Selecting Features in On-Line Handwritten Whiteboard Note Recognition: SFS or SFFS?

Joachim Schenk and Moritz Kaiser and Gerhard Rigoll
Institute for Human-Machine Communication
Technische Universität München
Theresienstraße 90, 80333 München
{schenk,kaiser,rigoll}@mmk.ei.tum.de

Abstract

When selecting features with the sequential forward floating selection (SFFS), the "nesting effect" is avoided, which is a common phenomenon if the computationally less expensive sequential forward selection (SFS) is used instead. In this paper, we answer the key question, if the more complex and sophisticated SFFS should be used in on-line HMM-based recognition of handwritten whiteboard notes. In addition, an efficient method of displaying the selected feature set, the "feature map", is introduced.

In an experimental section, both selection approaches are evaluated, the derived feature sets are compared, and a discussion on the selected features is given.

1. Introduction

In on-line handwriting recognition (HWR), amongst other tasks in pattern recognition, Hidden Markov Models (HMMs, see [1]) have been used for over 25 years, as they offer a combined segmentation and recognition, avoiding error-prone pre-segmentation [2]. More recently, the new task of on-line HMM-based HWR of whiteboard notes has been introduced [3], which plays an important role in so-called "smart meeting room" scenarios (see e. g. [4]). In [5] features were selected for HMM-based HWR of whiteboard notes using the sequential forward selection (SFS, see [6]).

In this paper, we first perform feature selection on the features used in our on-line HWR system for whiteboard notes [7, 8] with the SFS, confirming the findings presented in [5] and introduce a compact notation for the selected features. However, a known drawback of the SFS is the "nesting" effect: once a feature has been added to the final feature set, it cannot be removed [9]. We therefore extend the feature selection by applying the sequential forward floating selection (SFFS, see [9]) which overcomes the nesting effect, and answer the key question, whether the feature sets derived by the computationally more expensive but also more sophisticated SFFS outperforms the feature sets found by the simple SFS approach.

The next section gives a brief summary of our recognition system. Then, the SFS and the SFFS are reviewed. In Sec. 4, the experimental section, features are selected using the former introduced SFS and SFFS, and results are presented. Finally, conclusions and an outlook are drawn in Sec. 5.

2. Recognition System

In this section, we sketch our recognition system, including the preprocessing, feature extraction, and the HMM-based recognizer. A more in-depth discussion on the recognition system can be found in [7, 8].

Preprocessing The x- and y-coordinates as well as the pen's "pressure" p of the handwritten, heuristically line-segmented whiteboard notes are recorded using the EBEAM-System as explained in [3]. Hence, the handwritten script is described by the sample vectors $\mathbf{s}(t) = (x(t), y(t), p(t))^{\mathrm{T}}$. Afterwards a resampling of the data is performed, followed by a correction of the skew and the slant of the script trajectory, using a histogram-based approach as explained in [10]. Finally, all text lines are normalized to meet a distance of "one" between the corpus and the base line.

Feature Extraction Following the preprocessing, 24 state-of-the-art *on-line* and *off-line* features are extracted and form the complete feature set $\mathcal{F}$.

The extracted on-line features are: the pen's "pressure", indicating whether the pen touches the whiteboard surface (f_1); a velocity equivalent, which is computed before resampling (f_2) and later interpolated according to the resampling factors; the x- and y-coordinate after resampling ($f_{3,4}$), whereby the y-coordinate is smoothed by the moving average; the "writing direction", i. e. the angle α of the strokes, coded as $\sin\alpha$ and $\cos\alpha$ ($f_{5,6}$); and the "curvature", i. e. the

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.130

difference of consecutive angles $\Delta\alpha = \alpha(t) - \alpha(t-1)$, coded as $\sin\Delta\alpha$ and $\cos\Delta\alpha$ ($f_{7,8}$); a logarithmic transformation of the "vicinity aspect" v, $\mathrm{sign}(v)\cdot\log(1+|v|)$ (f_9); the "vicinity slope", i. e. the angle φ between the line $[\mathbf{s}(t-\tau), \mathbf{s}(t)]$, whereby $\tau < t$ denotes the τ^{th} sample point before $\mathbf{s}(t)$, and the bottom line, coded as $\sin\varphi$ and $\cos\varphi$ ($f_{10,11}$); and the "vicinity curliness", the length of the trajectory normalized by $\max(|\Delta x|; |\Delta y|)$ (f_{12}). Finally, the average square distance to each point in the trajectory and the line $[\mathbf{s}(t-\tau), \mathbf{s}(t)]$ is given (f_{13}).

The off-line features are: a 3×3 "context map" to incorporate a 30×30 partition of the currently written letter's image (f_{14-22}); and "ascenders" and "descenders" (i. e. the number of pixels above respectively beneath the current sample point) ($f_{23,24}$).

Recognizer Each of the $N = 56$ characters is represented by one linear continuous HMM with $S = 10$ emitting states and the output probability for each state is estimated by mixtures of $M = 32$ Gaussians. The parameters of the HMMs are trained using the Baum-Welch-algorithm; combined recognition and segmentation is enabled by the Viterbi-algorithm [1].

3. Feature Selection

Two standard procedures, namely the sequential forward selection (SFS, see [6]) and the sequential forward floating selection (SFFS, see [9]) are presented in this section, and a common notation is introduced.

Given a set $\mathcal{F} = \{f_1, \ldots, f_D\}$ of D features f_i the main idea behind features selection is to derive a new set $\mathcal{X}_k = \{x_1, \ldots, x_k\}$ containing $k \leq D$ features out of $\mathcal{F}$ in a way such that the performance of the underlying recognition system stays the same or even rises [9] while k declines. All feature selection algorithms use the value of some cost function $J(\mathcal{X}_i)$, where $J(\mathcal{X}_i) > J(\mathcal{X}_j)$ is true if feature set $\mathcal{X}_i$ performs "better" than feature set $\mathcal{X}_j$. In this paper, $J(\mathcal{X}_i)$ denotes the recognition accuracy [5]. The *significance* (i. e. the importance, see [9, 11]) S of each feature f_i (or even feature set $\mathcal{X}_i$) is given as *individual* significance

$$S_0(f_i) = J(f_i) \tag{1}$$

and *joint* significance $S(y_i, \mathcal{Y})$, i. e. the significance of a feature y_i in conjunction with other features in the set $\mathcal{Y}$. There are two types of joint significance:

$$S^-(x_i, \mathcal{X}_k) = J(\mathcal{X}_k) - J(\mathcal{X}_k \setminus x_i), x_i \in \mathcal{X}_k \tag{2}$$

$$S^+(f_i, \mathcal{X}_k) = J(\mathcal{X}_k \cup f_i) - J(\mathcal{X}_k), f_i \in \mathcal{F} \setminus \mathcal{X}, \tag{3}$$

where $\mathcal{Y} \setminus y$ denotes that the feature set $\mathcal{Y}$ does not contain the feature y. It shall be noted that Eq. 2 captures the change in significance when removing the feature x_i from the set $\mathcal{X}_k$, and in Eq. 3 the change in significance is given, when the feature f_i is added to the set $\mathcal{X}_k$.

Following [11], the "worst" feature x_w within the set $\mathcal{X}_k$ is derived to

$$x_\mathrm{w} = \operatorname*{argmin}_{x_i \in \mathcal{X}_k} S^-(x_i, \mathcal{X}_k) \Rightarrow J(\mathcal{X} \setminus x_\mathrm{w}). \tag{4}$$

Accordingly, the "best" feature f_b out of the set of remaining features $\mathcal{F} \setminus \mathcal{X}_k$ regarding to the feature set $\mathcal{X}_k$ is given by

$$f_\mathrm{b} = \operatorname*{argmax}_{x_i \in \mathcal{X}_k} S^+(f_i, \mathcal{X}) \Rightarrow J(\mathcal{X}_k \cup f_\mathrm{b}). \tag{5}$$

SFS The sequential forward selection (SFS, see [6]) starts with a feature set $\mathcal{X}_1$ containing only one feature x_1 which has the highest individual significance $S_0(x_i = f_i)$ out of the complete set of features $\mathcal{F}$, i. e.

$$x_1 = \operatorname*{argmax}_{f_i \in \mathcal{F}} J(f_i). \tag{6}$$

Then the initial feature set $\mathcal{X}_1$ is recursively augmented according to

$$\begin{aligned} x_{k+1} &= \operatorname*{argmin}_{f_i \in \mathcal{F} \setminus \mathcal{X}_k} S^+(f_i, \mathcal{X}_k), \\ \mathcal{X}_{k+1} &= \mathcal{X}_k \cup x_{k+1}, \end{aligned} \tag{7}$$

i. e. the feature $f_i = x_k$ is added which leads to the maximum joint significance. The augmentation is repeated until k features are selected. A pseudo code description of the SFS method is given in Alg. 1. The SFS algorithm has been applied for on-line handwriting recognition in [5], however on a different feature-set and with a different preprocessing than in our work.

```
Algorithm: SFS
Data: $\mathcal{F}$, $k$
Result: $\mathcal{X}_k$
Initialization: $x_1 = \operatorname*{argmax}_{f_i \in \mathcal{F}} J(f_i)$ ; $\mathcal{X} = \{x_1\}$, $\kappa = 1$ ;
while $\kappa < k$ do
    $x_{\kappa+1} = \operatorname*{argmax}_{f_i \in \mathcal{F} \setminus \mathcal{X}_\kappa} S^+(f_i, \mathcal{X}_\kappa)$;
    $\mathcal{X}_{\kappa+1} = \mathcal{X}_\kappa \cup x_{\kappa+1}$ ; $\kappa = \kappa + 1$;
```

Algorithm 1: Pseudo code description of SFS(k)

SFFS A known issue with the SFS is its monotonic growing feature set, i. e. once a feature is added to the final set of features, it cannot be removed. A feature selection method that allows for removing once selected features is the sequential forward floating selection (SFFS, see [11]) which uses the SFS in order to derive an initial feature set of cardinality of two (i. e. $\mathcal{X}_2$). The feature set is augmented by features similar to the SFS; however, in each iteration the feature set can be reduced by the least significant feature x_w. The SFFS selection algorithm is summarized by the pseudo code description shown in Alg. 2.

```
**Algorithm:** SFFS
**Data**: $\mathcal{F}$, $k$
**Result**: $\mathcal{X}_k$
Initialization: $\mathcal{X}_2 =$SFS$(2)$ ; $\kappa = 2$;
**while** $\kappa < k$ **do**
    $x_{\kappa+1} = \underset{f_i \in \mathcal{F} \setminus \mathcal{X}_\kappa}{\text{argmax}}\ S^+(f_i, \mathcal{X}_\kappa)$ ; $\hat{\mathcal{X}}_{\kappa+1} = \mathcal{X}_\kappa \cup x_{\kappa+1}$
    ;
    $x_\text{w} = \underset{x_i \in \hat{\mathcal{X}}_{\kappa+1}}{\text{argmin}}\ S^-(x_i, \hat{\mathcal{X}}_{\kappa+1})$;
    **if** $x_w \neq x_{\kappa+1}$ **then**
        $\hat{\mathcal{X}}_\kappa = \hat{\mathcal{X}}_{\kappa+1} \setminus x_\text{w}$ ; $x_\text{w} = \underset{x_i \in \hat{\mathcal{X}}_\kappa}{\text{argmin}}\, S^-(x_i, \hat{\mathcal{X}}_\kappa)$;
        **while** $(J(\mathcal{X}_\kappa \setminus x_w) > J(\mathcal{X}_{\kappa-1})) \ \wedge \ \kappa > 2$ **do**
            $\hat{\mathcal{X}}_{\kappa-1} = \hat{\mathcal{X}}_\kappa \setminus x_\text{w}$ ; $x_\text{w} =$
            $\underset{x_i \in \hat{\mathcal{X}}_\kappa}{\text{argmin}}\, S^-(x_i, \hat{\mathcal{X}}_\kappa)$;
            $\kappa = \kappa - 1$;
        $\mathcal{X}_\kappa = \hat{\mathcal{X}}_\kappa$;
    **else**
        $\mathcal{X}_{\kappa+1} = \hat{\mathcal{X}}_{\kappa+1}$;
        $\kappa = \kappa + 1$;
```

Algorithm 2: Pseudo code description of SFFS(k)

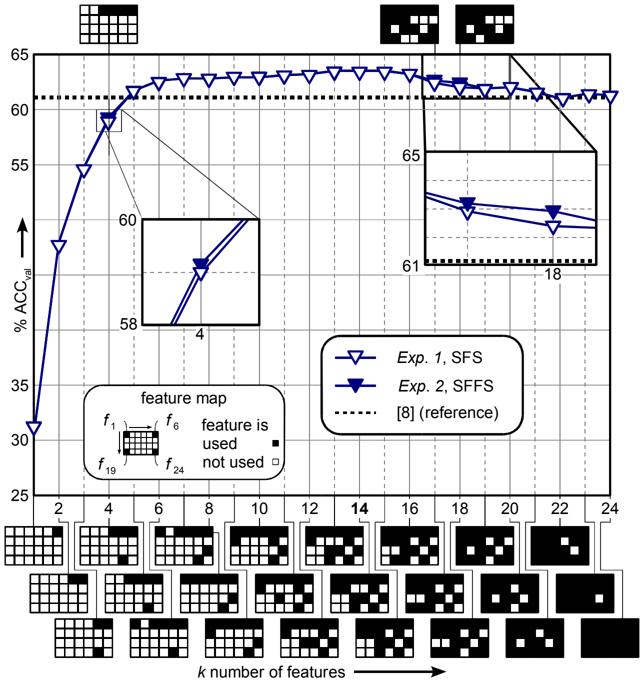


Figure 1: Character-ACC for varying sizes of the feature set $\mathcal{X}_k$ estimated on the validation set. The feature sets are derived using either the SFS or the SFFS. In addition, for each feature set the feature map is given.

4. Experiments

The experiments presented in this section are conducted on the IAM-OnDB database, containing handwritten, heuristically line-segmented whiteboard notes [12]. Comparability of the results is provided by using the settings of the writer-independent IAM-onDB-t1 benchmark, which consists of 56 different letters and provides writer-disjunct sets (one for training, two for validation, and one for testing). Statistical significance of the results is proved by the one-sided t-test, giving the probability p_N of rejecting the hypothesis "both approaches perform equally." Two experiments are conducted in which the SFS and the SFFS are used to select features in HWR of whiteboard notes. The selected features set are depicted as feature map: the features are placed in a 4×6 "matrix," where the feature number rises from left to right and top to bottom beginning with the feature f_1 in the upper left corner. A solid square (■) indicates that the current feature is part of the feature set (see Fig. 1).

In the first experiment (*Exp. 1*), features are selected with the SFS as described in Sec. 3 and summarized in Alg. 1. The results are shown in Fig. 1 (—▽—), where the character-ACC estimated on the validation set is plotted against the number k of used features in the feature set $\mathcal{X}_k^\text{SFS}$. The selected features in each feature set are shown as feature map. The peak character-ACC of $a_\text{v,SFS}$ is reached for the feature set $\mathcal{X}_{14}^\text{SFS}$, which contains $k = 14$ features. Compared to a baseline system ([8], see also line ┅┅ in Fig. 1) that uses all $D = 24$ features (the complete feature set $\mathcal{F}$) a relative, statistically significant improvement of $\Delta r = 3.6\,\%$ ($p_N > 0.99$) to $a_\text{v}^{SFS} = 63.5\,\%$ in character-ACC can be observed on the validation set.

For the second experiment (*Exp. 2*), the computationally more expensive SFFS is used for feature selection. Unlike the SFS, the feature sets found by the SFFS are nesting effect immune, as once selected features can be removed from the feature set. However, as the results of this experiments in Fig. 1 (—▼—) show, only the feature sets $\mathcal{X}_4^\text{SFFS}$, $\mathcal{X}_{17}^\text{SFFS}$, and $\mathcal{X}_{18}^\text{SFFS}$ differ from the corresponding feature sets found by the SFS, delivering slightly better recognition results. In case of $\mathcal{X}_4^\text{SFFS}$, instead of the ascender (f_{23}) the x-coordinate (f_4) is selected (see corresponding feature maps in Fig. 1). A reasonable explanation for the choice of f_3 instead of f_{23} is the redundancy introduced by the feature f_4 (the y-coordinate), which is added as fourth feature: sample points representing ascenders also show high values in their y-coordinate. Given the later added feature describing the y-position (f_4), the former added ascenders (f_{23}) loose significance and are replaced by the feature f_3, which describes the x-coordinate. This exchange is not possible when using the SFS due to the nesting effect. The most intriguing observation is that the best performing feature set, $\mathcal{X}_{14}^\text{SFFS}$, which consists of $k = 14$ features, is the same feature set as has been found by the SFS. Hence, the same relative, statistically significant im-

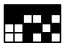
method	SFS	SFFS	[8]
$a_{\mathrm{t}}^{\mathrm{k,SFS}} / a_{\mathrm{t}}^{\mathrm{k,SFFS}}$	$69.2\,\%$		$66.8\,\%$
Δr	$3.5\,\%, p_N > 0.99$		
feature map	$\mathcal{X}_{14}^{\mathrm{SFS}} \equiv \mathcal{X}_{14}^{\mathrm{SFFS}}$: 		$\mathcal{F}$

Table 1: Character-ACC estimated on the test set using either $k = 14$ or all features [8].

provement of $\Delta r = 3.6\,\%$ compared to the baseline system as for the SFS can be observed.

In a final test, the parameters and the features contained in the feature set $\mathcal{X}_{14}^{\mathrm{SFS}} \equiv \mathcal{X}_{14}^{\mathrm{SFFS}}$, which delivered the best performance on the validation set are used for a final test on the test set. The results are shown in Tab. 1.

As can be seen from the feature map of the feature sets $\mathcal{X}_{14}^{\mathrm{SFS}} \equiv \mathcal{X}_{14}^{\mathrm{SFFS}}$ (see Fig. 1 and Tab. 1), the "writing direction" ($f_{5,6}$) and the pressure information f_1 are one of the most important features. This result confirms the findings presented in [5]. These features are also known to be vital for pen-based HWR, e. g. on tablets [13]. In contrast to [5], in our system the feature f_{23}, the ascender, is significant. This is due to a different preprocessing, while in [5] a text line is split up into subparts and each sub part is individually normalized in size, in our system the whole text line is size normalized in a holistic manner (see Sec. 2). Hence, in our system f_{23} contains important information on the size of the text line. Besides the ascenders, in our system, the off-line features f_{15}, f_{16}, f_{18} and f_{21}, which belong to the context map proved to be significant. Feature f_{18} describes the center of the context map and equals the weighted pressure information. The left side and the right side of the context map are described by the features f_{15} and f_{21}, respectively. These features help to distinguish between characters like "t" and "l" and "i" and "e". The fact that both the SFS and the SFFS deliver the same feature set with best performance shows that the selected features are stable and well suited for the task of HWR of whiteboard notes.

5. Conclusion and Outlook

In this paper we investigated feature selection in on-line continuous HMM-based HWR of whiteboard notes. First, results already presented in [5], where features are selected using the SFS, have been confirmed. Taking the nesting effect as example, a known issue with SFS, the use of the computationally more expensive SFFS has been motivated for feature selection. A major outcome of our experiments is that both the SFS and SFFS deliver the same optimal feature set. With those features, a baseline system, which uses all features could be outperformed. A peak character-ACC of $a_{\mathrm{v}} = 63.5\,\%$ estimated on the validation set and $a_{\mathrm{t}} = 69.2\,\%$ estimated on the test set can be reported. This translates to a relative, statistically significant improvement of $\Delta r = 3.6\,\%$ and $\Delta r = 3.5\,\%$, respectively, compared to a baseline system, which uses *all* $D = 24$ features. Significance of the results was proved by the one-sided t-test. In this paper we showed that using the SFFS for feature selection in on-line HWR of whiteboard notes does not lead to better feature sets.

In [7] we presented a recognition system based on *discrete* HMMs. In future work, we plan to extend the feature selection approach presented here for use with discrete HMM-based recognition systems.

References

[1] L.R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257 – 285, 1989.

[2] R. Plamondon and S.N. Srihari, "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 63 – 84, 2000.

[3] M. Liwicki and H. Bunke, "HMM-Based On-Line Recognition of Handwritten Whiteboard Notes," *Proc. Int. Workshop on Frontiers in Handwriting Recognition*, pp. 595 – 599, 2006.

[4] A. Waibel, T. Schultz, M. Bett, I. Rogina, R. Stiefelhagen, and J. Yang, "SMaRT: The Smart Meeting Room Task at ISL," *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. 4, pp. 752 – 755, 2003.

[5] M. Liwicki and H. Bunke, "Feature Selection for On-Line Handwriting Recognition of Whiteboard Notes," *Proc. Conf. of the Graphonomics Society*, pp. 101 – 105, 2007.

[6] A. W. Whitney, "A Direct Method of Nonparametric Measurement Selection," *IEEE Trans. on Comput.*, vol. 20, no. 9, pp. 1100 – 1103, 1971.

[7] J. Schenk, S. Schwärzler, G. Ruske, and G. Rigoll, "Novel VQ Designs for Discrete HMM On-Line Handwritten Whiteboard Note Recognition," *Proc. 30th Symposium of DAGM*, pp. 234 – 243, 2008.

[8] J. Schenk, J. Lenz, and G. Rigoll, "Novel Script Line Identification Method for Script Normalization and Feature Extraction in On-Line Handwritten Whiteboard Note Recognition," *Pattern Recognition Journal*, p. in press, 2009.

[9] M. Kudo and J. Sklansky, "Comparison of Algorithms that Select Features for Pattern Classifiers," *Pattern Recognition Journal*, vol. 33, no. 1, pp. 25 – 41, 2000.

[10] E. Kuvallieratou, N. Fakotakis, and G. Kokkinakis, "New Algorithms for Skewing Correction and Slant Removal on Word-Level," *Proc. Int. Conf. on Electronics*, vol. 2, pp. 1159 – 1162, 1999.

[11] P. Pudil, J. Novovičová, and J. Kittler, "Floating Search Methods in Feature Selection," *Pattern Recognition Letters*, vol. 15, no. 11, pp. 1119 – 1125, 1994.

[12] M. Liwicki and H. Bunke, "IAM-OnDB - an On-Line English Sentence Database Acquired from Handwritten Text on a Whiteboard," *Proc. Int. Conf. on Document Analysis and Recognition*, vol. 2, pp. 1159 – 1162, 2005.

[13] B.Q. Huang M.T. Kechadi, "A Fast Feature Selection Model for Online Handwriting Symbol Recognition," *IEEE Int. Conf. on Machine Learning and Applications*, pp. 251 – 257, 2006.