Feature Space Transforms for Czech Sign-Language Recognition

Jan Trmal, Marek Hruž
Jan Zelinka, Pavel Campr, Luděk Müller

Department of Cybernetics, Faculty of Applied Sciences
University of West Bohemia, Czech Republic
{jtrmal mhruz zelinka campr muller}@kky.zcu.cz

Abstract

In this paper we describe a HMM-based sign language recognition (SLR) system for isolated signs. In the first part we describe the image parametrization method producing features used for recognition. Our goal was to find the best combination of a feature space dimension reduction method and an HMM structure.

Index terms: PCA, LDA, ICA, HLDA, heteroscedastic, sign-language, Hu’s moments, tracking, skin-color detection

1. Introduction

Sign language recognition is a task being solved by many research institutes in the world. Despite of this fact it lacks an universal parametrization and recognition method that would be widely accepted as a baseline. In this paper we introduce a parametrization method based on skin-color detection and object tracking. Because of the nature of the extracted visual features, application of a feature space dimension reduction method is beneficial. It results in reduced computation demands, reduced memory demands and more robust parameters estimates leading to a higher recognition score. Despite of our extensive search, we haven’t succeeded to find a comparison of the feature space reduction methods used in speech recognition area on this task. Therefore, we evaluate several methods of feature space transforms and their influence on the recognition score.

2. Visual Feature Extraction System

In this section we present a Visual Feature Extraction (VFE) system used in our SLR. In this work we focus on the manual component of the sign language (SL). According to ([1]), sign linguists distinguish several basic components (or sign subunits) of a sign – handshape, hand orientation, location and movement.

Let’s consider an image or a sequence of images as the observed signal that bears the information about a sign. A VFE system has to be developed in order to successfully recognize the sign. The system must be able to determine the state of the manual component of the SL from the image. For this purpose, we need to detect and classify the objects in the image. In the case of the SLR, the objects of interest are the left and the right hand and the head.

2.1. Skin-Color Detection

A common method for detecting parts of a human body is the skin-color detection ([2]). Skin-color detection can be combined with motion cues ([3]) or edge cues ([4]). Although the method is widespread, it has several disadvantages. For example it is illumination dependent and there is a large variety in color of human skin. Therefore, an adaptation should be applied to the universal skin-color model ([5]). The skin-color detection is the first phase of VFE in our SLR system. Similar to ([6]) it makes use of a Gaussian Mixture Model (GMM) trained on a manually selected subset of all speakers in the database and uses a simple adaptation of the model for every instance of a speaker. For a given speaker we detect the face using Haar classifier. A convolution of a gaussian mask and the color of the detected pixels of the face in the RGB domain gives us an approximate speaker-dependent model. Then, the adapted model is obtained as a weighted average between the speaker dependent model and the universal model. For every pixel in the image we determine it’s likelihood of being a skin-color pixel. If the likelihood is higher than a specific threshold, the pixel is considered to be a part of a skin-color object.

2.2. Object Description

As already mentioned above, the basic components of a sign are: handshape, hand orientation, location, and movement. We need to obtain features that correspond to these components. The following attributes of the located objects are considered:

- location and movement - a set of 2D coordinates representing the mean of the contour (or center of mass) of an object for every frame
- hand shape and orientation - a set of seven Hu’s moments ([7]) of the object

The last step of VFE is the identification of the observed objects.

2.3. Tracking

There are three objects of interest in the scenario of SL. In our system, every object is tracked by its own tracker. The tracking is based on a distance measure of the objects. The tracker uses an extended set of features of the located objects: 7 Hu’s moments of the contour, a gray scale image (a template), position, velocity, perimeter of the contour, area of the bounding box, area of the contour.

For every new frame, each tracker computes the distance between the appearance of each object in the current frame and the appearance of the object tracked in the previous frame. For more information, see ([8]). Thus, getting (number of trackers) × (number of objects) distance measures. The distance function is zero for a perfect match and rises as the difference in the appearance of the object becomes greater. Next we create hypotheses about the configuration of the objects (i.e. which tracker tracks which object). For each configuration k we enumerate the total distance \( L_k = \sum_i \Psi(tr_i, O_{k_i}) \), where \( \Psi \) is the distance function, \( tr_i \) is the \( i-th \) tracker, \( O_{k_i} \) is the \( i-th \)
object of the $k$-th configuration. On the basis of the sum of the total distance, the best hypothesis of the object configuration is created and tested. Two conditions are tested:

- If two trackers track the same object then the object must consist of several objects in occlusion.
- If only one tracker tracks an object then the object cannot be in occlusion with other objects.

If the test fails, the hypothesis is rejected and the next best hypothesis is tested and so on.

2.4. Set of Features

For every frame of the video stream we identify the objects via the tracking process. Then, for every object we evaluate 11 features:

- $x, y$ - the center of mass of the object
- 7 Hu’s moments describing the shape of the object
- the angle of the object relative to the $x$-axis of the image
- a Boolean value representing whether the object is in occlusion.

The occlusion flag is used in post-processing. If an occlusion is detected, the Hu’s moments and the angle are linearly interpolated between the last values before occlusion and the first values after occlusion. Center of mass $(x, y)$ is not interpolated as it tells us how the occluded objects move. The final step of the post-processing is a normalization in the spatial domain. The mean position of the head is considered as the origin and the mean width of the head is considered as one unit. The normalized features (excluding the occlusion flag) are concatenated in the following order: left hand, right hand, and head. For every object, 10 features are obtained, which makes a total count of 30 features for every frame.

3. Reduction of Feature Space Dimensionality

Features obtained so far by the method described in the previous section are highly correlated and statistically dependent on each other.

This fact suggests that not diagonal but full covariance matrix of features should be considered for GMM modeling. Also, the number of correlated features can be higher than the size of independent feature set – it can be proved ([9]) that the Hu system is dependent and incomplete. For our purpose, the Hu’s moments are sufficient to describe the contours of the objects, but the dependence points at the possible use of a dimension reduction method.

In addition, it can be shown ([11]) that the number of the basic sign units (models of GMM/HMM recognizer) can be interpreted as the Cartesian product of 4 sets (corresponding to the basic manual components) with cardinalities of 30, 8, 20, 40 - even when no context like “tri-signs” (as an analogy to tri-phones used in speech recognition) is considered. For this reason, the total number of model parameters to be estimated would become extremely large, particularly when one considers the limited size of a training corpus. Even if we had a good model, the total number of parameters is a limiting factor, when recognition in real-time is needed. For this reason, the choice of a suitable projection scheme method is a very important subtask of SL recognition. In this paper we investigate and compare 5 projection schemes: PCA as the base-line, ICA, LDA, HLDA, and rHLDA (HLDA with more robustly estimated covariance matrices).

Let’s have a set $x$ of $m$-dimensional feature $(m = 30$ in this particular case) vectors $x = \{x_1, \ldots, x_k\}$. Our task is to find a linear projection matrix $A$, that projects the random variable $X$ into $n$-dimensional $(n < m)$ random variable $Y$ given the training set $x$.

3.1. Principal Component Analysis

The Principal Component Analysis (PCA) is a standard technique for statistical data analysis and information extraction. Given the set $x$ we want to find such a matrix $A$ that minimized redundancy between elements of data. The redundancy is measured by the correlations between elements of $F$.

Suppose we have the global correlation matrix $S$ then the PCA finds such a matrix $A$ for which the off-diagonal cross-correlations are minimized.

It is a well-known fact that the solution of the PCA problem is given by terms of eigenvectors and eigenvalues. The transformation matrix $A$ is constructed from eigenvectors whose corresponding eigenvalues are the largest ones.

3.2. Independent Component Analysis

As the name suggests, Independent Component Analysis (ICA) methods assume statistical independence among the elements of $\vec{y}$. The functions evaluating degree of independence take higher statistical moments into account, not only the second moments as PCA does. The crucial issue here is in choice of the criterion of statistical independence. There exists a wide range of criterion functions – the common measures are entropy, Kullback-Leibler divergence, and negentropy and their approximations. In our experiments, we used the FastICA algorithm with function $G(y) = -\exp(-y^2/2)$ (for more details see [10]). This choice has been experimentally shown to perform well for GMM modeling (see [11]).

3.3. (Heteroscedastic) Linear Discriminant Analysis

The Linear Discriminative Analysis (LDA) and the Heteroscedastic Linear Discriminant Analysis (HLDA) construct the transform matrix $A$ in such a way that the discrimination between classes is maximized.

The LDA method can be seen as a special case of HLDA method where all the within-class covariance matrices are the same ($S_1 = S_2 = \ldots = S_C = S_{w}$). Due to this simplification and contrary to the general HLDA, a closed-form solution of the problem exist. The solution is given by eigenvectors of matrix $S_b \times S_w^{-1}$, where $S_b$ is between-class covariance matrix.

The HLDA optimization scheme is based on maximization of diagonal matrix gaussian model. For HLDA, estimates of the global covariance matrix $S$ and within-class covariance matrices $S_w, c = 1, \ldots, C$, where $C$ is total number of modeled classes are needed.

3.4. Robust Heteroscedastic Linear Discriminant Analysis

When dealing with real-life data, we have to estimate the needed covariance matrices. The common way is to use the usual covariance matrix estimator, for the global covariance matrix given as $S = \frac{1}{n-1} \sum_{i=1}^{n} (\vec{x} - \vec{\mu}) (\vec{x} - \vec{\mu})^T$, for the other mentioned correlation matrices analogously.

However, when the amount of data is limited, the aforementioned estimator may not produce sufficiently robust estimate. This is especially the case when HLDA is applied, because for...
Table 1: Recognition accuracy for the dimension $p$ of a feature vector

<table>
<thead>
<tr>
<th>$p$</th>
<th>PCA</th>
<th>ICA</th>
<th>LDA</th>
<th>HLDA</th>
<th>rHLDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>77.7</td>
<td>76.6</td>
<td>76.4</td>
<td>80.4</td>
<td>78.6</td>
</tr>
<tr>
<td>7</td>
<td>80.4</td>
<td>80.4</td>
<td>75.4</td>
<td>84.3</td>
<td>81.9</td>
</tr>
<tr>
<td>8</td>
<td>84.3</td>
<td>83.6</td>
<td>75.6</td>
<td>87.7</td>
<td>82.0</td>
</tr>
<tr>
<td>9</td>
<td>84.9</td>
<td>86.8</td>
<td>77.6</td>
<td>86.1</td>
<td>85.2</td>
</tr>
<tr>
<td>10</td>
<td>86.9</td>
<td>89.2</td>
<td>79.9</td>
<td>88.9</td>
<td>88.0</td>
</tr>
<tr>
<td>11</td>
<td>88.1</td>
<td>89.6</td>
<td>82.6</td>
<td>89.4</td>
<td>89.8</td>
</tr>
</tbody>
</table>

each class the within-class covariance matrix must be estimated. There is a number of possible approaches when dealing with insufficient data. A simple way is to combine two (or possibly more) estimates ([12]), to use a statistical method to identify and remove outliers ([13] and [14]) or assuming the covariance matrix is generated using a model of a lower complexity ([15], [16]). During our experiments, we tested a combination of HLDA with the PPCA (Probabilistic Principal Component Analysis) covariance model ([15]). In this model, the covariance matrix $S$ is expressed in the form

$$S = \sigma I + WW^T,$$

(1)

where $\sigma > 0$ is estimated from data, $I$ is $m \times m$ identity matrix and $W$ is $m \times d$ matrix estimated from data. Requiring $2d < m - 1$, less parameters will be estimated from the given amount of data, therefore the estimate is expected to be more robust. Sometimes, this is also referred to as regularized estimates ([17]). We will denote the combination HLDA + PPCA covariance model as rHLDA in the rest of this paper.

4. Experiments and Results

4.1. Data

The corpus consists of 25 signs performed by 20 people. Every person performed each sign three times, however not all records are usable. Each video record contains one representation of a particular sign. The total number of all representations is 1513. The video files were parametrized using the described method.

4.2. Training

To obtain reliable information about the classification score, we employed $N$-fold cross-validation ($N = 15$). In each step, the train and test set was generated. The train set was then used for computation of the PCA decorrelation matrix – the HLDA and ICA failed to converge, when applied on the original data.

The original feature vector dimension was reduced to 11, keeping of $99.90\%$ of the original signal energy.

The decorrelated training data was then used to train the transform matrix $A$ using one of the described method $M$, $M \in \{LDA, HLDA, PCA, ICA, rHLDA\}$ and reducing the feature dimension to $p$. Finally, the $z$-state HMM model with $c$ mixtures was trained and the recognition score was evaluated.

For each of the mentioned methods $M$, the dimensionality reduction ability was tested (the dimension $p \in \{6, \ldots, 11\}$) and for each of these setups, several HMM models were trained and evaluated (from $z \in \{4, \ldots, 9\}$ states and $c \in \{1, \ldots, 50\}$ mixtures in each state). This evaluates to $180 \times 15$ experiments and $180 \times 50 \times 15$ results.

4.3. Evaluation Methodology

Because of the cross-validation approach, we obtained a set $\theta = \{\theta_1, \ldots, \theta_N\}$, where $\theta_n$ stands for accuracy in the $n$-th cross-validation setup.

The mean recognition score $\theta_{ACC}$ was computed as sample mean $\theta_{ACC} = \frac{1}{N} \sum_{n=1}^{N} \theta_n$. Because the distribution of $\theta_n$ is highly non-normal and non-symmetric, instead of using the $t$-test, we decided to use the bootstrap method ([18]) to determine the 95% confidence intervals $\kappa_-, \kappa_+.$

4.4. Results

The 95% confidence intervals were approximately $\pm 1.2\%$. Moreover, the two-sided Wilcoxon sign-rank test performed on set of all results suggests that the absolute difference in performance under 1% of score is not statistically significant. The p-value for 1% difference is approximately 0.7. The best scores for given method and dimension can be found in Table 1. The recognition scores of the three best performing methods for different number of HMM models can be seen on Graph 1.

5. Conclusion

In the paper, we introduced the signed-language recognition system based on skin-color detection and object tracking. We performed a comparison of several feature space reduction methods. From our point of view, the most perspective method is the rHLDA.

The classifier trained on features processed by rHLDA achieves not only the best score, but also the score is relatively robust to choice of the number of the states and the number of the mixtures during the HMM modeling. The classifier trained using this method performs best for 21 mixtures and $z = 5$. Moreover, when more data will be available, the HLDA (or rHLDA) method enables us to retrain the classifier using different class clustering (for example “each HMM state is one class” instead of “each sign is one class”).

6. Acknowledgments

Support for this work was provided by the Grant Agency of Academy of Sciences of the Czech Republic, under project No. IET101470416, by the Ministry of Education of the Czech Republic under project No. MŠMT LC536 and by the Grant Agency of the Czech Republic, project No. GAČR 102/08/0707. The access to the METACentrum computing facilities provided under the research intent MSM6383917201 is highly appreciated.

7. References

Figure 1: Performance graphs for the three best-performing feature space reduction methods, feature vector dimension $p = 11$.

References:


