AUTOMATIC INDEX CREATION FOR HANDWRITTEN NOTES

Shingo Uchihashi, Lynn Wilcox

FX Palo Alto Laboratory
3400 Hillview Avenue
Palo Alto, CA 94304
{shingo, wilcox}@pal.xerox.com

ABSTRACT

This paper describes a technique for automatically creating an index for handwritten notes captured as digital ink. No text recognition is performed. Rather, a dictionary of possible index terms is built by clustering groups of ink strokes corresponding roughly to words. Terms whose distribution varies significantly across note pages are selected for the index. An index page containing the index terms is created, and terms are hyper-linked back to their original location in the notes. Further, index terms occurring in a note page are highlighted to aid browsing.

1. INTRODUCTION

A key feature differentiating electronic note-taking from paper note-taking is the ability to create automatic indexes that make it easier to find specific information in notes. With paper note-taking, such indexes must be created manually. Since this is difficult, people tend to mark important items or keywords by underlining, circling, or asterisks. While this does not provide an index, it does help users locate important information while browsing.

In electronic text systems, techniques exist to create automatic “back-of-the-book” indexes [5]. This allows users to scan a list of keywords in the index and find occurrences of the index terms in the text. Indexing is also possible in electronic systems that use digital ink rather than text. One example is the application of keywords to sections of electronic notes, as in *Dynomite* [7] and *Marquis* [6]. Another is ink properties in *Dynomite*. An ink property is a data type applied to selected ink, that allows that ink to be subsequently retrieved by type. Example data types are “Name” or “ToDo” item. Ink index pages for a given ink property can be created, allowing users to quickly scan all notes with that property. In addition, notes on the ink index page are hyperlinked back to their original location in the notes.

A problem with creating either keyword or ink indexes for electronic notes is that they require significant cognitive effort. Even the simplest user interface does not overcome the fact that creating indexes while taking notes is difficult [1]. In theory, it is possible to create such indexes after note-taking. However, in practice people are usually not sufficiently disciplined or organized to do it.

One solution is to create indexes automatically. In electronic text systems, information retrieval techniques are often used for automatic indexing of text documents. For example, in [5], index terms are selected for Web pages based on relative frequency of occurrence. However, these techniques do not apply directly to digital ink, since words are not identified. In principle, digital ink could be converted to text using character recognition. However, recognition is not accurate on handwritten data.

In this paper we describe a method to generate indexes from digital ink data automatically. This is performed in two stages. In the first stage, groups of ink strokes, called ink words, are clustered using a hierarchical clustering technique. The distance between ink words is computed using dynamic programming. Each cluster containing more than one ink word is a potential index term. In the second stage, a Chi-square analysis of potential index terms is performed to determine those terms whose distribution varies significantly across note pages. The terms with the largest Chi-square are selected as index terms. An ink index page containing the index terms is created, with hyperlinks from the terms back to their original location on the page. In addition, index terms can be highlighted on the display to aid browsing.

2. RELATED WORK

A system for indexing historical handwritten documents is described in [2]. Images are segmented into words, and word equivalence classes are found by thresholding match scores between words. Since no stroke information is available, match scores are computed based on the word images. Index terms are chosen from the largest equivalence classes; stop words are manually deleted.

Poon, Weber, and Cass [4] describe a technique called scribble matching to find occurrences of a given word in a handwritten document. The technique is based on using dynamic programming to compute a score between the given handwritten word and the words in the document. A similar method is described in [3].

Schutze [5] describes a technique for creating a “back-of-the-book” index for Web pages. Index terms for a Web page a re selected by computing the ratio of the relative frequency of a term in the page to its frequency over a corpus of Web pages.

3. INK WORD CLUSTERING

3.1 Preparation

Ink strokes are first grouped into units corresponding roughly to words. We call these ink words, to distinguish them from the actual words in the notes. Grouping is based on time and spatial distance, as in [8]. The raw data for each stroke in an ink word is a sequence of sampled points on a trajectory separated equally in time. Strokes are re-sampled so that sample points are separated equally on the trajectory. This re-sampling is known to improve robustness of stroke matching. Features are computed using re-sampled points. The computed features are 1) the tangent angle θ_n, 2) derivative of the current tangent angle $\delta\theta_n$, 3) the second derivative of the current tangent angle $\delta^2\theta_n$, 4) sin of the tangent angle sin θ_n, and 5) cosine of the tangent angle cos θ_n. Each stroke produces one feature vector sequence. The idea of features is illustrated in Figure 1. Every feature sequence in the group of

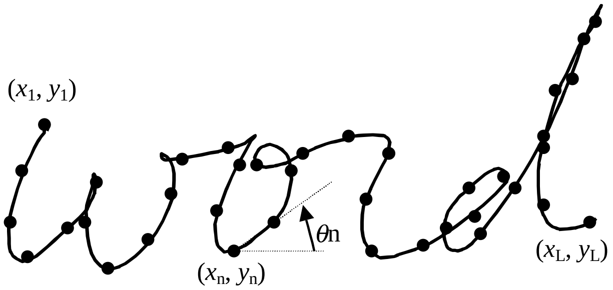


Figure 1. Features for matching

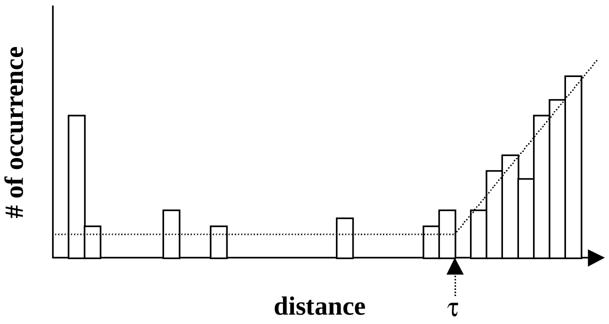


Figure 2. Knee of a curve

strokes in the ink word is connected together, creating a single feature sequence for each ink word.

Based on the features described above, the match score for pairs of ink words are computed using Dynamic Programming (DP) method. The match scores normalized by length of feature sequences are used as distance measure.

3.2 Automatic clustering

Ink words are clustered based on matching scores using a hierarchical clustering algorithm. Each ink word initially forms a cluster with a single instance of itself. The pairwise distance between all clusters is computed, and the two closest clusters are merged to form a single cluster. When computing a distance between two clusters which contains more than two instances, an average distance of all combination of two words each from different clusters is used. This process is then repeated, until the closest distance exceeds a certain threshold.

We introduced an automatic threshold estimation method. Given all the pairwise distances between ink words, a distribution of distances is formed. We modeled this distribution as a combination of two different distributions. Pairwise distances between identical words form one distribution, and distances between different words form the other. The former distribution is assumed to be sparse and concentrated near zero. The latter one is expected to be a large-mean and high-covariance Gaussian distribution. According to our model, the threshold for stopping hierarchical clustering can be estimated by finding a point where above two distributions cross.

The threshold is estimated as below. Compute the pairwise distance between all ink words, quantitize the distances by γ, and make a distribution curve. Then, we applied a very simple method to separate two distributions. We focused near-zero area and tried to find a knee of the curve τ by approximating the curve with a line of gradient 0 to τ and a line of a constant gradient from τ on. The approximation is done by minimizing the sum of squared errors to each of the line segments. This is depicted in Figure 2.

The procedure described above has two steps. First, make the sequence D={d_1, d_2, …, d_{S1}} using first S_1 values of the distribution and sequence D'={d_1', d_2', …, d_{S2}'} with consecutive S_2 values under a constraint S=S_1+S_2 (0<S_1, 0<S_2).

Second, τ, the knee of the curve, is found by minimizing the following equation.

$$\tau = \gamma \min_S \left\{ \frac{1}{S_1} \sum_{i=1}^{S_1} (d_i - \bar{d})^2 + \frac{1}{S_2} \sum_{j=1}^{S_2} (d_j' - j\hat{a} - \hat{b})^2 \right\} \quad (1)$$

$$\bar{d} = \frac{1}{S_1} \sum_{i=1}^{S_1} d_i \quad (2)$$

$$\hat{a} = \frac{\sum_{i=1}^{S_2} i(d'_i - \bar{d})}{\sum_{i=1}^{S_2} i^2} \quad (3)$$

$$\hat{b} = \bar{d} \quad (4)$$

4. INDEX TERM SELECTION

4.1 Chi square

Once potential index terms have been identified with the hierarchical clustering, the Chi-square statistic is used to select those terms that are useful for indexing. Index terms should occur frequently in some notes and rarely in others. In other words, their distribution should be significantly different from uniform. One measure of the non-uniformity of a distribution is the Chi-square statistic. Let L be the total number of note pages and let f_i be the number of times a term occurs in the i^{th} note page. Then the total number of occurrences of the term is f, where

$$f = \sum_{i=1}^{L} f_i \quad (5)$$

If the term was distributed uniformly through the note pages, the expected number of times it would occur in each note page would be f/L. The Chi-square statistic measures the deviation from the average and is written

$$\text{Chi-square} = \sum_{i=1}^{L} \left(\frac{f_i - \frac{f}{L}}{\frac{f}{L}} \right)^2 \quad (6)$$

To select index terms, the number of terms per page is specified. In our experience, approximately 3 terms per page is appropriate. Index terms are then selected as the 3L terms with the largest Chi-square values. Alternately, index terms can be picked for each page. In this case, the number of terms per page is specified, and the Chi-square is used as above except assuming there are two note pages, the page in question and all other pages. This method has the advantage that it produces a constant number of terms per page.

5. EXPERIMENTS

5.1 Ink word clustering

We tested the above techniques on notes generated during a user study of the *Dynomite* note-taking system [7]. Users were asked to take notes on CHI videos, and answer questions based on these notes. The purpose was to evaluate the indexing capabilities of the *Dynomite* system. Notes were taken in 6 sessions over a period of one month.

Ink words in the handwritten note pages were clustered using the hierarchical clustering algorithm. To test the ink work clustering, we selected notes from three of the users. To avoid junk scribbles such as commas, periods, and dashes, some heuristics were introduced. An ink word is rejected if 1) its width/height ratio is lower than 1.5 and 2) its feature sequence length is shorter than 15.

Some terms are defined for our evaluation. A link is a connection between ink words within a cluster. Every ink word can be reached from another ink word by following links in the hierarchical clustering. Every pair of ink words has a unique link path, therefore, if there are n ink words in a cluster, the number of links is *n-1*.

An ideal link is a connection between 2 ink words that are instances of the same word. We arbitrary choose a single linkage path for an ideal group of words. Actual links are compared with ideal links as follows. If an actual link between ink words corresponds to a path of ideal links between the ink words, the link is called a true link. If not, it is called a false link. If there is a link in the ideal case and no corresponding path of links in an actual case, it is called a missed link. We defined *Precision* and *Recall* as below.

$$\text{Precision} = \frac{\text{\# of true links}}{\text{\# of all links}} \tag{7}$$

$$\text{Recall} = \frac{\text{\# of true links}}{\text{\# of ideal links}} \tag{8}$$

Precision rates and Recall rates for various thresholds are shown in Table 1. Note that Precision and Recall are low and the characteristics of the 3 users are different, indicating that the variance of human handwriting is very high.

	PL		PG		RD	
thres	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.
1.0	0.946	0.177	0.889	0.098	0.884	0.192
1.5	0.734	0.378	0.800	0.220	0.761	0.370
2.0	0.598	0.550	0.626	0.311	0.537	0.492
2.5	0.430	0.610	0.513	0.421	0.374	0.542
3.0	0.374	0.727	0.375	0.482	0.309	0.562

Table 1: Precision and Recall for various threshold

The estimated thresholds for each users and corresponding Precision and Recall are shown in Table 2. From Table 1, it can roughly be read that a threshold which balances Precision and Recall is between 2.0 and 2.5. Table 2 shows the thresholds are estimated within the range. Note user RD has lower Precision and Recall than user PL. The threshold is estimated lower for RD than PL to maintain Precision high while achieving high Recall. Table 1 and Table 2 show our method can be applied to various types of handwriting maintaining reasonable thresholds for clustering.

user	# of pages	# of valid ink words	est'd thres.	Precision	Recall
PL	14	1096	2.24	0.505	0.610
PG	16	673	2.24	0.592	0.402
RD	17	1126	2.19	0.461	0.525

Table 2: Experiment Result

5.2 Ink term selection

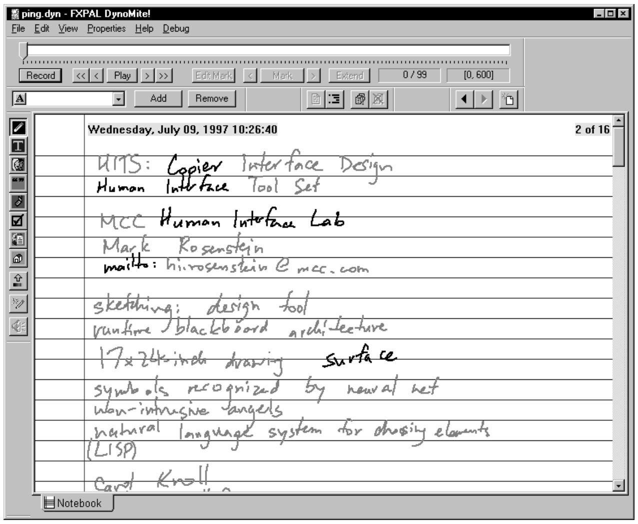
To test the automatic indexing, we selected 16 pages of notes from one of the users, PG. The notes were hand transcribed for comparison with text-based indexing methods.

Ink words in the handwritten note pages were clustered using the hierarchical clustering algorithm and an automatic threshold was selected as described above. Clusters containing short ink words were eliminated using the same heuristics described in the previous section. This resulted in 559 ink word clusters as potential index terms. In comparison, there were 573 different words in the hand transcribed notes (words with fewer than three characters were omitted).

Fifty terms were selected for indexing, based on 16 note pages and approximately 3 terms per page as described above. Index terms selected from the hand-written notes were similar to those selected from the hand transcribed text, with roughly two thirds of the terms being identical.

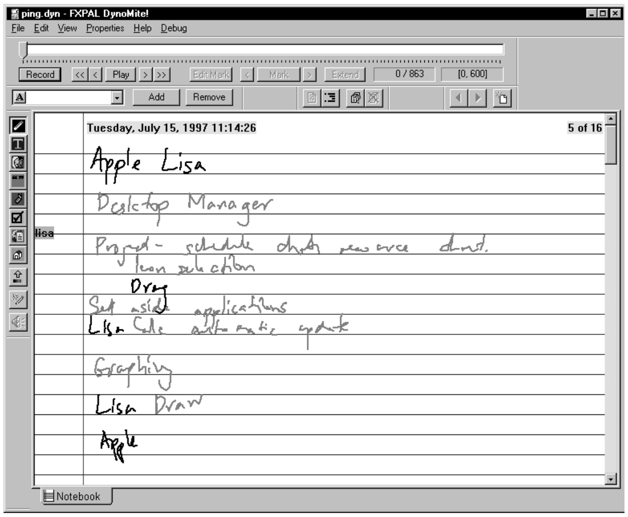
Figure 3 shows a note page with index terms in bold. The index terms for this page are “Copier”, “Human”, “Interface”, “Lab”, “surface”, and “mailto”. In comparison, the terms for this page selected based on transcribed text are “Human”, “Interface”, “surface”, “system”, and “mailto”. Figure 3 b) shows index terms for another page of notes. Here, index terms are “Apple”, “Lisa”, and “Drag”. In comparison, keywords selected from transcribed text are “Apple”, “Lisa”, “Desktop”, “selection”, and “chart”.

Figure 4 shows an index page created from the clustered index terms. Each term is labeled with the page it came from, and is hyperlinked back to the original note page. Note that it is possible for clusters to contain different words. For example, the second cluster on the index page contains the words “pointing”, “pointing”, and “painting”.


ping.dyn - FXPAL DynoMite!
File Edit View Properties Help Debug
Record << < Play > >> Edit Mark < Mark > Extend 0 / 99 [0, 600]
Add Remove
Notebook

Wednesday, July 09, 1997 10:26:40 2 of 16

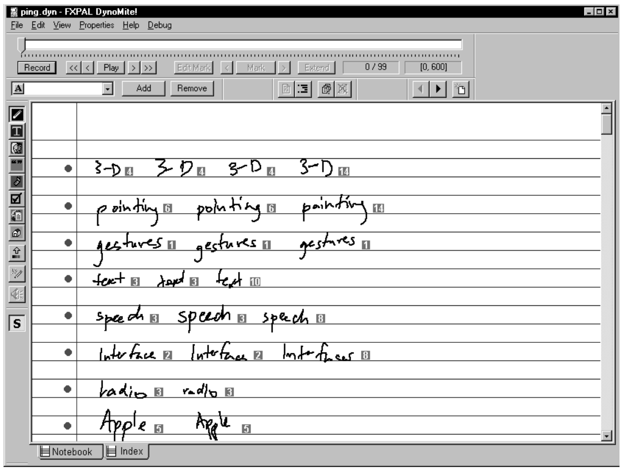
HITS: **Copier** Interface Design
Human Interface Tool Set
MCC **Human Interface Lab**
Mark Rosenstein
mailto: hi.rosenstein@mcc.com
sketching: design tool
runtime / blackboard architecture
17x24-inch drawing **surface**
symbols recognized by neural net
non-intrusive tangents
natural language system for classifying elements
(LISP)
Carol Knoll


ping.dyn - FXPAL DynoMite!
File Edit View Properties Help Debug
Record << < Play > >> Edit Mark < Mark > Extend 0 / 863 [0, 600]
Add Remove
lisa
Notebook

Tuesday, July 15, 1997 11:14:26 5 of 16

Apple Lisa
Desktop Manager
Project - schedule chart resource chart.
Icon selection
Draw
Set aside applications
Lisa Calc automatic update
Graphing
Lisa Draw
Apple

Figure 3. Index terms are shown in bold. a) top b) bottom


ping.dyn - FXPAL DynoMite!
File Edit View Properties Help Debug
Record << < Play > >> Edit Mark < Mark > Extend 0 / 99 [0, 600]
Add Remove
S
Notebook
Index

- 3-D 4 3-D 4 3-D 4 3-D 14
- pointing 6 pointing 6 pointing 14
- gestures 1 gestures 1 gestures 1
- text 3 text 3 text 10
- speech 3 speech 3 speech 8
- interface 2 interface 2 interfaces 8
- radio 3 radio 3
- Apple 5 Apple 5

Figure 4. Index page showing index term clusters and associated page numbers.

.

6. CONCLUSION

A technique for automatically creating an index for handwritten notes captured as digital ink is presented. We created a dictionary of possible index terms by clustering groups of ink strokes corresponding roughly to words. The thresholds for clustering were estimated automatically by a method introduced in this paper. We tested our method on various handwritten notes and the results indicate our method generates reasonable thresholds. Terms whose distribution varies significantly across note pages are selected for the index using Chi-square. Examples of usage for selected index terms are also illustrated.

7. REFERENCES

[1] S. Bly, L. Wilcox, P. Chiu, J. Gwizdka, “Finding Information in Handwritten Notes: A Study of Indexing,” FX PAL Technical Report, 1997.

[2] R. Manmatha, Chengfeng Han, E. M. Riseman and W.B. Croft, “Indexing Handwriting Using Word matching,” *Proc. of ACM Digital Libraries*, pp. 151--159, 1996.

[3] D. Lopresti and A. Tomkins, “On the Searchability of Electronic Ink,” Fourth International Workshop on Frontiers of Handwriting Recognition, Dec. 94, http://www.cs.cmu.edu/~andrewt/papers.html.

[4] A. Poon, K. Weber, T. Cass, “Scribbler: A Tool for Searching Digital Ink,” *Proc. of ACM CHI Companion '95*, pp. 252--253.

[5] H. Schutze, “The Hypertext Concordance: A Better Back-of-theBook Index,” Proc. COMPUTERM, ACL Coling, pp. 101--104, 1998.

[6] K. Weber and A. Poon. “Marquis: A Tool for Real-time Video Logging,” *Proc. of ACM CHI 94*, pp. 58--64.

[7] L. Wilcox, B. Schilit, N. sawhney, “Dynomite: A Dynamically Organized Ink and Audio Notebook,” *Proc. of ACM CHI 97*, pp. 186--193.

[8] P. Chiu and L. Wilcox, “A Dynamic Grouping Technique for Ink and Audio Notes,” *to be appeared in Proc. of USIT*, November, 1998.