Short Papers

Sign Language Recognition by Combining Statistical DTW and Independent Classification

Jeroen F. Lichtenauer, Emile A. Hendriks, and Marcel J.T. Reinders

Abstract—To recognize speech, handwriting, or sign language, many hybrid approaches have been proposed that combine Dynamic Time Warping (DTW) or Hidden Markov Models (HMMs) with discriminative classifiers. However, all methods rely directly on the likelihood models of DTW/HMM. We hypothesize that time warping and classification should be separated because of conflicting likelihood modeling demands. To overcome these restrictions, we propose using Statistical DTW (SDTW) only for time warping, while classifying the warped features with a different method. Two novel statistical classifiers are proposed—Combined Discriminative Feature Detectors (CDFDs) and Quadratic Classification on DF Fisher Mapping (Q-DFFM)—both using a selection of discriminative features (DFs), and are shown to outperform HMM and SDTW. However, we have found that combining likelihoods of multiple models in a second classification stage degrades performance of the proposed classifiers, while improving performance with HMM and SDTW. A proof-of-concept experiment, combining DFFM mappings of multiple SDTW models with SDTW likelihoods, shows that, also for model-combining, hybrid classification can provide significant improvement over SDTW. Although recognition is mainly based on 3D hand motion features, these results can be expected to generalize to recognition with more detailed measurements such as hand/body pose and facial expression.

Index Terms—Time series analysis, face and gesture recognition, 3D/stereo scene analysis, statistical dynamic programming, Markov processes, classifier design and evaluation, real-time systems.

1 Introduction

TIME-VARIABLE signals like speech, handwriting, hand gestures, and body movements cannot be compared in a euclidean space directly because of misalignments in time. Therefore, automatic recognition of these signals is not straightforward. In recent years, successful methods for speech recognition have been thankfully borrowed and adapted for sign language recognition. However, this has been done without questioning the exact linguistic role of the dynamics in sign language or possible conflicts between optimality in time synchronization and class discrimination. Therefore, in this article, we explore the consequences and benefits of separating time synchronization from classification in sign language recognition. The downside of this separation is that any information about relative timings is lost. The advantage of separate classification on synchronized features is that it allows the use of standard classification methods with possibly higher discriminative performance.

Dynamic Time Warping (DTW) and Hidden Markov Models (HMMs) are two methods that simultaneously align signals and compute a likelihood of similarity. Therefore, they both have been applied successfully to recognize speech [1], [2], [3], [4], online [5] or offline handwriting. Currently, they are also the most used methods for recognition of gestures [6], [7], [8], [9]. Over the years, DTW has lost some interest because HMM is able to statistically model a set of samples to generalize better, while DTW is an exemplar-based matching procedure, hence usually requiring matching with a plurality of prototypes to get comparable performance, resulting in a higher computational load. Recently, however, Bahlmann and Burkhardt have shown that DTW can also be applied to train a statistical model, using "Statistical DTW" (SDTW) [5], achieving higher performance than HMM.

Since Bahlmann and Burkhardt have applied SDTW to online handwriting recognition in [5], which can be seen as a 2D gesture recognition problem, it can be expected that an improvement over HMM can also be expected when SDTW is applied to sign language recognition. Our results show that this is indeed the case. However, we further improve upon SDTW, based on our main proposition:

Proposition 1. *The maximized likelihood that results in the optimal signal warping is not the optimal conditional likelihood estimation of the signal class.*

This proposition is supported by the following lemmas:

Lemma 1. *Transition probabilities in SDTW and HMM represent prior probabilities on path shape, which is necessary for warping in case of noisy or ambiguous observation likelihoods.*

Lemma 2. *When the meaning of a sequence has invariance to time distortion, the class-conditional probability estimate of a signal should exclude path shape likelihoods.*

Lemma 1 argues that transition probabilities should be applied to find the best warp of a signal, while Lemma 2 implies that they should not be used in classification of signals with invariance to time distortions. Furthermore, warping may benefit from cues that are the same for each sign, e.g., the transition from rest to movement at the onset of a sign and from movement to rest at the end. Such cues can be highly informative for warping, but completely uninformative for classification at the same time. To reduce the dimensionality and the influence of noise, parts that are irrelevant for the meaning of a sign are often best discarded from classification.

While most spoken languages can be regarded as 1D signals (sequences of audio patterns), sign languages make use of a combination of multiple cues that can be sequential (like in speech), but also parallel, consisting of different aspects/dimensions [10], [11]. The most commonly used dimensions are hand shape/orientation, changes in hand shape/orientation, hand location, movements of hand locations, hand-hand touching, hand-body touching (mostly specific locations on the face), lip movements, facial expression, and torso/shoulder pose and movements. Furthermore, in many cases, context is essential to uniquely define the meaning of a sign.

Regardless of which components of sign language are considered, they are all part of a dynamic process, as is speech. However, that does not necessarily mean that the dynamical aspects of sign language have the same behavior and play the same linguistic role as dynamics in spoken languages. At least three important distinctions have to be taken into account. First of all, the one-dimensionality of speech makes it sequential in nature. The (relative) timing and speed of a sequence of phonemes convey a lot

• *The authors are with the Delft University of Technology, Faculty of Electrical Engineering, Mathematics, and Computer Science, Information and Communication Theory Group, Mekelweg 4, 2628 CD Delft, The Netherlands. E-mail: j.lichtenauer@imperial.ac.uk, {E.A.Hendriks, M.J.T.Reinders}@TUDelft.nl.*

Manuscript received 13 Sept. 2007; revised 18 Feb. 2008; accepted 5 May 2008; published online 13 May 2008.
Recommended for acceptance by A. Martinez.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2007-09-0596.
Digital Object Identifier no. 10.1109/TPAMI.2008.123.

of the meaning in a word. On the contrary, sign language is composed of many parallel components. Because of this richness in dimensionality, it is possible to vary speed and timing significantly without changing the message [10], [12]. Second, for most signs, only a subset of the degrees of freedom are important. However, this can be a different set for different signs. Furthermore, motion path features like motion orientation and curvature are not defined during a standstill, which will result in extremely noisy values. Third, the moments before the actual stroke of a sign (the preparation), after finishing the sign (retraction), or in between different signs or parts of signs (transition) are not essential for recognition. These parts are either irrelevant or redundant [13]. However, they cannot simply be detected and excluded like silences in speech or pen-off periods in handwriting. The above can be summarized by the following observations:

Observation 1. *Because of the high dimensionality of sign language, time is relatively less important for the meaning of a sign than it is for a spoken word or written letter.*

Observation 2. *Signs in sign language are defined on (different) subsets of a person's degrees of freedom and can vary greatly on the other dimensions without any change of meaning.*

Observation 3. *Preparations, retractions, and transitions in sign language cannot be removed beforehand, unlike silences in speech or pen-off periods in handwriting.*

If Observation 1 is true, it may result in larger deviations of gesture speed and timing, and it also implies that the consequences of Lemma 2 have to be considered. Observation 2 implies that sign-specific feature selection would be necessary for good discrimination, which would have to be done after synchronization, just like removal of the irrelevant or redundant segments indicated by Observation 3.

The important consequence of Proposition 1 is that warping and classification of time-variable signals should be regarded as two distinct problems, instead of naively incorporating it into one integral Bayesian model. Observations 1 to 3 imply that this holds in particular for sign language. Therefore, we use SDTW only to warp a signal onto a reference model, and regard the time-normalized signal as a fixed-size feature set. To remove irrelevant and redundant parts and dimensions, we apply robust statistics to select only discriminative features (DFs). The proposed method is computationally attractive, as time warping is solved by dynamic programming, and the classification step is even significantly less costly. Our experiments are limited to hand-motion trajectories and apparent hand-size change in isolated signs. This is because these are the few components that the current state of the art in human motion analysis allows to track in reasonably soft-constrained situations without manual initialization of tracking. Therefore, they are currently the most relevant properties for practical applications. We assume that if information about (relative) timing can be disregarded for classification when only these properties are used, this will certainly be the case if even more parallel aspects (e.g., detailed hand/body pose and facial expression) are considered, which would be inevitable to obtain perfect recognition [14].

2 Related Work

We are not the first to combine a variable-time signal match, like DTW/HMM, with fixed-vector-size mappings or classifiers in order to improve results. Previous approaches can be roughly divided into methods that apply DTW/HMM on mappings of the fixed-size measurement vectors of all time frames (to get a more informative observation likelihood) and methods that use the results of a fixed number of different DTW/HMM evaluations as the input of a second-stage classifier. In [15], a Multilayer Perceptron provides estimates of the emission probabilities for all phonemes of speech, subsequently used for matching a HMM. In [16], a Neural Network classifies the measurements of separate frames into a first and second guess of a speech phoneme, and a DTW match uses the phoneme matches with a template word as a distance measure. In [17], the measurements for a frame of a gestured command, recorded by a camera, are converted into a probability estimate of each state by a Radial Basis Function network. The resulting state emission probabilities are used for an HMM. In [18], Chinese sign language is measured with data gloves. Signs that are not well separated by HMM alone are classified in an extra recognition step by a Support Vector Machine (SVM) using a DTW kernel. In [19], a sequential HMM is trained for each hand gesture measured from two cameras. The HMM match result is split into five components which are used as features for a multiclass SVM classifier, trained by applying one HMM to all training gestures. The final classification is obtained by majority voting of the results of the HMM/SVM pairs for all gesture classes. A similar approach is chosen in [20] to classify online hand writing characters. Instead of using HMM, here, SDTW is used as a kernel for SVM.

The above works confirm that results can be improved over HMM/DTW alone. However, all methods have relied directly on the likelihoods obtained from DTW or HMM. Instead, we consider DTW/HMM primarily as a registration method. We use the complete set of registered features as a richer sign representation instead of, or in addition to, the outputs of HMM/DTW. This approach may even be combined with mappings of input vectors per frame as well, although this is beyond the scope of this article.

Alon et al. [21] have proposed Dynamic Space Time Warping (DSTW), which considers multiple possible 2D hand locations in each frame. This reduces the consequences of imperfect tracking. Although our experiments use single-hypothesis 3D tracking, the principle of separating warping and classification may easily be extended to DSTW. One advantage of our approach is that the negative influence of irrelevant variations in preparations, transitions, and retractions can be reduced by applying feature selection on the registered feature set. Instead, Yang et al. [22] have resolved this problem by including a separate model with constant distance that is fitted to sequences that do not fit well to any known sign. The disadvantage is that it introduces the possibility of falsely inserting the transition model in the place of a sign that differs more from its model than is accounted for.

3 Static Dynamic Time Warping

STDW was first introduced in [5]. A description of DTW is given in [23, Section 4]. DTW compares each test signal $t = [\mathbf{t}_1, \ldots, \mathbf{t}_{N_t}]$ to a stored reference $r = [\mathbf{r}_1, \ldots, \mathbf{r}_{N_r}]$. The difference between SDTW and normal DTW is that, instead of comparing a test signal t to a reference signal r, the reference $\mathcal{R} = [\mathcal{R}_1, \ldots, \mathcal{R}_{N_\mathcal{R}}]$ in SDTW is not a signal but a statistical model consisting of a Normal distribution for each time point j with mean $\boldsymbol{\mu}_j$, covariance matrix $\boldsymbol{\Sigma}_j$, and transition probabilities $\alpha_j(\Delta\phi)$: $\mathcal{R}_j = \{\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j, \alpha_j(\Delta\phi)\}$. $\Delta\phi \in \mathbb{P}$ is a transition of the warping path to a point with state j, where $\mathbb{P}$ are the possible transitions from $(\phi_t(n-1), \phi_\mathcal{R}(n-1))$ to $(\phi_t(n), \phi_\mathcal{R}(n))$.

The matching cost $C^*(t, \mathcal{R})$ is defined by

$$C_\Phi(t, \mathcal{R}) = \frac{\sum_{n=1}^{N} d\big(\mathbf{t}_{\phi_t(n)}, \mathcal{R}_{\phi_\mathcal{R}(n)}\big) w(\mathbb{P}(n))}{\sum_{n=1}^{N} w(\mathbb{P}(n))}, \tag{1}$$

$$C^*(t, \mathcal{R}) = C_{\Phi^*}(t, \mathcal{R}) = \min_{\Phi} C_\Phi(t, \mathcal{R}), \tag{2}$$

where $\Phi = \{\phi_t(1), \ldots, \phi_t(N_\phi), \phi_r(1), \ldots, \phi_r(N_\phi)\}$ are the steps of the path through the 2D correspondence matrix $\mathbf{C}(t, \mathcal{R})$ of the time frames of t and $\mathcal{R}$, $\mathbb{P}(n)$ is a transition step between two subsequent points of the path Φ through $\mathbf{C}(t, \mathcal{R})$, and $w(\mathbb{P}(n))$ is a function that assigns a weight to a transition type. $C^*(t, \mathcal{R})$ is also used as the final match cost. Some constraints of the path are implied to confine the minimization procedure to practical results. The most common constraints are that the path starts in $\Phi(1) = (1, 1)$ and ends in $\Phi(N_\phi) = (N_t, N_r)$, and the set of possible transitions $\mathbb{P}$ is limited to

$$\begin{aligned}\mathbb{P}(n) &= [(\phi_t(n+1) - \phi_t(n)), (\phi_r(n+1) - \phi_r(n))] \\ &\in \{[0, 1], [1, 0], [1, 1]\},\end{aligned} \tag{3}$$

corresponding to horizontal, vertical, and diagonal steps, respectively. The distance function $d(\mathbf{t}_i, \mathcal{R}_j)$ is defined as the inverse log likelihood:

$$\begin{aligned}d(\mathbf{t}_i, \mathcal{R}_j) = \frac{1}{2}\Big(\ln\big(|2\pi\Sigma_j|\big) + (\mathbf{t}_i - \boldsymbol{\mu}_j)^T \Sigma_j^{-1} (\mathbf{t}_i - \boldsymbol{\mu}_j)\Big) \\ - \ln\big(\alpha_j(\Delta\phi)\big).\end{aligned} \tag{4}$$

Equation (2) is approximated efficiently using dynamic programming by the omission of the denominator in (1) in the choice of subpaths. The denominator is applied only on the finally chosen path. The transition weighting function $w(\mathbb{P}(n))$ can be chosen so that all possible subpaths leading to one location in $\mathbf{C}(t, \mathcal{R})$ have an equal sum of weights: unbiased. In that case, the approximation of (1) by dynamic programming is exact. In [3] and [4], it is explained how to obtain such unbiased weighting functions. We will use the (most used) biased method that assigns $w = 1$ to all three transition types. Instead of a solution of (2), this weighting gives preference to more diagonal, shorter paths. According to Lemma 1, a bias toward more linear paths may actually be an advantage, as it acts as a path shape prior in case of noisy measurements.

The biased SDTW, as defined above, is equivalent to a forward HMM with self-transitions and no skips, to which "null-transitions" are added. The null-transitions can be used to step to a next state (or the same state) without advancing in time. This allows unlimited compression of the model in time. The damage of a missing part can be limited to the missing part only (like one less repetition of a repetitive motion or an extremely high signing speed causing a significant reduction of time points), while, with a regular HMM, an observation is assigned only once to any state instance, causing the left-out part of the trained HMM to steal away observations belonging to other surrounding states and limiting the amount that a sign can be compressed in time.

An SDTW model $\mathcal{R}$ is trained on a set of examples by iteratively warping all training samples with an initial model $\mathcal{R}$ and reestimating each $\boldsymbol{\mu}_j$, Σ_j, and $\alpha_j(\Delta\phi)$ from the aligned observations [5]. Note that, similar to a Markov Model, the transition probabilities $\alpha_j(\Delta\phi)$ at step n only depend on $\phi(n)$ and the previous $\phi(n-1)$. However, a gap or insertion in a sign t (e.g., fewer or more repetitions in a repetitive motion) requires a number of subsequent repetitions of a time frame of t or a state of $\mathcal{R}$, respectively, while, otherwise, steps in both are required (more or less diagonal path). Therefore, the memory-less assumption of transitions does not hold for gaps and insertions.

4 Classification

(S)DTW (or fitting an HMM) finds the best hidden sequence of a specific model in another sign by maximizing the likelihood of the observation over possible time synchronizations. A limitation of a SDTW/HMM likelihood model is that the observation likelihood is modeled independently per state/frame, usually by mixtures of Gaussians, and modeling of interframe dependencies is limited to these observation likelihoods. Furthermore, the same feature types are used for all signs and frames, even though the relevance of these measurements may vary significantly between signs and frames. Our proposed Combined Discriminative Feature Detectors (CDFD) classifier, explained in Section 4.2, not only applies feature selection but also uses an alternative likelihood model that overcomes shortcomings of the independence assumption. In Section 4.3, we propose another classifier, Quadratic Classification on DF Fisher Mapping (Q-DFFM), that works in the joint feature space of all selected features of all frames together. But first, the next paragraph describes a robust method to discard non-DFs that is used in both proposed classifiers.

4.1 Discriminative Feature (DF) Selection

Following Observations 2 and 3, we expect that recognition would greatly benefit from leaving out segments and dimensions completely from classification, if they are irrelevant or do not differ between sign classes. The features used for classification are reduced to a set of DFs by a robust statistical test. A feature $f_j(m)$ of type m (see Table 1 in the supplemental material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2008.123) corresponding to the synchronized time frame j is selected for classification only if the middle 50 percent of its distribution (between the 0.25 and 0.75 quantile) over the set χ_p of training examples of the correct sign (positive examples) has an overlap of less than 25 percent with the distribution of the set χ_n of training examples of incorrect signs (negative examples).

4.2 Combined Discriminative Feature Detectors (CDFD)

After feature selection, a relatively large number of features still remains (around 500 selected out of ~1,900), while it is difficult to obtain a large multisigner training set (variation of a single signer does not generalize well to others). Because of the curse of dimensionality, we assume independence between features. The classification is based on assuming an independent Normal distribution $L(\hat{t}, \mathcal{R}, j, m)$ of each feature type m in reference frame j (1D):

$$\begin{aligned}L(\hat{t}, \mathcal{R}, j, m) &= \ln\big\{p\big(\hat{t}_j(m) | \mathcal{R}_j(m)\big)\big\} \\ &= -\frac{1}{2}\left(\ln\Big(2\pi\sigma_j^2(m)\Big) + \frac{\big(\hat{t}_j(m) - \mu_j(m)\big)^2}{\sigma_j^2(m)}\right).\end{aligned} \tag{5}$$

Usually, the feature log likelihoods, computed in (5), would be naively combined by their sum. However, this would result in a low likelihood of a sloppy but completely correct sign. Using a strictly statistical approach, this problem can only be solved by accounting for dependence between frames, which is difficult with a small training set. To overcome this problem, CDFD first converts the feature likelihood distributions to piece-wise uniform functions, which can be seen as Feature Detectors (FD):

$$q(\hat{t}, \mathcal{R}, j, m) = \begin{cases} 1, & \text{for } L(\hat{t}, \mathcal{R}, j, m) \geq T_j(m) - T_g, \\ 0, & \text{for } L(\hat{t}, \mathcal{R}, j, m) < T_j(m) - T_g, \end{cases} \tag{6}$$

where T_g is the gauge parameter that will determine the operating point of the final classifier and $T_j(m)$ is the calibrated threshold that accepts 90 percent of the positive training data for a particular feature at $T_g = 0$. The choice of 90 percent as the calibration point is a trade-off between generalizing (including the complete range of allowed variation of a feature) and the expected reliability of the training set (rejecting outliers). The training set contains tracking errors and signs that were not performed well enough to be correct. We assume that these errors are below 10 percent for all selected features. Excluding the outer 10 percent of the positive distribution should eliminate the influence of these outliers in

determining the default boundary of allowed variation. We have added an experiment in [23, Section 2] that shows the sensitivity to the choice of the acceptance rate for $T_j(m)$. Decreasing the acceptance rate with 10 percent results in a decrease of approximately 0.5 percent in the partial Area Under the Receiving Operator Characteristic (ROC) curve between 0 and 0.1 false positive rate $(\mathrm{pAUC}_{0.1})$.

With $q(\hat{t}, \mathcal{R}, j, m)$, all outliers outside of the allowable variation are penalized equally with a score of 0, no matter how great their distance to the mean feature value. Likewise, all variation inside the allowed interval gets the same score of 1. This makes it possible to accept sloppy but completely correct signs (e.g., signs that are made smaller than usual), while rejecting incorrect signs that are very similar to a subset of the feature models (e.g., incomplete signs).

The classifier output is generated by

$$Q_{\mathcal{R}}(t) = \sum_{j=1}^{N_{\mathcal{R}}} \sum_{m=1}^{N_m} s_j(m) q(\hat{t}, \mathcal{R}, j, m), \tag{7}$$

where $s_j(m)$ is 1 for selected features (from Section 4.1) and 0 otherwise, and N_m is the number of feature types, equal to 25 (see Table 1 in the supplemental material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2008.123). A sign is classified by

$$C_{\mathcal{R}}(t) = \begin{cases} correct, & Q_{\mathcal{R}}(t) \geq T_C, \\ incorrect, & Q_{\mathcal{R}}(t) < T_C, \end{cases} \tag{8}$$

where T_C is the value that classifies 50 percent of the positive training set correctly at $T_g = 0$ (median of $Q_{\mathcal{R}}$). T_C determines the fraction in time that a sign needs to be correct, hence allowing for some tracking errors or small human errors and/or hesitations in making the correct sign. Because, in our application, a sign has to be made correct from beginning to end, T_C is kept fixed. Instead, T_g determines the allowed variation and is adapted to the allowed sloppiness (the operating point).

4.3 Quadratic Classification on DF Fisher Mapping (Q-DFFM)

Instead of combining independent feature detectors, Q-DFFM estimates a statistical model of a sign class that includes dependencies between features and time frames. To overcome the curse of dimensionality, the dimensionality of the DF set is reduced by Fisher mapping [24], which is a form of Linear Discriminant Analysis (LDA). The final classifier should distinguish between only two classes ("correct" and "incorrect"). However, the incorrect class of the training set is composed of many different sign classes. This fact can be exploited by finding a set of projections of DF that optimally separates all different sign classes. It can be expected that such a mapped space captures information that is generally useful to distinguish different sign classes. The Fisher mapping attempts to find the best linear separation between each class and the other classes. Projecting the initial feature space onto the separating directions for all classes results in a $N_C - 1$ dimensional feature space, with N_C being the total number of sign classes in the training set. When it is not possible to separate all classes with the provided measurements, some of the Fisher dimensions will be useless. Therefore, only the most discriminating $N_F \leq N_C - 1$ dimensions are used. Note that any $N_F < N_C - 1$ results in loss of optimality for separating all classes [25]. In [25], a method is proposed to regain optimality. This may lead to better performance, although our dimensionality reduction is meant for nonlinear separation of the target class instead of linear separation of all classes. The entire training set is mapped by the (subset of the) Fisher mapping, on which a quadratic (Gaussian) two-class classifier is trained. The target sign class is one of the two classes, while all examples of other sign classes in the training set are merged into a single "background" class. The likelihood ratio between the two estimated Gaussian distributions is the final classifier.

5 Experiments

Sign classification is evaluated on a set of 120 different signs of the Dutch Sign Language (DSL), each performed by 75 different persons. The images are captured at 640 × 480 pixels and 25 fps. Most sign examples contain partial occlusions of hands with each other or with the face/neck. A description of the system setup is given in [23, Section 5].

The data that are used for recognition consist of estimates of 3D locations of both hands over time, plus the size of the segmented hands in the image. They are measured from the images of a calibrated stereo camera. The image analysis procedures to extract the measurements from the camera output are described in [23, Section 6], and the supplementary video "3DTracking.avi," which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2008.123. From these measurements, a set of 25 higher level features are extracted. Nine measurements consist of hand coordinates relative to the face or to the other hand, 14 describe the motion of the hands, and two correspond to the size changes of the segmented hands. To reduce variation due to signer speed, the features corresponding to change are soft-thresholded. As the average sign length is around 3 seconds, or 75 frames, the total number of features can be more than 1,500 per sign. The features obtained for each video frame are described in Section 7 of the supplemental material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2008.123. Note that hand motion features are only a small fraction of all the cues that define meaning in sign language. Other cues (like hand shape, facial expression, and context) are often static during a sign stroke. The dynamic motion features are expected to be more affected by Observation 1 than other features. It is assumed that better modeling of dynamic components alone will also lead to improved performance when other cues are considered as well.

We consider two different scenarios of sign language recognition: target-class (two-class) and multiclass classification. In target-class classification, the target class is the correct sign, while examples of other signs form a second (background) class. Although multiclass classification is mostly chosen for research, it is not practically feasible when the number of classes becomes large, or multiple classes cannot be distinguished based on the measured features (full overlap). Furthermore, a rejection step for unknown classes is often omitted. In practice, rejection of unknown classes (incorrect gestures that can be anything) is one of the most important requirements. Moreover, discriminating between known sign classes is sometimes even undesirable in case of full overlap in the measured features. Forcing an algorithm to discriminate between two indistinguishable signs will deteriorate recognition performance for both.

A target-class classification experiment consists of 120 fivefold cross-validations. One test run consists of training and testing a classifier that should distinguish one specific sign class (denoted as the "positive" class) from any other sign or gesture (negative class). Because the purpose of target-class classification is to reject unseen classes, the 120 sign classes were split into 96 training classes and 24 test classes. Only the target class has examples both in the training set (60 examples) and the test set (15 examples). Whenever a target class is one of the assigned training classes, only the remaining 95 nontarget classes are used for training, and when the target class is one of the assigned test classes, only the remaining 23 nontarget classes are used for testing. For Q-DFFM, the number of dimensions N_F of the Fisher mapping is optimized by maximizing the average partial Area Under the ROC curve

TABLE 1
A Comparison between Standard HMM and SDTW

a	40 state HMM	84.61%
b	40 state SDTW trained as HMM	87.57%
c	40 state SDTW	90.60%
d	~74 state SDTW	90.54%

The measures are the average percentages of $\mathrm{pAUC}_{0.1}$ *of the ROC curves.*

between 0 and 0.1 false positive rate $(\mathrm{pAUC}_0.1)$ of a sixfold cross-validation on the training set.

Note that the classifier is tested only on negative classes that it has never seen before. Unlike multiclass classification, the performance in this test will not necessarily decrease with a larger number of classes, as the classifier is tested as a one-against-all (two-class) classifier. The performance may even increase if more classes are added to the negative training set, as it will improve generalization.

To test target-class classification in Multiple Class Likelihood (MCL) space, the samples of the negative training classes for each model were also split into 60 training samples and 15 test samples, just like the positive set, to prevent the positive test samples of a target class from being used as negative samples for training the single-class models of other classes.

In the target-class case, classifier performance can be evaluated by the ROC curve that shows all possible operating points. One point on the ROC curve denotes the false positive error rate with the corresponding true positive rate for a specific classification threshold. The area under the ROC curve is averaged over all cross-validations and positive sign classes to obtain a total score. The larger the area, the better. As we are only interested in the operating points with realistic (usable) results, only the ROC curve partial AUC between a false positive rate of 0 and 0.1 are considered $(\mathrm{pAUC}_{0.1})$.

Multiclass classification was also tested in fivefold cross-validation, using a feature space of all 120 single-class models, trained on 60 and tested on 15 samples of each class. The performance is evaluated by the average of the classification error rate in each cross-validation.

5.1 SDTW Outperforms HMM

In the first experiment, we compared SDTW to HMM. Results can be seen as the pAUCs in Table 1. The HMMs have 40 states and Bakis topology (left to right with single-state skips and self-transitions). The length of the HMM is a trade-off between modeling detail and the minimum length (maximum speed) that can be recognized. This HMM size is comparable to [26], where the average length was 41 at the same frame rate of 25 fps. The HMMs are trained with Baum-Welch, but evaluated using the Viterbi algorithm. In SDTW, the transition probabilities were not used (assumed equal). To see the influence of different aspects of SDTW, the HMM is converted to full-scale SDTW in three steps. First ("b" in Table 1), the trained HMM models are evaluated as SDTW by using their state means and covariances. This already gives a significant performance improvement over "a." Apparently, HMM really suffers from the rigid warping constraints. In "c," the training is also done using SDTW. This results in a comparable performance increase over "b" as "b" had over "a." This is not so surprising, as the same warping restrictions of HMM are expected to be a limitation during training as well. The third step, "d," increases the length of the SDTW model to the average length of the positive training examples. This is not possible with HMM since an HMM can never have more states than the number of frames of the smallest sequence. However, no significant performance change can be observed due to increasing the number of states (p-value 0.52 in a paired t-test of the pAUCs). This is because the DSL signs do not have as many different details as the number of frames recorded here (25 per second). Therefore, multiple frames can be modeled with the same state. Adding more states does not provide better modeling accuracy.

TABLE 2
A Comparison between Different Ways of Handling Transitions in SDTW

a	SDTW + trans. prob. in warp only	90.54%
b	SDTW + trans. prob. in warp & likelihood	90.27%
c	unbiased SDTW	88.14%
d	cityblock SDTW	87.81%
e	unbiased SDTW + trans. prob. in warp only	87.69%

The measures are the average percentages of $\mathrm{pAUC}_{0.1}$ *of the ROC curves.*

5.2 Transition Probabilities Are Questionable

To test the influence of transition probabilities in SDTW, several possibilities of using transition probabilities have been compared. According to Lemma 2, we expect a negative effect of using transition probabilities in the likelihood computation. However, according to Lemma 1, we expect a positive effect of transition probabilities for the warping itself. The results in Table 2 are only partially consistent with our predictions. As predicted by Lemma 2, using transition probabilities in the class-likelihood computation resulted in a decrease of performance ("b" versus "a"). Although small, the difference with using transition probabilities only in warping was significant with $\text{p-value} = 8 \times 10^{-22}$. Note that we cannot be certain if this performance decrease is due to Lemma 2 or because of a poor modeling of transition probabilities (e.g., the memory-less assumption).

Adding transition probabilities only in the warping step ("a" in Table 2) had no effect compared to SDTW without transition probabilities ("d" in Table 1). The p-value was 0.74. This might be explained from the huge scale difference between transition probabilities and the observation likelihoods. Because of the high-dimensional Gaussian models, likelihoods can differ so much that they totally dwarf the influence of the transition probabilities. However, there will probably also be cases where the differences of the observation likelihoods are not so large, so that still does not explain why no difference can be seen at all. Another explanation may be that the bias toward short paths in the SDTW warping may overrule the influence of the transition probabilities. To test this hypothesis, the experiments are repeated using unbiased warping ("e" in Table 2). Although, now, the transition probabilities in warping indeed have an influence ($\text{p-value} = 9 \times 10^{-6}$ compared to "c"), the effect is opposite from what was expected. The transition probabilities actually have a negative influence on warping instead of positive, suggesting poor modeling capability. Apparently, the standard use of transition probabilities in SDTW is questionable. On the contrary, the bias in SDTW does provide an important positive effect on performance (Table 1 "d" versus Table 2 "c"). This is because it gives preference to diagonal transitions, and therefore shorter, more linear paths. Linear warping is mostly the best "guess"/prior when observation likelihoods are inconclusive; hence, this result is in support of Lemma 1. City block STDW (Table 2 "d") is also unbiased because the diagonal transition is omitted. It performs a little worse than unbiased SDTW because the extra diagonal step provides more freedom to avoid low observation likelihoods.

5.3 Hybrid Approach Best for Single-Model Target Classification

First, we have compared the hybrid methods to SDTW and HMM when a single model is used. The results are shown in the "single-model" column of Table 3. Besides our proposed CDFD and Q-DFFM methods ("d" and "e", respectively), we have also tested a Fisher classifier ("f") and SVM with radial basis function kernel ("g"). However, for these methods, we have also applied our DF selection, since, otherwise, the dimensionality was too high. The

TABLE 3
Results for Target-Class Classification Using a Single Model or Using Three Different Methods in MCL Space

	Method	single model	MCL Fisher	MCL QDC	MCL SVM
a	HMM	84.61%	**96.97%**	95.94%	91.71%
b	SDTW	90.54%	**97.22%**	96.29%	95.93%
c	SDTW+CDFL	95.35%			
d	SDTW+CDFD	95.46%	90.86%	74.17%	**92.86%**
e	SDTW+Q-DFFM	*96.62%	94.84%	91.17%	**95.31%**
f	SDTW+DF-Fisher	92.70%			
g	SDTW+DF-SVM	95.29%			
h	HMM+CDFD	91.03%			
i	HMM+Q-DFFM	95.73%			
			MCL&F Fisher	MCL&F QDC	MCL&F SVM
j	SDTW&DFFM5		**97.50%	-	-

The measures are the average percentages of $\mathrm{pAUC}_{0.1}$ of the ROC curves.

Combined Discriminative Feature Likelihood (CDFL) method ("c") is an intermediate step between the likelihood computation in SDTW and CDFD. Instead of Gaussian modeling of all features per frame together, as in SDTW, all features are modeled independently by a 1D Gaussian, and only the selected features are used. The result of CDFL is significantly better than SDTW ("b"). This is in accordance with Observation 2, which implies that a lot of measurements can be discarded to improve performance.

The benefit of a hybrid approach is clearly visible, with the best result for Q-DFFM ("e" $*$). The result for Q-DFFM is even better than CDFD ("d") with $\text{p-value} = 5 \times 10^{-9}$. However, the practical advantage of CDFD is that the operating point can be set more intuitively. The threshold is directly proportional to the allowed variation in the selected measurements. Although this may not be a theoretical advantage, in practice, setting the operating point of a classifier is a problem in itself. An example that could be recognized by Q-DFFM but not by SDTW is shown in Fig. 1. Especially, the ending (retraction) differs a lot, but is irrelevant for recognition. Two other examples are shown in [23, Section 3], and the original videos are also included, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2008.123.

Furthermore, we have also tested combinations of $\mathrm{HMM} + \mathrm{CDFD/Q\text{-}DFFM}$ ("h" and "i") to see if the proposed limitations of HMM are really a problem. Indeed, the results are worse when compared to the same classification methods combined with SDTW warping ("h" versus "d" and "i" versus "e"). These differences can be due to the rigid HMM warping and/or the lower number of states in HMM (40), which is a consequence of the warping rigidity.

5.4 Hybrid Model Combining Requires Different Approach

So far, we have classified a target class by its likelihood using the sign's own SDTW model. Combining the outputs of multiple models is a common practice to increase performance. Therefore, in this experiment, we concatenate the likelihood output of the target-class classifier for the actual target class and the outputs of the target-class classifiers trained for all 95 or 96 (depending on the cross-validation step) nontarget classes (as they were trained for the experiments in Section 5.3). This forms a 96- or 97-D MCL feature space in which a second-stage classifier can be trained. This is done with the likelihood estimations of HMM, SDTW, $\mathrm{SDTW} + \mathrm{CDFD}$, and $\mathrm{SDTW} + \mathrm{Q\text{-}DFFM}$. We applied three different classifiers in MCL space: Fisher, Quadratic (Gaussian) Discriminant Classifier (QDC), and linear SVM. The results for target-class classification in MCL space are shown in the last three columns of Table 3.

While HMM and SDTW have improved by combining multiple likelihood models ("a," "b"), the hybrid target-class classification methods have decreased in performance ("d," "e"). This is probably because the single-model hybrid classifiers are too specialized for discrimination of one class. CDFD likelihood is meaningful only for signs of which at least T_C of DF are similar to the target sign. The Q-DFFM likelihood is a linear separation with the target sign on one end and all the other signs on the other. Therefore, the MCL dimensions corresponding to nontarget classes may contain less information about the real target class for CDFD and Q-DFFM than HMM and SDTW models do. A change of HMM/SDTW likelihood for the target-class model, due to an allowable sign variation, may strongly correlate to changes of likelihood for models of other classes. These relations can be exploited by the second-stage classifier. Furthermore, the single-model hybrid approaches are more vulnerable to errors in the warping of the single SDTW model.

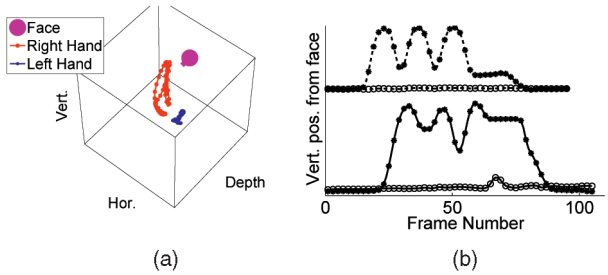

Fig. 1. Example of the sign "to chop," which is detected by Q-DFFM but not by SDTW. (a) The trajectory in 3D space and (b) the relative height of both hands against time. The instructor's sign is shown on top and the test sign below. The flat lines correspond to the hand that was not used in this sign.

Contrary to the Q-DFFM output, DFFM does contain information that can be useful for other classes, for its 96 (or 97) dimensions consist of linear separations for all sign classes in the training set based on the alignment to the target-class SDTW model. The DFFM mappings for the $N_C - 1 = 95$ (or 96) SDTW models in the training set could be combined in a single 9,120- (or 9,216-) D feature space that would be highly correlated. However, this high dimensionality poses a computationally complex optimization problem which is beyond the scope of this article. Instead, as a proof of concept for this multimodel hybrid approach, we have expanded the SDTW MCL space with the five best separating dimensions of each DFFM mapping of SDTW-synchronized features. This results in a $(96 \text{ or } 97) \times (1 + 5) = 576$- (or 582-) D feature space. The results for a Fisher classifier in this MCL&F space is shown in Table 3 ("j"). QDC and SVM were not able to run at this data size. Despite the 500 percent increase of dimensionality, the performance with the added information has increased from 97.22 percent to 97.50 percent with a p-value of 0.009.

5.5 Multiclass Classification Not Suitable to Detect a Target Class

Multiclass classification is performed in the exact same way as the model combining for target-class classification in Section 5.4. Only, now, the second-stage classifier is trained as a multiclass classifier combining the outputs of single-model target-class classifiers for all 120 sign classes. This gives five recognition rates (corresponding to the cross-validations) for classifying a sign as 1 of 120 classes. The average rates and standard deviations are shown in Table 4. Since this is a multiclass problem, Nearest Neighbor (NN) is used instead of SVM. Because there are three pairs and two triplets of sign classes that cannot be distinguished by motion alone, the maximum achievable recognition rate by using motion alone is 94.1 percent. If hand shape would be added, only one ambiguous pair would remain, raising the limit to 99.2 percent. It cannot be expected that this rate can be achieved using the 2D hand size change as the only hand shape feature.

Remarkably, HMM ("a") now performs better than SDTW ("b"). However, the difference is practically 0 $(\text{p-value} = 0.94)$. The single-model hybrid methods underperform in the combined space, just like when combined for target-class classification. Again, the hybrid model-combining method SDTW&DFFM5 ("e")

TABLE 4
Classification Accuracy for Multiclass Classification Using Three Different Discriminants in MCL Space

	Method	MCL Fisher	MCL QDC	MCL NN
a	HMM	**90.8%(σ2.0)**	70.3%(σ1.5)	76.2%(σ4.3)
b	SDTW	**90.8%(σ1.4)**	79.2%(σ1.7)	80.0%(σ3.6)
c	SDTW+CDFD	76.0%(σ2.89)	**36.6%(σ6.6)**	43.0%(σ3.6)
d	SDTW+Q-DFFM	83.7%(σ1.1)	**84.3%(σ2.2)**	80.0%(σ2.7)
		MCL&F Fisher	MCL&F QDC	MCL&F NN
e	SDTW&DFFM5	*__92.3%(σ1.2)__	-	82.9%(σ2.6)

The measures are the average percentages of correct classification.

results in a significant improvement over SDTW with an average accuracy of 92.3 percent. One of the five cross-validations even reached a recognition rate of 93.7 percent, which comes close to the maximum of 94.1 percent that can be achieved with hand motion alone. The improvement over HMM outputs combined with Fisher has a p-value of 0.019.

Since the result of multiclass classification can be used to detect a target class, we can compare multiclass classification with the target-class classifiers described above. The SDTW&DFFM5 Fisher-combined multiclass classifier would erroneously recognize a random sign of an unseen class as the target class at a false positive rate of 1/120, while a target sign would be recognized correctly 92.3 percent of the time. For the best target-class classifier (SDTW&DFFM5, Table 3 "j"), the true positive rate at a false positive rate of 1/120 is 93.0 percent on average. This is significantly higher, with p-value 0.054 over the five cross-validations.

6 Conclusion

We have proposed and evaluated a hybrid approach to sign language recognition by using SDTW only for time warping and a separate classifier on the warped features. One of the main advantages of this approach is that non-DFs can be discarded to reduce dimensionality and noise. This is especially important in sign language, as signs are often constrained only within a subset of all possible degrees of freedom. The two single-model classification methods we proposed (SDTW + CDFD and SDTW + Q-DFFM) both significantly outperform SDTW by itself in target-class classification.

We have also confirmed that SDTW provides a significant improvement over HMM because of the warping rigidity in HMM. However, we have observed that transition probabilities in SDTW provide a poor prior on DTW path shape and can even decrease recognition performance. On the other hand, the DTW warping bias, introduced by not compensating for the increased length of nondiagonal transitions, actually improved performance, acting as a prior on path shape with preference for shorter, more linear paths.

Furthermore, we have found that when a second-stage classification on the likelihood outputs of multiple target-class classifiers is applied, results from multiple SDTW or HMM models improve, while the hybrid methods degrade. We have shown that a successful model-combining hybrid method can be obtained by including the DFFM mappings for separate SDTW models in the feature space for the second stage, in addition to SDTW likelihoods. This resulted in a significant improvement over HMM and SDTW both in target-class classification using combined models and in multiclass classification.

Although recognition relied mainly on 3D hand motion features, it can be expected that these results generalize to more detailed measurements such as hand/body pose and facial expressions.

Acknowledgments

This work was done in collaboration with the NSDSK and supported by the VSB fund.

References

[1] H. Sakoe and S. Chiba, "A Dynamic Programming Approach to Continuous Speech Recognition," *Proc. Seventh Int'l Congress on Acoustics (ICA '71)*, vol. 3, pp. 65-69, 1971.

[2] G. White and R. Neely, "Speech Recognition Experiments with Linear Prediction, Bandpass Filtering, and Dynamic Programming," *Proc. Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP '76)*, pp. 183-188, 1976.

[3] C. Myers and L. Rabiner, "A Comparative Study of Several Dynamic Time-Warping Algorithms for Connected Word Recognition," *The Bell System Technical J.*, vol. 60, no. 7, pp. 1389-1409, Sept. 1981.

[4] J. di Martino, "Dynamic Time Warping Algorithms for Isolated and Connected Word Recognition," *New Systems and Architectures for Automatic Speech Recognition and Synthesis*, pp. 405-418, Springer-Verlag, 1985.

[5] C. Bahlmann and H. Burkhardt, "The Writer Independent Online Handwriting Recognition System *Frog on Hand* and Cluster Generative Statistical Dynamic Time Warping," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 3, pp. 299-310, Mar. 2004.

[6] T. Starner, "Visual Recognition of American Sign Language Using Hidden Markov Models," master's thesis, Massachusetts Inst. of Technology, Media Arts and Sciences, Jan. 1995.

[7] D. Gavrila and L. Davis, "Towards 3-D Model-Based Tracking and Recognition of Human Movement: A Multi-View Approach," *Proc. IEEE Int'l Workshop Face and Gesture Recognition (FG '95)*, pp. 272-277, June 1995.

[8] A. Corradini, "Dynamic Time Warping for Off-Line Recognition of a Small Gesture Vocabulary," *Proc. IEEE ICCV Workshop Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems (RATFG-RTS '01)*, pp. 82-89, July 2001.

[9] S. Yang and R. Sarkar, "Gesture Recognition Using Hidden Markov Models from Fragmented Observations," *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR '06)*, pp. 766-773, 2006.

[10] W. Stokoe, "Sign Language Structure: An Outline of the Visual Communication System of the American Deaf," *Studies in Linguistics: Occasional Papers*, vol. 8, 1960.

[11] W. Stokoe, "Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf," *J. Deaf Studies and Deaf Education*, vol. 10, no. 1, pp. 3-37, 2005.

[12] E. van der Kooij, "Phonological Categories in Sign Language of the Netherlands. The Role of Phonetic Implementation and Iconicity," PhD dissertation, Leiden Univ./LOT, 2002.

[13] G. ten Holt, A. Koenderink, H. de Ridder, E. Hendriks, and M. Reinders, "How Much of a Sign Do We Really Need? Recognising Parts of Sign Language Signs," *Theoretical Issues in Sign Language Research 9*, Dec. 2006.

[14] L. Ding and A. Martinez, "Recovering the Linguistic Components of the Manual Signs in American Sign Language," *Proc. IEEE Conf. Advanced Video and Signal Based Surveillance (AVSS '07)*, pp. 447-452, 2007.

[15] N. Morgan and H. Bourlard, "Continuous Speech Recognition Using Multilayer Perceptrons with Hidden Markov Models," *Proc. Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP '90)*, pp. 413-416, 1990.

[16] Y. Matsuura, H. Miyazawa, and T. Skinner, "Word Recognition Using a Neural Network and a Phonetically Based DTW," *Proc. IEEE Int'l Workshop Neural Networks for Signal Processing (NNSP '94)*, pp. 329-334, Sept. 1994.

[17] A. Corradini and H. Gross, "Camera-Based Gesture Recognition for Robot Control," *Proc. Int'l Joint Conf. Neural Networks (IJCNN '00)*, vol. 4, pp. 133-138, July 2000.

[18] J. Ye, H. Yao, and F. Jiang, "Based on HMM and SVM Multilayer Architecture Classifier for Chinese Sign Language Recognition with Large Vocabulary," *Proc. Third Int'l Conf. Image and Graphics (ICIG '04)*, pp. 377-380, Dec. 2004.

[19] O. Aran and L. Akarun, "Recognizing Two Handed Gestures with Generative, Discriminative and Ensemble Methods via Fisher Kernels," *Proc. Int'l Workshop Multimedia Content Representation, Classification and Security (MCRCS '06)*, vol. 4105, pp. 159-166, Sept. 2006.

[20] C. Bahlmann, B. Haasdonk, and H. Burkhardt, "Online Handwriting Recognition with Support Vector Machines—A Kernel Approach," *Proc. Eighth Int'l Workshop Frontiers in Handwriting Recognition (IWFHR '02)*, pp. 49-54, 2002.

[21] J. Alon, V. Athitsos, Q. Yuan, and S. Sclaroff, *Simultaneous Localization and Recognition of Dynamic Hand Gestures*, vol. 2, pp. 254-260, 2005.

[22] R. Yang, S. Sarkar, and B. Loeding, "Enhanced Level Building Algorithm for the Movement Epenthesis Problem in Sign Language Recognition," *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR '07)*, pp. 1-8, 2007.

[23] J. Lichtenauer, E. Hendriks, and M. Reinders, "Sign Language Recognition by Combining Statistical DTW and Independent Classification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, Nov. 2008, supplemental material, http://doi.ieeecomputersociety.org/10.1109/TPAMI.2008.123.

[24] R. Fisher, "The Use of Multiple Measurements in Taxonomic Problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179-188, 1936.

[25] O. Hamsici and A. Martinez, "Bayes Optimality in Linear Discriminant Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 647-657, Apr. 2008.

[26] U. von Agris, D. Schneider, J. Zieren, and K.-F. Kraiss, "Rapid Signer Adaptation for Isolated Sign Language Recognition," *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR '06)*, June 2006.