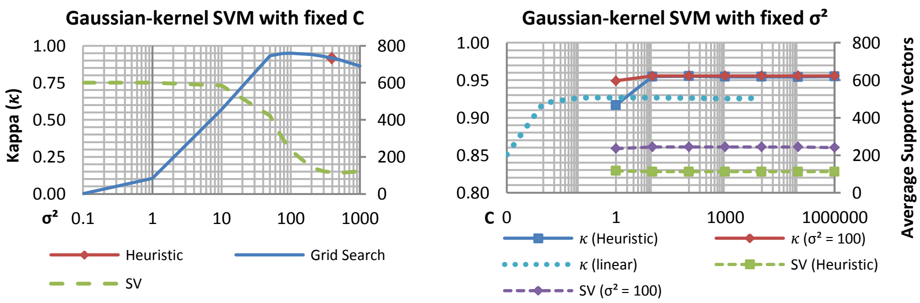


**Fig. 2.** Cuts of the grid-search procedure for Gaussian SVMs.

For the ANN experiments, we found out that ANNs could approach the same SVM's performance rates, but at huge training costs, especially considering the costs on running multiple random initializations to ensure a good local minimum. The best values for $\kappa$ were amid 300~500 neurons. However, the maximum performance obtained by ANNs ($\hat{\kappa}_{ANN} = 0.9249$) was very similar to the baseline linear SVM ($\kappa = 0.9268$) which can be seen in figure 2. Furthermore, the best SVM found ($\hat{\kappa}_{SVM} = 0.9586, \widehat{var}(\hat{\kappa}_{SVM}) = 5.10 \times 10^{-6}$) had also shown better results than the best ANN ($\hat{\kappa}_{ANN} = 0.9241, \widehat{var}(\hat{\kappa}_{ANN}) = 8.93 \times 10^{-6}$). Considering a $\kappa$ test, the differences are statistically significant under a 0.05 significance level.

The next step was to create the dynamic gesture recognizers. After obtaining the best ANN and SVMs (in terms of higher $\kappa$), we tagged the entire dataset of continuous gestures, obtaining a set of symbol sequences and their related labels. First we created HMM classifiers with the same number of symbols as letters in each word. Then we used those HMMs as initialization points to the HCRFs. All results for $\kappa$ were averaged using ten-fold cross-validation, with variance pooled from all validation runs, as shown in Table 1.

**Table 1.** Performance for all possible classifier combinations.

| **Static Gesture** | **Dynamic Gesture** | **Training** | **Validation** |
|---|---|---|---|
| | | **Kappa** ± (0.95 C.I.) | **Kappa** ± (0.95 C.I.) |
| SVM | HMM | 0.9469 ± 0.0207 | 0.8192 ± 0.1063 |
| SVM | HCRF | 0.9837 ± 0.0117 | 0.8332 ± 0.1032 |
| ANN | HMM | 0.9482 ± 0.0204 | 0.8035 ± 0.1092 |
| ANN | HCRF | 0.9903 ± 0.0090 | 0.8236 ± 0.1039 |

Results shown on Table 2 allow an interesting analysis. It can be seen that, albeit being the best combination, SVM+HCRF results in the validation set are not statistically different from others, meaning we cannot reject the hypothesis that validation results are equivalent. There is not enough evidence to conclude that validation results do not lie within a common range, as can be seen by noticing the overlap of confidence intervals.

However, the same is not true for the training results. It can be seen that training statistics are indeed significant. This opens up for an interesting interpretation of the results: in this particular experiment, discriminative models offered the same degree of generalization while retaining more information about training data. Since in cross-validation there is more data in training sets than in the validation set at each run, this indicates the models were able to learn more data without compromising generalization. In other words, HCRFs have shown greater learning ability with *less overfitting*.

## 7 Conclusion

In this paper, we detailed our experiments with SVMs and HCRFs in comparison to ANNs and HMMs in the task of fingerspelling recognition. In our first experiment in the task of static gesture recognition we achieved statistically significant results favoring SVMs over ANNs. However, despite the statistical significance and the better accuracy of the SVM models, our second experiment revealed how the choice of the gesture classifier had much greater impact than any particular choice of frame classifier. For either possible choice, using HCRFs instead of HMMs as the sequence classifiers resulted in an increased capacity for retaining training information while at same time providing comparable generalization ability. As a result, classifiers combining HCRFs and ANNs presented overall good results, with equivalent or even better accuracy than models combining HCRFs and SVMs.

However, due the high-dimensionality of the input vectors, the chance for the static gesture recognition problem being approximately linearly separable was high. As linear SVMs can be written in compact form, they were found to be able to offer similar or faster speeds than ANNs while providing comparable recognition rates. From a training perspective, SVMs were also much easier to learn. Training and selecting SVMs was simpler than dealing with the possible local-minima and long training times of ANNs. The cost of performing GS over multiple parameters could have been further reduced if we had focused only on heuristic values for $\sigma^2$, which would reduce the hyperparameter tuning problem to a single univariate search over $C$.

**Acknowledgements.** We would like to express our thanks to CNPq, for partially sponsoring this project; and also thank our anonymous referees for sharing their valuable comments and views with us.

## References

1. Pizzolato, E., Anjo, M., Pedroso, G.: Automatic recognition of finger spelling for LIBRAS based on a two-layer architecture. In : Proc. of the 2010 ACM Symposium on Applied Computing, Sierre, Switzerland, pp.969-973 (2010)
2. Platt, J., Cristianini, N., Shawe-taylor, J.: Large Margin DAGs for Multiclass Classification. In : Advances in Neural Information Processing Systems, pp.547-553 (2000)
3. Starner, T.: Visual Recognition of American Sign Language Using Hidden Markov Models. Thesis (M.S.), Massachusetts Institute of Technology (1995)
4. Starner, T., Weaver, J., Pentland, A.: Real-time American sign language recognition using desk and wearable computer based video. Pattern Analysis and Machine Intelligence, IEEE Transactions on 20(12), 1371 -1375 (1998)
5. Bauer, B., Kraiss, K.-F.: Video-based sign recognition using self-organizing subunits. In: Proc. 16th Int. Conf. on Pattern Recognition (ICPR 2002), vol. 2, pp.434-437 (2002)
6. Bowden, R., Windridge, D., Kadir, T., Zisserman, A., Brady, M.: A Linguistic Feature Vector for the Visual Interpretation of Sign Language. In Pajdla, T., Matas, J., eds. : European Conference on Computer Vision, pp.391-401 (2004)
7. Dias, D., Madeo, R., Rocha, T., Bíscaro, H., Peres, S.: Hand movement recognition for brazilian sign language: a study using distance-based neural networks. In: Proc. of the 2009 Int. Joint Conf. on Neural Networks, Atlanta, Georgia, USA, pp.2355-2362 (2009)
8. Wang, S., Quattoni, A., Morency, L.-P., Demirdjian, D.: Hidden Conditional Random Fields for Gesture Recognition. In: IEEE Conf. on Computer Vision and Pattern Recognition, 2006. CVPR '06, Washington, DC, USA, pp.1521-1527 (2006)

9. Morency, L.-P., Quattoni, A., Darrell, T.: Latent-Dynamic Discriminative Models for Continuous Gesture Recognition. In: IEEE Conf. on Computer Vision and Pattern Recognition, 2007. CVPR '07, pp.1-8 (2007)
10. Elmezain, M., Al-Hamadi, A., Michaelis, B.: Discriminative Models-Based Hand Gesture Recognition. In : International Conference on Machine Vision, Los Alamitos, CA, USA, pp.123-127 (2009)
11. Mitra, S., Acharya, T.: Gesture recognition: A survey. IEEE Transactions on Systems, Man and Cybernetics - Part C: Applications and Reviews 37(3), 311-324 (2007)
12. Yang, H.-D., Sclaroff, S., Lee, S.-W.: Sign Language Spotting with a Threshold Model Based on Conditional Random Fields. IEEE Trans. Pattern Anal. Mach. Intell. 31(7), 1264-1277 (July 2009)
13. Juang, B., Rabiner, L.: Automatic Speech Recognition--A Brief History of the Technology. In: Elsevier Encyclopedia of Language and Linguistics 2nd edn. Oxford: Elsevier (2005)
14. Ng, A., Jordan, M.: On Discriminative vs. Generative classifiers: A comparison of logistic regression and naive Bayes. In Dietterich, T., Becker, S., Ghahramani, Z., eds. : Neural Information Processing Systems 2. MIT Press (2001) 841-848
15. Quattoni, A., Collins, M., Darrell, T.: Conditional Random Fields for Object Recognition. In Saul, L., Weiss, Y., Bottou, L., eds. : Advances in Neural Information Processing Systems 17, Cambridge, MA, pp.1097-1104 (2005)
16. Sutton, C., McCallum, A.: Introduction to Statistical Relational Learning. In Taskar, L., ed.: An Introduction to Conditional Random Fields for Relational Learning. MIT Press (2007)
17. Igel, C., Hüsken, M.: Improving the Rprop Learning Algorithm. In: Symposium A Quarterly Journal In Modern Foreign Literatures, pp.115-121 (2000)
18. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition. Springer (2009)
19. Huang, L.: Variable selection in multi-class support vector machine and applications in genomic data analysis. North Carolina State University (2008)
20. Michel, P., El Kaliouby, R.: Real time facial expression recognition in video using support vector machines. In : Proceedings of the 5th international conference on Multimodal interfaces, Vancouver, British Columbia, Canada, pp.258-264 (2003)
21. Cortes, C., Vapnik, V.: Support-Vector Networks. In: Machine Learning, pp.273-297 (1995)
22. Rabiner, L.: A tutorial on hidden Markov models and selected applications in speech recognition. In Waibel, A., Lee, K.-F., eds. : Readings in speech recognition. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1990) 267-296
23. Lafferty, J., McCallum, A., Pereira, F.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In : Proceedings of the Eighteenth International Conference on Machine Learning, San Francisco, CA, USA, pp.282-289 (2001)
24. Platt, J.: Sequential minimal optimization: A fast algorithm for training support vector machines. Advances in Kernel Methods and Support Vector Learning 208, 1-21 (1998)
25. Caputo, B., Sim, K., Furesjo, F., Smola, A.: Appearance-based object recognition using SVMs: which kernel should I use? In : Proceedings of the NIPS workshop on statistical methods for computational experiments in visual processing and computer vision (2002)
26. Valentini, G., Dietterich, T.: Bias-Variance Analysis of Support Vector Machines for the Development of SVM-Based Ensemble Methods. J. Mach. Learn. Res. 5, 725-775 (December 2004)