Werner Wegstein

Zeichenerkennung und elektronische Texterfassung

1. Texterfassung

Der Nutzen des Computereinsatzes bei der Erarbeitung wissenschaftlicher Texteditionen ist heute unbestritten, die Verwendung der elektronischen Textdatenverarbeitung bedarf in den Geisteswissenschaften keiner besonderen Rechtfertigung mehr. Freilich zieht ein solcher Schritt weitere nach sich. Gerade bei längerfristig angelegten größeren Editionsprojekten zeigt sich rasch, wie notwendig es ist, auch über frühere, noch konventionell editierte Texte in elektronischem Zugriff verfügen zu können.

Ein Weg dorthin führt über die 'manuelle Texterfassung': Der Text wird über die Tastatur von Schreibkräften in das Textverarbeitungssystem eingespeichert, dabei mit den nötigen Strukturkennzeichen versehen und gegebenenfalls umkodiert, ein zeitaufwendiges und fehleranfälliges Verfahren, unabhängig davon, ob beim Abschreiben nun zu viel (Hyperkorrekturen) oder zu wenig (Schreibfehler) mitgedacht wird.

2. Optische Zeichenerkennung

Angesichts der rasant steigenden Leistung von Computern und Peripheriegeräten bietet es sich geradezu an, den Texterfassungsvorgang schneller und zuverlässiger mit elektronischen Mitteln durchführen zu lassen. Mittels eines Scanners, der optische Information in elektronische Bildinformation umwandelt, wird von dem gedruckten Text zunächst ein elektronisches Abbild in Form eines Punktmusters erstellt. Die übliche optische Auflösung heutiger Standardscanner liegt bei 400 dpi (dots per inch), d.h. eine DIN A4-Seite wird in knapp 15 Millionen Bildpunkte aufgerastert und an den Computer übertragen. Anschließend zerlegt ein Programm mit der Fähigkeit zur optischen Zeichenerkennung (Optical Character Recognition, OCR) das Bildpunktkonglomerat in Zeilen und isoliert innerhalb der Zeilen einzelne Punktmuster als potentielle Buchstaben. Mit unterschiedlichen Verfahren wird jedem dieser Buchstabenbilder der entsprechende Zeichencode des Computersystems zugeordnet: Zeichenerkennung durch Mustervergleich (pattern matching) ermittelt den Zeichencode durch den abweichungstoleranten Vergleich des Buchstabenbildes mit vorher gespeicherten Buchstabenmustern, denen in einer Trainingsphase der richtige Zeichencode bereits zugeordnet wurde; die Erkennung nach buchstabentypischen Formeigenschaften (feature recognition) extrahiert, vereinfacht ausgedrückt, aus dem Buchstabenbild Formcharakteristika, für *H* etwa zwei senkrechte Linien links und rechts, die in der Mitte durch eine horizontale Linie verbunden sind, und findet den Codewert des

Buchstabens über eine Tabelle der Formeigenschaften. Am Ende entsteht - im Idealfall - ein getreues elektronisches Abbild der gedruckten Seite, nicht mehr als Bildpunktmuster, sondern in einer Zeichencodierung, die mit anderen Programmen weiterverarbeitet, abgefragt, verglichen oder sortiert werden kann. Nebenbei reduziert sich durch dieses Verfahren auch der Speicherplatzbedarf entscheidend: Aus der Bildinformation von rund 1,5 MByte pro DIN A4-Seite wird Textinformation von etwa 3 KByte.

3. Demonstrationstext

Was für Standardtexte aus dem kommerziellen Umfeld entwickelt ist, wird auf dem Sektor wissenschaftlicher Texteditionen oder älterer Textausgaben schnell mit unerwarteten Anforderungen konfrontiert. Dies soll an einem einfachen Beispiel aus der deutschen Philologie demonstriert werden, den "Sieben Staffeln des Gebetes" des David von Augsburg.[1] Zur Demonstration habe ich Seite 52 ausgewählt (siehe hier auf der folgenden Seite).

Formatprobleme bietet der einspaltige Text kaum: Die Zeilenzählung steht durchgängig am linken Rand, die Schrift ist klar strukturiert, Ligaturen sind nicht verwendet. Anders verhält es sich mit dem Zeichensatz: Zwar sind die (sechs) ú noch durch ein Zeichen aus dem erweiterten IBM-Zeichensatz abzubilden (Nr. 163 auf Codepage 437 oder 850), dies gilt jedoch nicht mehr für die fünf v́, und schon gar nicht für die 17 Stellen, an denen Kleinbuchstaben (o̊, o̊, v̊, v̊) übergeschrieben sind, ebensowenig für geschwänztes ȝ (dreimal) und sogenanntes Schaft-s ſ (41 Belege). Fontprobleme bereitet schließlich die Verwendung der Kursive nicht nur für ganze Wörter, sondern auch in einzelnen Buchstaben am Anfang (91 *e*nm v̊ze), innerhalb (95 bestu*mpf*eten) und am Ende (102 betten*t*) von recte gesetzten Wörtern. Der Kommentar ist recte und kursiv in kleinerem Schriftgrad gesetzt und verwendet für Namen als weiteren Font Kapitälchen.

4. Forderungen an das OCR-Programm

Aus dem Format-, dem Zeichensatz- und dem Fontbefund der Textvorlage lassen sich die Anforderungen an ein OCR-Programm ableiten. Es versteht sich von selbst, daß das Programm Seiten- und Zeilenformat (auch bei mehrspaltigem Satz) nicht verändert. Dazu rechne ich auch die sichere Erkennung und Einhaltung der Wortgrenzen, selbst in schwierigeren Fällen, in denen der Schriftraum des auslautenden Buchstabens mit dem des nachfolgenden anlautenden sehr eng zusammengerückt sein kann, wie bei der Abfolge von *f* und *j*.

---

[1] Die Textausgabe hat seinerzeit neue Editionsvorstellungen verwirklicht: David von Augsburg: Die sieben Staffeln des Gebetes, in der deutschen Originalfassung hg. von Kurt Ruh (Kleine Deutsche Prosadenkmäler des Mittelalters 1), München 1965. Vgl. auch Kurt Ruh: Votum für eine überlieferungskritische Editionspraxis. In: Probleme der Edition mittel- und neulateinischer Texte. Kolloquium der Deutschen Forschungsgemeinschaft, Bonn 26.-28. Febr. 1973. Hg. von Ludwig Hödl und Dieter Wuttke. Boppard 1978, S. 35-40.

die man vf gat in die porte, das ſint ſiben offenvnge ze der vollecomenheit des gebettis. [Swie ir vil lúzel ſint, die ze den allen hie koment, das ioch die vollekomeneſten ſelten ze *dem* ſehſten comen oder zv̊ dem fv́nften, ir iſt ǒch gnv̊g lúzil, die v́bir den anderen komen, ǒch vnder geiſtlichen lúten. Vnde das noch me ze wunderen iſt: ir iſt gar lúzil ǒch vnder gelerten lúten, die virſten kv́nnen, was ez iſt, *[150^{vb}]* oder ioch gelouben wellen, daz ez iſt, ſwie doch dv́ heilige ſchrift da von *an* mengen enden vil beſcheidenliche lert der *ez* virſten kan[15].]

Der erſte grat des gebettes iſt das genǒte gebet mit dem mvnde[16] [, des den menſchen niht ſo vil lvſtit, er enmv̊ze ſich dar zv̊ vlizen, das er es ſpreche, es ſin die tage zit oder ander geſezzede oder ſelbe genomens gebet]. Da hǒrent drv́ dink zv̊ [, das es ze ſime rehte kome]: Das eine iſt, das man es dvrnehtecliche ſpreche [, niht mit beſtu*mpf*eten worten von gacheit, niht ſlaflichen von tracheit, niht verlazenlichen von italkeit[17]]. Das ander iſt, das man ez dvrch got tv̊, niht dvrch lop der lúte [, niht mit geli*ch*ſenheit] vnde vmbe menſchen lon [, alſe die da wellent, daz man in deſte gerner gebe]. Das dritte, das das herce mit den worten gehelle das der mvnt bittet, das der gedank da bi ſi, alſe ǒch ſante Bernhart[18] leret: 'So ir bettent,

[15] *Zu 79 ff vgl. Comp. III, c. 65, n. 3, S. 354 f.*
[16] *Vgl. zum Thema Comp. III, c. 53, S. 296 ff (De tribus modis orandi, et primo de vocali oratione) und Dav. v. A.,* Pf. *I, S. 324,38 ff.*
[17] *Vgl. Bernhard, Serm. in Cant. 47, n. 8 (*Migne *PL. 183, Sp. 1011): Strenue quidem, ut sicut reverenter, ita et alacriter Domino assistatis: non pigri, non somnolenti, non oscitantes, non parcentes vocibus.*
[18] *ebd. Pure vero, ut nil aliud dum psallitis, quam quod psallitis cogitetis.*

Die Kursivierungen im Editionstext markieren Eingriffe des Editors in die Textgestalt der Leithandschrift als Resultat der textkritischen Untersuchung der Überlieferung. Sie sind im Variantenapparat dokumentiert und erlauben die diplomatische Rekonstruktion der handschriftlichen Lesart. Daher genügt es hier nicht, wenn das OCR-Programm die Buchstabenformen des kursiven Fonts richtig erkennt, der Fontwechsel sollte auch in der Ausgabedatei entsprechend markiert sein, und zwar nicht nur bei den vollständig kursivierten Wörtern, sondern auch für die Einzelbuchstaben. Ferner muß das Programm hoch- oder tiefgestellte Zeichen als solche erkennen und abbilden können, und es muß in der Lage sein, zusammenhängende Buchstaben zu

verarbeiten, gleichgültig, ob die Kombination als Ligatur Bestandteil des Schriftfonts ist (wie im allgemeinen *ff*, *fi*, *fl*) oder erst durch den Druck oder Satz zustande kommt (wie auf der Musterseite *ch*). Unabhängig vom vorwiegend eingesetzten Erkennungsverfahren (Mustervergleich gegenüber Erkennung nach buchstabentypischen Formeigenschaften) muß es möglich sein, dem Programm Buchstabenformen sowie Buchstaben- und Akzentkombinationen anzutrainieren, die nicht zum Standardzeichensatz gehören (bis hin zu ganzen Fonts in einem nicht-lateinischen, also z.B. dem griechischen oder kyrillischen Alphabet).

Kommt das Programm mit der Fontproblematik nicht zurecht, so bedeutet dies aufwendige und fehleranfällige Nachkorrekturen an sieben Stellen der Musterseite. Kann das Programm Buchstaben, die nicht zum Standardzeichensatz gehören, nicht verarbeiten und liefert es statt dessen ein Rückweisungszeichen für jeden nicht erkannten Buchstaben, so ist es für die Bearbeitung des Demonstrationstextes kaum mehr brauchbar, denn bei 71 Rückweisungen auf 25 Zeilen ist der Korrekturaufwand indiskutabel.

Das für die Demonstration zur Verfügung gestellte Programm Optopus in der Version 2.5, mit dem wir auch an der Universität Würzburg seit einigen Jahren arbeiten, erfüllt die gesetzten Anforderungen, freilich mit einigen Einschränkungen. Das Programm basiert primär auf dem Verfahren des abweichungstoleranten Mustervergleichs und setzt infolgedessen eine Trainingsphase voraus, in der die einzelnen Buchstabenmuster für jeden Schriftfont eigens trainiert werden müssen. Während des Trainings zeigt sich meist schon rasch, welche Buchstaben und welche Eigenheiten von Schriftfont oder Satz bei der Erkennung Schwierigkeiten bereiten oder zu falschen Codezuordnungen führen. Hier kann der Systembenutzer durch die Veränderung von Parametern die Erkennungsleistung und Erkennungssicherheit auf die jeweilige Vorlage individuell abstimmen.

Beim Training der Musterseite fallen vier Problemfälle auf: die hochgestellte kleine *8* der Anmerkungsziffer *18* wurde als kleines *s* identifiziert; der eingestellte Wortabstand erwies sich als zu groß, da an einigen Stellen die Wortgrenze nicht erkannt wurde; in dem kursiven Font der Anmerkungen wurde das *e* in *Thema* als *c* identifiziert; in Anmerkung 17 wurde die *1* in der Spaltenangabe *1011* als *I0II* gelesen. Durch Eingriffe bei Trainingsmustern und Veränderungen von Steuerungsparametern lassen sich diese Phänomene im allgemeinen korrigieren. Mit einigen Kniffen gelingt es auch, die Kursivierungsinformation bei einzelnen Buchstaben eines Wortes in die Ausgabedatei zu übertragen.

Es verdient festgehalten zu werden, daß im Editionstext kein Fehler auftrat und die Fontinformation in der Ausgabedatei korrekt abgebildet wurde. Mehrfaches Scannen der Demonstrationsseite und anschließendes Lesen mit den gleichen Trainingsmustern führen stets auf das gleiche Ergebnis in der Ausgabedatei.

5. Arbeitsweise und Erkennungsrate

Die Arbeitsweise des Programms und die Erkennungsrate beeinflussen unmittelbar die Entscheidung, ob sich der Einsatz des Scanners gegenüber manueller Erfassung lohnt. Ich beschränke mich hier auf die Eigenschaften des Demonstrationsprogramms

Optopus. Es kann nach dem Training entweder interaktiv oder in einem automatischen Arbeitsmodus eingesetzt werden. Interaktiv bedeutet, daß jemand, der mit der Bedienung des Programms vertraut ist, den Erkennungsvorgang am Bildschirm mitverfolgt und auf Unregelmäßigkeiten entsprechend reagiert. Bei Buchstaben, die nicht sicher erkannt werden, hält das Programm an, zeigt auf dem Lesebildschirm den entsprechenden Buchstaben, dazu die Zeichenumgebung aus der Bilddatei, eventuell auch noch einen Identifizierungsvorschlag. Mit diesen Informationen kann ohne umständliche Suche in der gedruckten Textvorlage sofort zuverlässig entschieden werden, ob der Lesevorschlag richtig ist oder über die Tastatur korrigiert werden muß. Sofern es sich bei der Problemstelle nicht nur um eine Störung durch Verschmutzung oder um einen zerbrochenen Buchstaben handelt, wenn also beispielsweise ein noch nicht oder nicht ausreichend trainierter Buchstabe vorliegt, so kann dieser gleich für den weiteren Erkennungsprozeß als Vergleichsmuster noch nachtrainiert werden. Solange das System nicht allzu oft pro Seite anhält, ist solches interaktives Arbeiten effektiv und akzeptabel. Nebenbei kommt das Mitlesen einer Plausibilitätskontrolle gleich, so daß ein auf diese Weise erfaßter Text - nach meinen Erfahrungen - schon recht fehlerarm sein dürfte.

Bei automatischem Arbeiten fügt das Programm an Problemstellen ein Rückweisungszeichen ein, das anschließend in einem separaten Arbeitsgang (z.B. in Verbindung mit der Nachbearbeitung) korrigiert werden muß. Je nachdem wie bedienungsfreundlich der Korrekturvorgang ausgeführt werden kann, entscheidet sich, ob der Einsatz des OCR-Programms sich vom Zeitaufwand her lohnt. Eine auf den ersten Blick respektable Erkennungsrate von 95% bedeutet beispielsweise, daß auf einer DIN A4-Seite mit rund 2000 Zeichen 100 Korrekturen anfallen.[2]

6. Fehlerverhalten und Nachbearbeitung

Von Fehlern war bislang noch gar nicht die Rede. Ich bin auch überzeugt, daß unter Idealbedingungen gute OCR-Programme heute schon nahezu fehlerfrei arbeiten. Nur kommen in der Praxis solche Idealbedingungen leider kaum vor. Zerbrochene oder ineinanderlaufende Buchstaben, welliges Papier, das schon beim Scan-Vorgang Verzerrungen erzeugt, Vergilbungen und Verschmutzungen, wechselnder Sättigungsgrad der Druckfarbe, von der Rückseite durchscheinende Buchstaben, zu enger Zeilendurchschuß, problematische Schriftschnitte, von Frakturschrift ganz zu schweigen, - die Liste der Faktoren, die den Erkennungsvorgang beeinträchtigen, ließe sich leicht noch weiter fortsetzen. Sie machen die zuverlässige Unterscheidung von Buchstaben, die ohnehin nur in minimalen Details von einander abweichen, sehr schwierig: *I*, *l* und *l*, *c* und *e* (erst recht wenn kursiv), *h* und *b*, *n* und *u*, je nach Erkennungsverfahren auch *B*, *8* und *ß*. An solchen schwierigen Stellen, über die ein Leser dank der Unterstützung durch den Kontext, praktisch ohne es zu merken, einfach hinwegliest, kann ein OCR-Programm nur die optische Information auswerten, und wo diese ver-

---

[2] Zum Vergleich: Der Bundesverband Druck rechnet für die Erfassung von 1000 Zeichen (einfacher Fließtext) 8,5 Minuten, davon 5,9 Minuten für das Schreiben, 2,6 Minuten für das Korrigieren. Zitiert nach: Makrolog newsletter 1/90, S. 8.

zerrt oder gestört ist, liefert es im günstigsten Fall ein Rückweisungszeichen, im ungünstigsten kommt es zu einer Fehlzuordnung, weil eben nicht mehr erkannt werden kann, als vorhanden ist. Einige OCR-Programme versuchen, durch die Integration eines sprachabhängigen Lexikons die Erkennungsleistung hier zu verbessern. Daraus können sich freilich wieder neue Probleme ergeben. Für historische Texte, die der orthographischen Normierung nicht entsprechen, für Sprachmischungen und für viele europäische, nicht-englische Fremdsprachen ist damit ohnehin wenig gewonnen, weil es hierfür solche Korrekturwörterbücher (noch) nicht gibt und ebensowenig Rechtschreibkorrekturprogramme, die für den gleichen Zweck verwendbar wären.

Eine Nachbearbeitung auch des mit einem OCR-Programm erfaßten Textes ist in jedem Fall nötig, wenn es darauf ankommt, mit fehlerfreien Texten zu arbeiten. Denn druckbedingte Lesefehler sind bei keinem OCR-Programm von vornherein auszuschließen. Wenn der Text allerdings schon einmal in elektronischer Form vorliegt, bieten Programmsysteme wie TUSTEP zahlreiche Hilfen (Wortregister, Konkordanzen, Textvergleich, Zeichengruppenabgleich und anderes mehr), solchen Fehlern auf die Spur zu kommen.[3]

Schließlich bleibt festzuhalten, daß - jedenfalls im wissenschaftlichen Bereich - die von einem OCR-Programm eingelesene und von Fehlern bereinigte elektronische Textfassung keineswegs schon identisch ist mit einer 'elektronischen Edition'. Dazu sind noch weitere Überlegungen und Arbeitsschritte nötig, die hier nicht mehr diskutiert werden sollen.

7. Ausblick

Die optische Zeichenerfassung hat in den letzten Jahren beachtliche Fortschritte gemacht. Dies dokumentieren für den wissenschaftlichen Bereich überzeugend David Birnbaums Grundsatzreferat "Optical Character Recognition & Non-Latin Alphabets" und sein Erfahrungsbericht über das Einlesen kyrillischer Texte mit Makrologs Optopus (Version 2.3R) im Vergleich zu dem System K5100 von Xerox Imaging Systems, hier eher bekannt unter dem Namen 'Kurzweil'.[4] Auch in Würzburg sind unsere Erfahrungen mit dem Optopus-System 2.5 rundum positiv, wenn man seine Leistungsgrenzen respektiert.

Trotzdem hat die Anwendung der OCR-Programme auf unkonventionelle wissenschaftliche Texte mit komplexer Satzstruktur vielfach noch immer starke Züge des Experimentellen an sich.[5] Daß dies nicht nur für den wissenschaftlichen Anwen-

---

[3] Vgl. Wilhelm Ott: Edition und Datenverarbeitung. In: Herbert Kraft: Editionsphilologie. Mit Beiträgen von Jürgen Gregolin, Wilhelm Ott und Gert Vonhoff. Darmstadt 1990, hier S. 66-68.

[4] Bits & Bytes Review. Reviews & News of Products & Resources for Academic Computing. Vol. 2, Num. 6-7, Summer 1991, S. 22 - 36.

[5] Zu dieser Einschätzung kommt auch: The Humanities Computing Yearbook 1989-90. A Comprehensive Guide to Software and other Resources von Ian Lancashire. Oxford 1991, S. 540: "...techniques have been refined considerably in the past five years and have now reached an acceptable level of reliability for many clerical purposes. To satisfy needs of researchers in non-English languages, on the other hand, vendors have some way to go yet".

dungsbereich gilt, zeigt der jüngste mir bekannte Vergleich von OCR-Programmen aus dem kommerziellen Umfeld, der Schreibmaschinentext, Zeitungs- und Katalogseiten als Testmaterial verwendet und zu einem fast schon deprimierenden Schluß kommt.[6]

Wer nicht enttäuscht werden will, tut deshalb gut daran, vorher auszuprobieren, wie es mit der OCR-Tauglichkeit der zu bearbeitenden Textvorlagen bestellt ist, damit später keine unliebsamen Überraschungen auftreten.

Nun wird freilich allenthalben intensiv an neuen, verbesserten Programmversionen unter neuen Oberflächen und Betriebssystemen gearbeitet: 'Optopus goes Windows' (Makrolog newsletter 1/92) und die Xerox Imaging Systems bzw. 'Kurzweil' bieten mit ScanWorX bereits eine OCR-Software-Lösung für Unix-Systeme von Sun und IBM RS/6000 mit erstaunlichen Leistungseigenschaften. So ist es durchaus vorstellbar, daß beim Erscheinen dieses Beitrags bereits ein weiterer Durchbruch in der OCR-Technologie erzielt worden ist, der das Experimentelle des Verfahrens wieder ein Stück weiter zurückdrängt. Bis dahin bleibt es dabei: Ausprobieren!

---

6 Christiane Parusel/Ulrich Stühlen: OCR-Programme. Text(v)erkennung: In: PC Professional. Oktober 1992. S. 216-258. Hier S. 254: "Keines der Programme erfüllt vollständig die Erwartungen, die wir an ein Lesesystem zur Erfassung von Manuskripten stellen. Obwohl die Leistung der OCR-Software schon wesentlich besser ist als noch vor zwei Jahren, sind Fehlerquote und erforderliche Nacharbeit oft an der Grenze des Sinnvollen. ... Bei der Bewertung stand daher immer im Vordergrund, ob eine nennenswerte Zeitersparnis gegenüber einer manuellen Texterfassung besteht." Von den getesteten Programmen (Catchword 1.21, GO-OCR 2.0, LI-OCR VII for Windows, Omnipage Professional 2.0, Prolector 1.12, Readiris 2.0, Recognita Plus 1.2 International, Wordscan Plus 1.1c) erreichen nur Omnipage Professional, Wordscan Plus und GO-OCR trotz gewisser Einschränkungen eine lobenswerte Erwähnung. Die Empfehlung der Redaktion schließt mit der Feststellung: "In den meisten Fällen ist das manuelle Erfassen den OCR-Systemen noch vorzuziehen. Als preiswerte Alternative empfiehlt sich ein Lernprogramm zum Blindschreiben".