Clustering Writing Styles with a Self-Organizing Map

Vuokko Vuori
Laboratory of Computer and Information Science
Helsinki University of Technology
P.O. Box 9800, FIN-02015 HUT, Finland

Abstract

This work shows how a Self-Organizing Map (SOM) can be applied in the analysis of different handwriting styles. The analyzed handwriting samples have been collected in on-line fashion with special writing equipments such as pressure sensitive tablets. The handwriting style of an individual subject is represented by a vector, components of which reflect the tendencies of the writer to use certain prototypical styles for isolated alphanumeric characters. This study shows that correlations between different writing styles, both character-wise and writer-wise can be found. Clusters of different personal writing styles can be found by studying the U-matrix viasualization of the SOM trained with data collected from over 700 subjects. An examination of the component planes of the SOM reveals some interesting correlations between the prototypical character styles.

1. Introduction

In this work, natural writing styles of several hundreds of writers are analyzed. The aim of the study is to find a representation for personal writing styles which enables their comparison and detection of possible clusters. In addition, correlations between the writing styles of characters of different classes are searched for. This work tries to find answers to questions such as: “If I know how you write letter ’a’, can I infer something about the way you write letter ’d’ based on what I know about other writers?”. This kind of information might be useful in automatic recognition of handwritten characters [7] by helping to distinguish confusing characters without using any linquistic or geometrical context of the characters, dictionary, or any other language model. In addition, it might be useful by speeding up the recognition process when used in the pruning or ordering of the prototype set representing the different writing styles of the characters. For earlier studies on automatic characterization of handwriting styles, see [1], [3], [10], [11], and [17].

The writing style of a single writer is represented by a vector, components of which indicate the writer’s tendencies to use the writing styles identified by the character prototypes. The prototypes have been selected by hand from the results of four different clustering algorithms applied to a database of handwritten character samples collected in a on-line mode from over 700 subjects [14]. In order to find correlations between and within the writing styles of different writers, the writing style vectors are analyzed and visualized with a Self-Organizing Map (SOM) [6]. The SOM-algorithm performs a nonlinear mapping which preserves the local topological properties of the data set. Clusters of the writing style vectors can be found by studying the U-matrix [12] of a SOM. The clusters can be explained by examining the component planes of the SOM. Also, correlated writing styles for isolated characters can detected easily as they produce similar component planes.

2. Writing style vectors

The writing style of an individual writer is represented by a vector called here *a writing style vector*. Each component of a writing style vector corresponds to a specific character prototype and indicates the tendency of the writer to use that particular style for writing characters of the class of the prototype. The next sections will explain in detail the steps which have been taken in order to form the writing style vectors for the writers. First, the dissimilarity measure between the character samples is described. Next, the clustering algorithms and the final prototype selection procedure are explained. Finally, the transformation from a dissimilarity measure into a similarity measure is presented and the formation of the writing style vectors from averaged similarity measures is explained.

2.1. Dissimilarity measure

The dissimilarity measure used in the character comparisons is based on the Dynamic Time Warping (DTW) algorithm [9], which is a nonlinear curve matching method. The connected parts of a drawn curve in which the pen is pressed down on the writing surface are considered as strokes. The dissimilarity measure is defined on stroke basis so that it is infinite between two characters having different numbers of strokes. The strokes and data points are matched in the same order as they were produced and the first and last data points of the two curves are strictly matched against each other. The DTW-algorithm finds the point-to-point correspondence between the curves which satisfies these constraints and yields the minimum sum of the costs associated with the matchings of the data points. A cost for matching two data points is their squared Euclidean distance.

Prototype-based classifiers using DTW-based distances have been shown to be well suited for the handwriting recognition task by several researchers, and good recognition accuracies can be obtained if the prototype set has a good coverage of the different handwriting styles [15]. In this work, the DTW-based dissimilarity measure is used in the clustering algorithms as a distance measure.

2.2. Clustering and prototype selection

The character database was clustered in order to find all the different writing styles for each character class and to select a set of prototypes which captures the within-class style variations well. All the character classes and stroke number variations were treated separately. This approach does not take in account the between-class variations and the found prototypes are not optimized in the sense of their classification capacity. For some previous works on prototype selection, see [2], [8], and [18].

Four different algorithms were used for the clustering of the character samples: *TreeClust*, *MinSwap*, and two variations of the C-means algorithm [4], named here *CMeans 1* and *CMeans 2*. All the four clustering algorithms were agglomerative and hierarchical. Clusters were represented by prototypes which were the samples having the minimum sum of distances to the other samples in the same cluster. *TreeClust*, *MinSwap*, and *CMeans 2* started form a situation in which all the samples were prototypes, i.e. formed their own clusters, while in the beginning of the *CMeans 1*-algorithm, only a random subset of the samples was selected to be the initial prototype set.

As the clustering algorithms proceeded, the number of clusters was reduced by merging of clusters. In *TreeClust*-, *CMeans 1*-, and *CMeans 2*-algorithms those two clusters whose prototypes were most similar to each other were merged into one. *MinSwap*-algorithm tried several alternative mergings, first the clusters with the most similar prototype pair, then the clusters with the next similar pair etc.

A new prototype was selected among the samples which belonged to the new cluster. After that, *MinSwap*, *CMeans 1*, and *CMeans 2* reassigned the samples into the clusters according to the closest prototypes and then reselected the prototypes. This was continued until a stable division was found. *MinSwap* did the same thing but also calculated how many of the samples were swapped out from the new cluster into the other clusters, or vice versa, and selected the alternative merging which gave rise to the minimum number of these swappings.

The number of clusters was first determined automatically by using two clustering indices. However, it turned out that much better results could be obtained by selecting the prototypes by hand among the cluster centers found by the four clustering algorithms because the results obtained with the different clustering algorithms and indices varied considerably [14]. This guaranteed that each different writing style found with any of the clustering algorithms was present in the final prototype set and that the prototypes were not too similar to each other. The total number of selected prototypes was 2591. Some of the selected prototypes can be seen in Figure 2. Even if some of the prototypes look very similar to each other, say the prototypes of letter ’I’ in the 5th and 6th rows or prototypes of digit ’5’ in the last row, they do have different numbers of strokes, different drawing orders, or directions for the strokes.

2.3. Transforming dissimilarity into similarity

The dissimilarity measure obtained with the DTW-algorithm has a range from zero to infinity and it depends on the numbers of data points and strokes. Therefore, the dissimilarities between strokes have been normalized by the number of data point matchings and the total dissimilarities have been divided by the number of strokes. After these normalizations, the dissimilarities (D) have been transformed into similarity measures (S) in the following way:

$$S = e^{-\alpha\sqrt{D}} \tag{1}$$

The similarity measure is a decreasing function of the normalized dissimilarity measure and its range is between zero and one. The value of parameter $\alpha = 6.52 \times 10^{-4}$ was selected so that the distribution of the similarity measures between character samples and their best matching correct prototypes is approximately even. In practice, this was achieved by fitting a linear function, which was defined by parameter α, in the minimum squared error sense to the logarithm of the cumulative probability function of the dissimilarity measures.

2.4. Forming the writing style vectors

Writer’s tendencies to use the prototypical styles for isolated characters are measured by average similarity values. The average similarity value of a prototype is calculated by:

1) evaluating the similarity values between the prototype and all the writer's character samples of the same class and having the same number of strokes, 2) summing up the similarity values, and 3) finally dividing the sum by the number of its terms. The average similarity values are concatenated into a writing style vector. The dimensionality of a writing style vector is the same as the size of the prototype set. If a subject had no samples at all for some class, all the average similarity values corresponding to that class were considered to be missing from the writing style vector and did not have any effect in the training of the SOM. If a writer had only one character sample for some class, his or her tendencies to use the prototypical styles of that class were estimated by single similarity values instead of averaged similarity values. In such cases, the writer's tendencies to use the prototypical styles consisting of a different number of strokes than the collected sample are zero. In addition, a single sample leads to an assumption that the writer uses only the writing style corresponding to the best matching prototype as the similarity values between the sample and the other prototypes are in most cases very close to zero. For the same reason, the sum of the average similarity values calculated for prototypes of the same class and having the same number of strokes is rarely over one.

3. Data

The experiments were performed with two public databases: IRONOFF [13] and UNIPEN train_r01_v07 [5]. Only isolated digits and upper and lower case letters were used in the experiments. The two databases were combined into one, all the character samples were manually checked and obviously erroneous ones were removed. Most of the erroneous samples were incorrectly segmented. In total, 3 174 erroneous samples were found. The total number of samples in the cleaned database was 130 831. These samples were written by 728 subjects. The subjects were of various ages and from several countries and both handedness groups were represented. In my opinion, it is justified to assume that the database has a rather good coverage of the existing writing styles.

The character samples have been collected with pressure-sensitive displays or tablets which are able to record the x- and y-coordinates of a moving pen point. As there were several contributors and therefore many different collection softwares and devices, all the character samples were preprocessed so that their data points were similarly distributed. It was done by first interpolating straight lines between the original data points and then resampling new data points which were equally spaced on the estimated pen trace. In order to make the DTW-based comparison of the character samples reasonable, the size and location variations of characters were be normalized. The mass centers of the character were moved to the origin of the coordinate system. The characters were scaled so that the longer sides of the smallest boxes drawn around the characters and aligned with the coordinate axes had a constant value. The scaling of the characters was performed prior to the resampling. No other features were used for representing pen traces but the x- and y-coordinates.

4. Creating a SOM of different writing styles

A SOM is a neural network in which the neurons are connected to each other so that they form a regular lattice. Each neuron acts both as an input and output neuron and is associated with a reference vector. The reference vectors are compared with the network's input. The outputs of the neurons depend on how similar the input and reference vectors are. The neuron, reference vector of which is most similar to the input vector, is called the best-matching map unit (BMU). During the training of the network, the reference vectors of the BMUs and their neighboring neurons are updated so that they better represent the input vectors, in this work the writing style vectors. Due to such training, different neurons will specialize in representing different areas of the input space. In addition, neurons near to each other in the neuron lattice tend to correspond to areas close to each other in the input space. Therefore, a SOM can be seen as a nonlinear mapping from the input space to the lower-dimensional lattice space. The SOM's ability to represent the training data faithfully depends on the true dimensionality of the data set and on the size and dimensionality of the neuron lattice.

As the main interest of this work is to find correlations between the writers, all the styles used by only a single writer were omitted from the writing style vectors. So, all the prototypes for which the average similarity was above 0.05 only for a single writer were consider uninteresting. This way, the dimensionality of the writing style vectors was reduced from 2 591 down to 1 764. The kept prototypes were used by 146 subjects on the average. Approximately 11% of the average similarity values were missing from the writing style vectors. The 1 764-dimensional writing style vectors were further analyzed with a SOM in hope of finding interesting structures such as clusters of writers.

Various alternatives for the SOM's size, lattice, neighborhood function, training algorithm, training parameter and epochs, initialization, and updating rule were experimented with. Different SOMs were compared with each other by using two quality measures: quantization error and ability to preserve the topology of the data. The former measure is the average distance between each writing style vector and its BMU. The latter one is the proportion of all data vectors for which the first and second BMUs are not adjacent units.

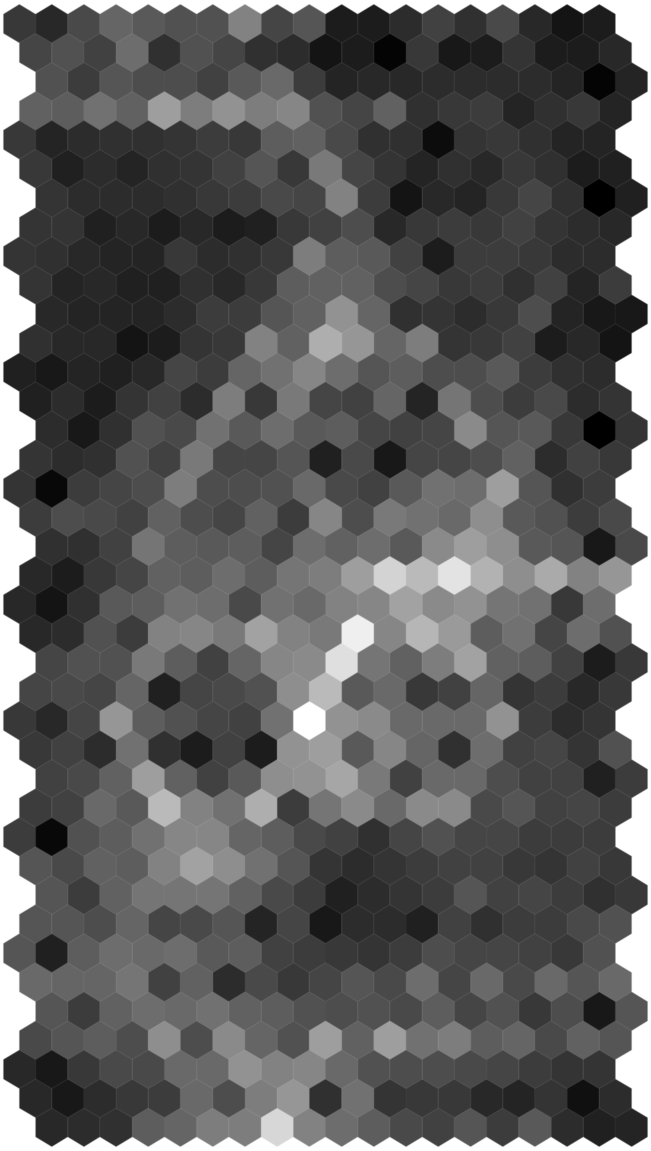

Figure 1. U-matrix formed for the 1764-dimensional writing style vectors.

The size of the SOM was fixed to 20×10 neuron units which is approximately 30% of the number of writers. The topology of the map was selected to be a sheet with hexagonal lattice and Gaussian neighborhood function. A linear initialization along the first two principal directions of the data proved to produce better results than a random initialization. The batch training algorithm was applied with Euclidean metric as their combination provided much faster and reliable convergence than an on-line training algorithm or a metric based on the angle between two vectors.

The training was carried out in three phases. In the first phase, rough training, the radius of the neighborhood was linearly decreased from 10 to 6 during 10 training epochs. In the second phase, the radius was decreased from 5 to 3 during 50 epochs. Finally, in the fine-tuning phase, the radius was decreased from 2 to 1 during 100 epochs. An epoch means that the BMUs are found for all the training samples and total errors are calculated for all the map units, both for the BMUs and their neighboring map units on the hexagonal lattice. The neighborhood function and its radius determine how the errors are distributed to the map units around the BMUs. After finding the total errors, all the map units are then updated simultaneously on the basis of the total errors so that they better represent the training samples. The number of the epochs in the fine-tuning phase is perhaps unnecessarily large but there was no need to optimize it as the batch training was rather fast taking about ten minutes in total. The proportion of all writing style vectors for which the first and second best-matching map units were not adjacent was 0.01. Therefore, it can be said that the map preserves the local topological relations of the writing style vectors rather well.

5. Analysis of the writing style map

The U-matrix of a SOM is helpful in detecting clusters on the map. Its coloring is based on the distances between neighboring map units. Areas in which the neighboring map units are similar to each other are colored with dark gray, whereas light shades indicate that the differences between the neighboring units are more significant. Therefore, clusters of personal writing styles can be seen on the U-matrix as dark areas surrounded by lighter areas. The SOM can also be visualized with images colored according to the values of the components of the reference vectors. These images are called components planes. Component planes show how the tendencies to use the corresponding prototypical character styles vary over the map.

The U-matrix and some interesting component planes of the constructed SOM are shown in Figures 1 and 2. It can be seen from the U-matrix of the SOM that the writing styles can roughly be divided into several clusters. There are small clusters in the left and right lower corners of the SOM surface, a slightly bigger one above them on the vertical middle line of the map, three small clusters on the right edge of the map, a triangular-shaped cluster near the upper edge of the map, and three clusters on the left edge on and above the horizontal middle line of the map.

The interesting component planes are those which show significant variance between the map units. Here, the component planes whose range is at least 0.30 have been selected for further examination. In these cases, it can be claimed that there really are some differences in the ten-

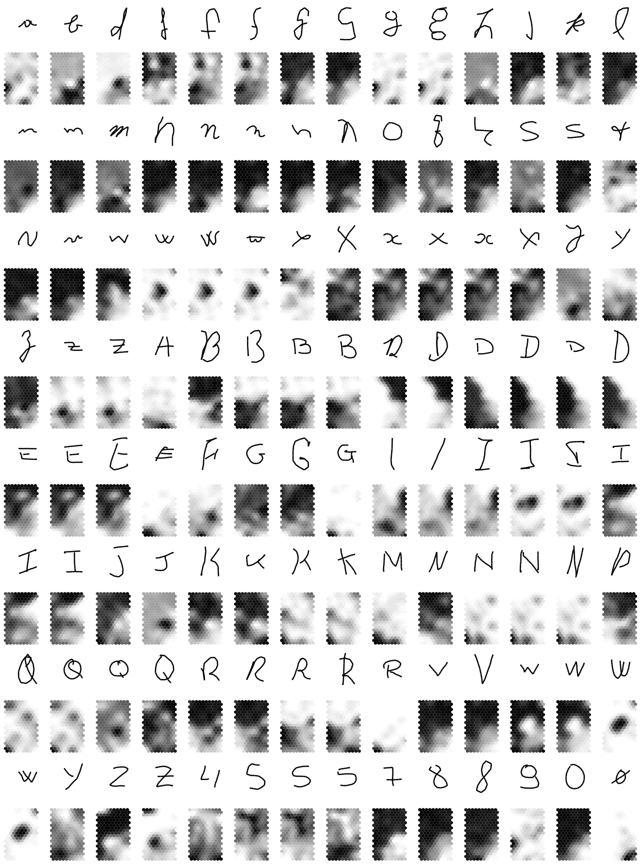

Figure 2. Some interesting component planes with the corresponding prototypes. The range of the component values is at least 0.3.

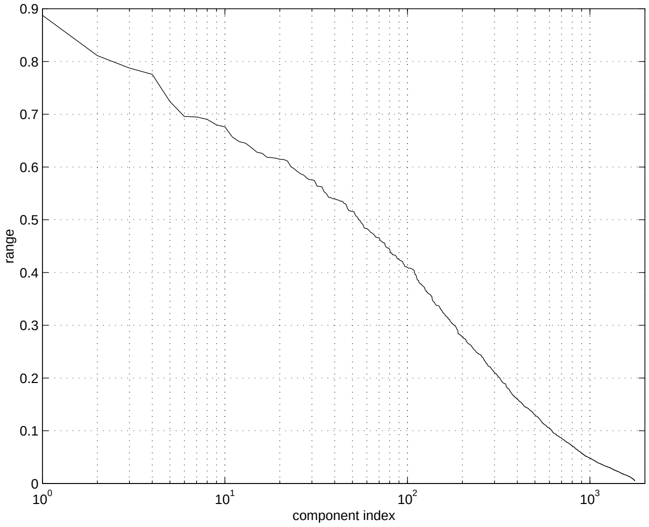

Figure 3. Ranges of the components of the writing style in the decreasing order of magnitude.

dencies of the writers to use the prototypes corresponding to the component planes in the different areas of the SOM. The ranges of all component planes are shown in Figure 3 in the decreasing order of magnitude. From that figure it can be seen that roughly only 10% (i.e. 180 of 1761) of all the character prototypes might explain the clusters of different writing styles.

The clusters can be further analyzed by studying the component planes shown in Figure 2. Dark shades on the component planes indicate that writers mapped in that location have a high tendency to use the corresponding writing style. Light shades mean that the writing style is not likely to be used in those parts of the map. From Figure 2 it can be seen that there is no straightforward explanation for all the clusters. Even though the component planes are clearly organized, the areas in which the writers have high tendencies to use alternative styles for writing characters of certain class are overlapping. In addition, not all the clusters have their own marking prototypes which are not used anywhere else on the map. For an example of alternative prototypes with overlapping component planes, look at the two-stroke letters ’B’ or ’D’ which are the 6th to 8th or 11th to 14th items in the 4th row. However, some of the alternative styles are rarely used by the same writer, see the prototypes of one-stroke and two-stroke letters ’w’, ’z’, ’B’, ’D’, ’G’, ’R’, ’W’, ’Z’, and digit ’0’. Therefore, if the recognition system have already observed that the current user writes digit ’0’ with two strokes, the single-stroke prototype of the same class can be pruned away from the prototype set quite safely. This will make the recognition of letters ’O’ and ’o’ easier.

6. Conclusions

The first analysis of the 728 writing style vectors showed that approximately 32% of the 2 591 prototypical styles were used by a single subject. In a previous experiment of much smaller scale, 22% of the 327 prototypical styles found from a database of character samples written 45 subjects were used only by one writer [16]. These results show that the personal writing styles contain character shapes which cannot be learned from a character database collected from other writers, even if the database is rather large as in this work. Therefore, in order to achieve satisfactory recognition result for all kinds of writers, a recognition system based on character prototypes has to be able to learn new writing styles.

The analysis of the writing style vectors with a SOM showed that some correlations between different writing styles can be found on a character level. The personal writing styles were characterized with vectors whose components reflected the tendency of a writer to use some prototypical styles for isolated characters. According to the U-matrix of the SOM, several clusters of writers can be found. However, the interpretation of these clusters is not straightforward: the component planes are indicating high tendencies of the writers to use the corresponding prototypes in the locations of several clusters and the areas where alternative styles are likely to be used are overlapping. The differences between the alternative styles have to be drastic enough, for example different number and drawing order of the strokes, in order to see clear negative correlation between the corresponding component planes.

The results of the experiments with a SOM justify the use of the knowledge on the writing styles of other writers in the adaptation of a recognition system into a new writing style only to some extent. On the basis of the analysis of ranges of the component planes, the number of character prototypes which might explain the writing style clusters is rather small. Therefore, on the basis of only a few arbitrary characters samples the knowledge in which cluster a new writer belongs to cannot be established and the prototype set cannot be pruned effectively without compromising the recognition accuracy. However, the prototypes can be ordered according to the estimated components of the writing style vector and this might speed up the recognition process. These claims ought to be proved experimentally and that is what we intend to to do in our future work.

References

[1] V. Bouletreau, N. Vincent, R. Sabourin, and H. Emptoz. Synthetic parameters for handwriting classification. In *Proceedings of 5th International Conference on Document Analysis and Recognition*, volume 1, pages 102–106. IEEE, 1997.

[2] S. D. Connell and A. K. Jain. Learning prototypes for online handwritten digits. In *Proceedings of the 14th International Conference on Pattern Recognition*, pages 182–184, 1998.

[3] J.-P. Crettez. A set of handwriting families: style recognition. In *Proceedings of 3th International Conference on Document Analysis and Recognition*, pages 489–494, Montreal, Canada, August 1995.

[4] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, 1973.

[5] I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, and S. Janet. Unipen project of on-line data exchange and recognizer benchmark. In *Proceedings of International Conference on Pattern Recognition*, pages 29–33, 1994.

[6] T. Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer-Verlag, 1997. Second Extended Edition.

[7] R. Plamondon and S. N. Srihari. On-line and off-line handwriting recognition: A comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):63–84, January 2000.

[8] L. Prevost and M. Milgram. Modelizing character allographs in omni-scriptor frame: a new non-supervised clustering algorithm. *Pattern Recognition*, 21:295–302, 2000.

[9] D. Sankoff and J. B. Kruskal. *Time warps, string edits, and macromolecules: the theory and practice of sequence comparison*. Addison-Wesley, 1983.

[10] L. Schomaker, G. Abbink, and S. Selen. Writer and writing-style classification in the recognition of online handwriting. In *Proceedings of the European Workshop on Handwriting Analysis and Recognition: European Perspective*, 1994. The Institution of Electrical Engineers, Digest Number 1994/123, (ISSN 0963-3308).

[11] J. Subrahmonia. Similarity measures for writer clustering. In L. R. B. Schomaker and L. G. Vuurpijl, editors, *Proceedings of the 7th International Workshop on Frontiers in Handwriting Recognition*, pages 541–546, Amsterdam, September 2000. Nijmegen: International Unipen Foundation. ISBN 90-76942-01-3.

[12] A. Ultsch and H. P. Siemon. Exploratory data analysis: Using Kohonen networks on transputers. Technical Report 329, Univ. of Dortmund, Dortmund, Germany, December 1989.

[13] C. Viard-Gaudin, P. M. Lallican, S. Knerr, and P. Binter. The IRESTE on/off (IRONOFF) dual handwriting database. In *Proceedings of 5th International Conference on Document Analysis and Recognition*, pages 455–458, Bangalore, India, September 1999.

[14] V. Vuori and J. Laaksonen. A comparison of techniques for automatic clustering of handwritten characters. 2002. Accepted to be published in the proceedings of the 16th International Conference on Pattern Recognition.

[15] V. Vuori, J. Laaksonen, E. Oja, and J. Kangas. Experiments with adaptation strategies for a prototype-based recognition system for isolated handwritten characters. *International Journal on Document Analysis and Recognition*, 3(3):150–159, March 2001.

[16] V. Vuori and E. Oja. Analysis of different writing styles with the self-organizing map. In *Proceedings of the 7th International Conference on Neural Information Processing*, volume 2, pages 1243–1247, November 2000.

[17] L. Vuurpijl and L. Schomaker. Coarse writing-style clustering based on simple stroke-related features. In A. C. Downton and S. Impedovo, editors, *Progress in Handwriting Recognition*, pages 37–44. World Scientific Publishers, 1997.

[18] L. Vuurpijl and L. Schomaker. Finding structure in diversity: A hierarchial clustering method for the categorization of allographs in handwriting. In *Proceedings of 5th International Conference on Document Analysis and Recognition*, pages 387–393. IEEE, 1997.