

Jahresschrift für mitteldeutsche Vorgeschichte	80	S. 325 - 338	Halle (Saale)	1998
--	----	--------------	---------------	------

## Externe Eichung für das geographische Informationssystem VISOR

Thomas Richter, Halle (Saale)

Der Einsatz von geographischen Informationssystemen (Geoinformationssystemen, GIS) gewinnt immer mehr an Bedeutung. Auch in der Archäologie werden verstärkt GIS-Systeme eingesetzt<sup>1</sup>.

Ein Anwendungsbereich von GIS ist die Darstellung von Sachdaten mit räumlichem Bezug auf geeignetem digitalem Kartenmaterial (Mapping). Im Landesamt für Archäologie - Landesmuseum für Vorgeschichte - Sachsen-Anhalt (LfA) wurden fünf Lizenzen des dem Desktop Mapping zuzuordnenden Programms VISOR angeschafft. VISOR ist momentan sowohl in einer NEXTSTEP- als auch in einer MS-Windows-Version verfügbar. Dies gab den Ausschlag für die Wahl dieses hybriden Raster/Vektor-GIS-Systems zum Einsatz im heterogenen Computer-Netzwerk des LfA. Zudem ist eine hohe Flexibilität bei der Auswahl der Quellen anzuzeigender Daten durch den konsequenten Client/Server-Aufbau des Systems gegeben. In den nachfolgenden Ausführungen ist mit der Bezeichnung VISOR stets der VISOR-Viewer der NEXTSTEP-Version des GIS-Systems gemeint, auch wenn eine Reihe der Aussagen ebenfalls für die Windows-Version zutrifft<sup>2</sup>. Die komplexe Vorgehensweise der Erstellung einer blattschnittfreien Kartengrundlage für ganz Sachsen-Anhalt auf Basis der Topographischen Karte im Maßstab 1 : 25 000 (TK 25) beschrieb schon R. Weise<sup>3</sup>.

VISOR wird im LfA derzeit zur Visualisierung von Luftbildfundstellen eingesetzt sowie zur Überprüfung von Fundstellenkoordinaten im Rahmen der Schnellerfassung archäologischer Denkmale.

### Kurze Übersicht zum Aufbau von VISOR

VISOR bietet die Darstellung von Vektorobjekt-Layern. Diese kann man sich als durchsichtige Folien mit darauf projizierten geometrischen Objekten wie Punkten, Linien, Kreisen, Rechtecken, Polylinien, Polygonen, skalierbaren Texten und skalierbaren Icons mit unterschiedlichen Attributen wie Füll- und Umrandungsfarbe, Transparenz, Schraffur, Strichstärke und Linientyp vorstellen.

Den Hintergrund bilden nach Maßstab gestaffelte (nach Bedarf ein- und ausblendbare) Rasterkarten. Dieser Kartenhintergrund kann aus Einzelkarten, aber auch aus zu einer blattschnittfreien Gesamtkarte zusammengeführten Aneinanderreihung von Einzelkarten, aufgebaut sein. Da die im Speicher zu haltenden Datenmengen bei Rasterkarten die Speicherausstattung üblicher Rechnersysteme weit übersteigen können, wird die angezeigte Rasterkarte in eine große Anzahl kleinerer Bilddateien geringer Dateigröße geschnitten (gekachelt), die beim Scrollen im Karten-Fenster von VISOR bei Bedarf durch eine intelligente Softwaresteuerung nachgeladen werden. Erreicht wird ein minimaler Hauptspeicherbedarf im Computer, und beim Laden der Kacheln von einem Netz-

werk-Server bleibt die Netzwerkbelastung gering. Folgende Kartenbasen sind bereits vorhanden bzw. im Aufbau:

1. Eine blattschnittfreie Übersichtskarte Sachsen-Anhalts im Maßstab 1 : 300 000 (farbig), die vom Landesvermessungsamt Sachsen-Anhalt bezogen wurde. Die Einbindung in VISOR ist abgeschlossen.
2. 28 Kreiskarten im Maßstab 1 : 50 000 (farbig) wurden von der Firma Geospace GmbH Jena jeweils im Stück mit einem A0-Trommelscanner eingescannt und als einzelne Kreiskarten in VISOR integriert.
3. Erzeugung einer blattschnittfreien Gesamtkarte Sachsen-Anhalts im Maßstab 1 : 25 000 (monochrom) auf Basis der über 200 TK-25-Meßtischblätter<sup>4</sup>. Der Realisierungszeitraum ist offen.

#### Arbeitsschritte beim Aufbau einer digitalen Rasterkartenbasis

Die Erzeugung der unter 1. und 2. beschriebenen Rasterkarten-Basen ist zwar zeitlich aufwendig, aber nach Bewältigung einiger softwaretechnischer Probleme unproblematisch zu realisieren, da die in VISOR einzubindenden Karten bereits als 90 bis 650 MB große Einzelbilder vorliegen und nach Dateiformatkonvertierungen und einer Helligkeitsoptimierung gekachelt werden können. Problemfaktoren sind die Vielzahl von TIFF-Bildformaten und die Algorithmen zur Komprimierung der Bilddateien sowie die Softwarebibliotheken zur Konvertierung und Anzeige der Bilddaten. Da die Routinen zur Bildanzeige vom VISOR-Entwickler megatel GmbH vorgegeben waren, wird nach geeigneten Programmen zur Farbreduktion und Helligkeitsanpassung von TIFF-Bilddateien unterschiedlicher Herkunft gesucht. Nach umfangreichen Tests konnten alle Softwarekomponenten ermittelt und die Vorgehensweise zum Erzeugen der Kartenbasen festgelegt werden.

Eine Kartenbasis im Maßstab 1 : 25 000 (Punkt 3) ist schwieriger zu erzeugen, kann aber allmählich aufgebaut werden. Zuerst wird die benötigte Verzeichnisstruktur für die Kartenbasis aufgebaut, und in diese werden nach und nach die einzelnen Kacheln hineinkopiert. Dieses erlaubt ein gebietsweises Aufbauen der Gesamtkarte, z. B. kann in VISOR das zuerst bearbeitete Kartenareal des Saalkreises eingesetzt werden.

#### Probleme beim Einsatz der auf der TK 25 aufbauenden Rasterkartenbasis mit VISOR

Die Erstellung einer auf der TK 25 basierenden Gesamtkarte Sachsen-Anhalts mit den im LfA vorhandenen Hilfsmitteln wirft Probleme auf, die ausführlich von R. Weise beschrieben wurden<sup>5</sup> und nur bedingt zu bewältigen sind. Dies betrifft:

1. das Zusammenfügen von Kartenvierteln nach dem Scannen zum einzelnen Kartenblatt (da nur jeweils ein Viertel eines Kartenblatts gescannt werden kann) und der Bildung von Unstetigkeiten an den Berührungskanten der Kartenviertel,
2. das Wegschneiden von Kartenrändern mit einem geeigneten Bildbearbeitungsprogramm,
3. das Aneinanderfügen von über 200 trapezförmigen Kartenblättern zu einer Gesamtkarte, bei dem Kompromisse eingegangen werden müssen,

4. die Bildung nichtlinearer Verzerrungen während des Scan-Vorgangs, die nur durch teure Spezialsoftware in zeitaufwendigen Arbeitsgängen kompensiert werden könnten,
5. den hohen Zeitaufwand von geschätzten 875 Arbeitstagen für die Erzeugung der Kartenbasis.

Es entstehen einerseits unschöne, manchmal gut sichtbare Unstetigkeiten an den Berührungsstellen zwischen Kartenvierteln innerhalb eines Einzelblattes und der Kartenblätter untereinander. Andererseits treten nichtlineare Verzerrungen innerhalb der Gesamtkarte auf. Mathematisch gesehen, wird das orthogonale Gauß-Krüger-Koordinatensystem der resultierenden Gesamtkarte an den Übergangsstellen sprunghaft, sonst flächenweise nichtlinear verzerrt. Dies ist mit den in VISOR standardmäßig integrierten Eichroutinen nur unzureichend auszugleichen, selbst bei Ermittlung einer großen Anzahl von Eichpunkten auf der Gesamtkarte. Als Eich-(Referenz-)Punkte werden u. a. Schnittpunkte des auf der Karte eingezeichneten Koordinatennetzes verwendet.

Die Eichung (Georeferenzierung) der Karte in VISOR, bei der einzelnen Bildpunkten auf der Karte als Referenzpunkte entsprechende Geo-Koordinaten (z. B. Gauß-Krüger-Koordinaten) zugewiesen werden und die von einer idealen Kartenbasis mit orthogonalem Koordinatensystem ausgeht, zeigt das entstehende Problem. Die von VISOR an einer von den Referenzpunkten abweichenden Stelle auf der Karte berechneten Koordinaten können mehr oder weniger von den Sollwerten abweichen. Der Vergleich von durch VISOR berechneten Koordinaten von Punkten auf der Karte mit bekannten Geo-Koordinaten erbrachte in Tests Abweichungen von teilweise mehr als 50 Metern, vor allem an den Unstetigkeitsstellen.

Ein Ansatz zur Lösung dieser Probleme, das Erzeugen einer "idealen" Gesamtkarte unter Einsatz von Transformations- und verschiedener Bildbearbeitungsprogramme, wurde wegen des zu erwartenden extrem hohen Personal-, Zeit- und Kostenaufwands schnell verworfen.

Ansatz zu einer genaueren Koordinatenberechnung durch Einsatz einer externen Eichroutine

Aus den vorher genannten Gründen wurde eine andere Vorgehensweise durchgeführt. Anstatt die Kartengrundlage zu optimieren, wird die Berechnung der Koordinaten auf der Karte im GIS-VISOR angepaßt. Die zusammengestellte Gesamtkarte weist nach wie vor (manchmal gut sichtbare) Unstetigkeitsstellen an den Kartenübergängen sowie nichtlineare Verzerrungen durch Alterung des Papiers der Kartenblätter und durch den Scanprozeß auf. Die geringen nichtlinearen Verzerrungen sind jedoch durch das Auge beim Betrachten der Hintergrundkarte in VISOR gar nicht wahrnehmbar. Dies ergibt, wie Abb. 1 stark übertrieben verdeutlicht, mehr oder weniger große Abweichungen im Koordinatensystem auf der VISOR-Hintergrundkarte, die durch eine sogenannte externe Eichroutine kompensiert werden müssen - extern deshalb, weil die internen Georeferenzierungs-Methoden des VISOR-GIS, die "ideales" gescanntes Kartenmaterial mit einem exakt orthogonalen Koordinatensystem voraussetzen, durch eine eigene (noch näher zu spezifizierende) Georeferenzierungsmethode ersetzt werden müssen



Abb. 1: Stilisierte Darstellung des eingezeichneten Koordinatensystems auf einer Karte mit nicht-linearen Verzerrungen sowie einer Unstetigkeitsstelle durch Zusammenfügen von Kartenblättern im Computer

Die digitalisierte Gesamtkarte besteht aus einer großen Anzahl von Bildpunkten (Pixel). Bei der Koordinatenberechnung sind jedem Punkt in diesem Pixel-Koordinatensystem (z. B. Punkt 1 000 in horizontaler und gleichzeitig Punkt 500 in vertikaler Richtung) die entsprechenden Geo-Koordinaten (Rechts- und Hochwert) zuzuweisen.

Auf den TK-25-Blättern ist ein Koordinatennetz im Gauß-Krüger-Koordinatensystem (Bessel Ellipsoid) mit einem Linienabstand von 1 000 Metern eingezeichnet. Dieser Linienabstand beträgt 4 cm auf dem Kartenblatt. Die Schnittpunkte der Gitterlinien können als Eichpunkte verwendet werden, da ihre Gauß-Krüger-Koordinaten einfach berechnet werden können. Auf diese Weise erhält man ein dichtes Netz von Referenzpunkten mit genau bekannten Koordinaten (Pixel-Koordinaten und Geo-Koordinaten) für die Berechnung jedes beliebigen Punktes auf der digitalisierten Karte.

Nachteilig ist die begrenzte Einsetzbarkeit einer externen Eichroutine auf ein bestimmtes GIS-System, in diesem Fall VISOR, und auf eine Rechnerplattform (NEXTSTEP), infolge der Spezifika der vorgegebenen Programmierumgebungen. Dafür erhält man hohe Flexibilität beim Einsatz unterschiedlich gut präparierten Kartenmaterials.

Ende 1994 verhandelte das LfA erfolgreich mit der Firma megatel GmbH Bremen und konnte erreichen, daß in die NEXTSTEP-Version von VISOR eine Schnittstelle für externe Koordinaten-Berechnungsroutinen eingebaut wurde.

Im späten Frühjahr 1995 erfolgten die Programmierarbeiten für eine externe Eichroutine durch den Verfasser.

#### Allgemeine Anforderungen an eine externe Eichung (Koordinatenberechnung)

An eine externe Eichroutine (sie entspricht einer Geo-Koordinaten-Berechnungsroutine) werden im LfA verschiedene Anforderungen gestellt, die im folgenden dargelegt werden.

Das LfA besitzt momentan 13 NeXT-Computer, von denen insgesamt sieben Colorstations und Cubes farbfähig sind und für die Kartenarbeit mit VISOR eingesetzt werden können. Diese im Jahre 1991 gekauften Maschinen sind mit 25 MHz M68040- Prozessoren ausgerüstet und den Leistungsanforderungen moderner Programmsysteme sowie dem Leistungsbedarf aktueller NEXTSTEP-Versionen nur noch bedingt gewachsen. Beim Arbeiten mit VISOR wird während des Ladens eines Objektlayers in Abhängigkeit von der Anzahl und des Typs geladener Objekte und beim Bewegen des Mauszeigers ständig eine hohe Zahl von Koordinatenberechnungen durchgeführt. Die externe Berechnungsroutine für Koordinaten muß deshalb so effizient wie möglich ausgelegt sein, um einerseits Wartezeiten beim Laden der Objektlayer im erträglichen Rahmen zu halten, andererseits bei Mausbewegungen in Echtzeit die Koordinaten für die Stellen, über die sich der Mauszeiger bewegt, berechnen zu können. Besonders beim Laden von Layern mit vielen tausend Objekten und/oder komplexer Struktur können sonst Wartezeiten von mehreren Minuten auftreten, die Programmanwendern nur bedingt zuzumuten sind.

Neben den älteren NeXT-Rechnern besitzt das LfA mehrere PC mit 486DX2/66- und Pentium 60 bis 120 MHz-Prozessoren, die mit dem NEXTSTEP-Betriebssystem ausgerüstet sind. Die Koordinatenberechnung muß auch für NEXTSTEP für Intel-Prozessoren verfügbar sein.

Da die Berechnungsroutinen zeitkritisch sind (in Echtzeit ablaufen und keine Verzögerungen im Programmablauf verursachen dürfen) und eine eventuelle Portierung des Programmcodes auf andere Rechnersysteme zu ermöglichen ist, wurde bei der Programmierung auf den Einsatz bequemer, nichtportabler und von den Laufzeiteigenschaften her langsamer Mittel der objektorientierten Programmierung und moderner hochfunktionaler Klassenbibliotheken (unter NEXTSTEP dem Objective C-Runtime-System und der Klassenbibliothek Application Kit) weitgehend verzichtet. Stattdessen wurden nahezu alle ProgrammROUTINEN (mit Ausnahme der Schnittstelle zu VISOR) in ANSI C entwickelt. Es wurde die schnellere, aber auch fehlerträchtigere strukturelle Programmierung von Standard C eingesetzt. Zudem ist bei zeitkritischen Schleifenoperationen der Einsatz von Funktionsaufrufen (mit ProgrammROUTINEN, die innerhalb der Schleife ablaufen) vermieden worden. Teile des Quellcodes werden dadurch schwer überschaubar, und der Code ist fehleranfälliger und schwerer zu pflegen. Lohn der Mühe ist ein schneller Maschinencode.

Außerdem kann durch intelligente Programmierung, wie zum Beispiel dem Einsatz von Caching-Mechanismen bei zeitkritischen und komplexen Berechnungsroutinen Rechenzeit eingespart werden (u. a. zu Lasten eines erhöhten Speicherbedarfs).

#### Auswahl einer einfachen mathematischen Berechnungsprozedur für Geo-Koordinaten

Gesucht wurde eine Methode, die wenig Rechenaufwand nach sich zieht und einfach in der Programmiersprache C zu implementieren ist, ohne spezielle mathematische Bibliotheken zu benötigen.

Wie bereits dargelegt, kann eine Reihe von Referenzpunkten (Eichpunkten) auf der Karte unter Nutzung des eingezeichneten Koordinatengitters festgelegt werden. Jeder beliebige Punkt, für den Geo-Koordinaten zu berechnen sind, liegt dann entweder:

1. zwischen vier der Referenzpunkte ( $P_1$ ),
2. auf der Linie zwischen zwei Referenzpunkten, was als Spezialfall von 1 angesehen werden kann ( $P_2$ ),
3. genau auf einem der Referenzpunkte ( $P_3$ ),
4. außerhalb des Netzes von Referenzpunkten ( $P_4$ ).

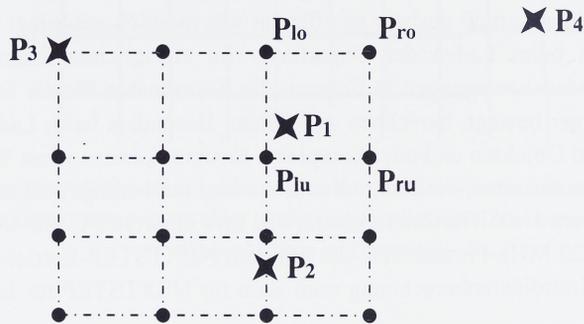


Abb. 2: Darstellung verschiedener Punkttypen, für die Koordinaten zu berechnen sind

Durch einfache lineare Interpolationsverfahren können die Geo-Koordinaten des gesuchten Punktes aus den Pixel- und Geo-Koordinaten der Referenzpunkte für die Unterscheidungsfälle 1 und 2 ermittelt werden, und bei Spezialfall 3 genügt eine einfache Zuweisung der Geo-Koordinaten des Referenzpunktes. Bei Punkt 4 spielt noch der Abstand des Punktes zu den nächsten Referenzpunkten eine Rolle. Es ist dann problematisch, geeignete Referenzpunkte zu bestimmen, wenn die Position des zu bestimmenden Punktes ungünstig liegt (Abb. 2).

Folgende Randbedingungen wurden festgelegt, die die Programmierung der Berechnungsroutinen erleichtern:

- a) Es werden nur Positionen von Punkten berechnet, die innerhalb des Netzes oder auf der Linie zwischen zwei Referenzpunkten liegen (1., 2. und 3.). Außerhalb liegende Bereiche werden ignoriert. Das Netz der Referenzpunkte muß dicht an die Kartenränder heranreichen.
- b) Die für die Berechnungen eingesetzten Eichpunkte umschließen einen rechteckigen Bereich.

Die Formeln für die Berechnung der Geo-Koordinaten eines gesuchten Punktes für Fall 1 bei Berücksichtigung von jeweils zwei der Umgebungsreferenzpunkte sind:

$$RW_{d1} = RW_{1u} + (RW_{ro} - RW_{1u}) \frac{x - x_{1u}}{x_{ro} - x_{1u}} \quad HW_{d1} = HW_{1u} + (HW_{ro} - HW_{1u}) \frac{y - y_{1u}}{y_{ro} - y_{1u}} \quad (1)$$

$$RW_{d2} = RW_{1o} + (RW_{ru} - RW_{1o}) \frac{x - x_{1o}}{x_{ru} - x_{1o}} \quad HW_{d2} = HW_{ru} + (HW_{ru} - HW_{1o}) \frac{y - y_{ru}}{y_{1o} - y_{ru}} \quad (2)$$

$$RW_u = RW_{1u} + (RW_{ru} - RW_{1u}) \frac{x - x_{1u}}{x_{ru} - x_{1u}} \quad HW_u = HW_{1u} + (HW_{ru} - HW_{1u}) \frac{y - y_{1u}}{y_{ru} - y_{1u}} \quad (3)$$

$$RW_o = RW_{1o} + (RW_{ro} - RW_{1o}) \frac{x_{1o} - x}{x_{ro} - x_{1o}} \quad HW_o = HW_{1o} + (HW_{ro} - HW_{1o}) \frac{y - y_{1o}}{y_{ro} - y_{1o}} \quad (4)$$

$$RW_l = RW_{1u} + (RW_{1o} - RW_{1u}) \frac{x - x_{1u}}{x_{1o} - x_{1u}} \quad HW_l = HW_{1u} + (HW_{1o} - HW_{1u}) \frac{y - y_{1u}}{y_{1o} - y_{1u}} \quad (5)$$

$$RW_r = RW_{ru} + (RW_{ro} - RW_{ru}) \frac{x - x_{ru}}{x_{ro} - x_{ru}} \quad HW_r = HW_{ru} + (HW_{ro} - HW_{ru}) \frac{y - y_{ru}}{y_{ro} - y_{ru}} \quad (6)$$

Erklärung: RW = Rechtswert, HW = Hochwert sowie x = horizontale Pixelkoordinate und y = vertikale Pixelkoordinate. Die Indizes werden durch die in den Formeln verwendeten Umgebungspunkte festgelegt (d1 = Diagonalpunkte links unten und rechts oben, d2 = Diagonalpunkte rechts unten und links oben, u = rechter und linker unterer Eichpunkt, o = rechter und linker oberer Eichpunkt, l = linker unterer und oberer Referenzpunkt, r = rechter unterer und oberer Umgebungspunkt).

Für Fall 2 entspricht Formelpaar 3 dem Formelpaar 4 und 5 der sechs Fälle, je nachdem ob der Punkt auf einer vertikalen oder horizontalen Linie zwischen zwei Referenzpunkten liegt. Formelpaar 1 und 2 sind nicht mehr gültig, es sei denn, man sieht Fall 2 als Spezialfall von 1 an und wählt zwei weitere Referenzpunkte und spannt damit wieder ein Rechteck aus Eichpunkten auf. Bei der Programmierung wurde letzteres angewendet.

Bei Einhaltung oben genannter Randbedingungen können je sechs Hoch- und Rechtswerte für den gesuchten Punkt berechnet werden, die durch Bildung von Mittelwerten auf je einen Hoch- und Rechtswert reduziert werden.

Völlig analog sind die Berechnungen von Pixel- aus Geo-Koordinaten (intern operiert VISOR mit dem Pixel-Koordinatensystem). Die Interpolationsalgorithmen sind dieselben. Deshalb wird auf eine Darstellung der Formeln verzichtet (Rechtswerte sind durch x-Koordinaten zu ersetzen und umgekehrt, gleiches gilt für Hochwerte und y-Koordinaten).

## Implementierung

Die allgemeinen Anforderungen an die Programmierung der externen Eichroutinen wurden weiter oben beschrieben. Die Referenzpunkte werden in einer ASCII-Datei namens "eichdatei" abgespeichert, die sich im jeweiligen Maßstabsverzeichnis der Karte befindet. Enthält dieses Maßstabsverzeichnis eine Datei "gauge.ext" mit dem Namen eines Bundles, lädt VISOR dynamisch beim Aufrufen dieser Karte ein solches in NeXT-Terminologie Bundle genanntes Objekt (unter MS Windows würde man dieses als Plugin bezeichnen), das den Objektcode für die externe Eichroutine enthält. Theoretisch könnte man in Abhängigkeit von der aufgerufenen Karte zwischen verschiedenen externen Koordinaten-Berechnungsroutinen wechseln.

Dieses Bundle lädt in einer Initialisierungsprozedur über das Netzwerk die Datei mit den Eichpunkten. Beim Einsatz einer sehr hohen Zahl von Eichpunkten (getestet wurden Dateien mit bis zu 90 000 Eichpunkten), schwillt diese auf mehrere Megabyte-Größe an und erhöht einerseits die Netzwerkbelastung, andererseits die Ladezeit der Referenzpunktdatei auf unzumutbare Werte. Bei einer hohen Eichpunktzahl wird die Eichdatei deshalb komprimiert (die Dateigröße sinkt auf etwa ein Sechstel). Das Laden übers Netzwerk und anschließende Auspacken kostet selbst mit älterer, langsamer NeXT-Hardware deutlich weniger Zeit als die Übertragung von bis zu drei Megabyte (rund 90 000 Referenzpunkte) an unkomprimierten Eichpunktdateien. Die (ausgepackte) Eichdatei beinhaltet als erstes eine Beschreibungszeile. Jede weitere Zeile der Eichdatei enthält, durch Tabulator getrennt, die Pixelkoordinaten (horizontal und vertikal) sowie die Gauß-Krüger Koordinaten (Hoch- und Rechtswert). Letzter Wert ist eine Kartenummer, die derzeit jedoch nicht ausgewertet wird und dazu gedacht ist, einmal spezielle Berechnungsroutinen einzusetzen, wenn die Umgebungspunkte eines zu berechnenden Punkts auf unterschiedlichen Karten liegen. In solchen Fällen treten oft Unstetigkeiten an den Kartenübergängen auf. Die Angaben zu jedem Eichpunkt werden in einer C-Struktur gespeichert, die in ein Container-Objekt (Storage) für schnellen wahlfreien Zugriff eingefügt wird. Dieses Storage-Objekt (neben einem weiteren Storage-Objekt als Cachespeicher) ist die einzige eingesetzte Klasse aus dem Application-Kit von NEXTSTEP. Ansonsten wird nur ANSI C verwendet. In der Workspace-Console werden danach die Beschreibungszeile der Eichdatei sowie die Eckkoordinaten des Eichdateigitters in Pixel- und Geo-Koordinaten und die Eichpunktzahl angezeigt.

Von der Firma Megatel wurde eine Reihe von Methoden vorgegeben, die im Bundle zur Koordinatenkonvertierung zu implementieren sind. Innerhalb dieser Methoden (dies sind die Prozeduren und Funktionen der objektorientierten Programmierung) sind sämtliche eigenen Programmerroutinen unterzubringen. Es handelt sich um die Methoden:

1. setMap:(const char \*)aMapFile andScale:(long)aScale

Übergabe des Maßstabsverzeichnisses, in dem sich die Karten befinden, für die eine externe Koordinatenberechnung erfolgen soll.

2. convert:(FLIGPoint \*)fligpoint toPixel:(NXPoint \*)pixel  
Berechnung von Pixel- aus Geo-Koordinaten.

3. convert:(NXPoint \*)pixel toGeo:(FLIGPoint \*)fligpoint  
Berechnung von Geo- aus Pixel-Koordinaten

#### 4. (double)distance:(NXPoint \*)p1 :(NXPoint \*)p2

Berechnung des Abstands zwischen zwei Punkten (Distanzmessung).

Methode 1 wird nur beim Aufruf der Karte ausgeführt. Die anderen drei Methoden sendet VISOR an das Eich-Bundle, wenn entsprechende Operationen notwendig werden. Z. B. sind beim Öffnen eines Vektorlayers Umrechnungen von Geo-Koordinaten in Bildschirmkoordinaten notwendig und umgekehrt beim Bewegen des Mauszeigers über die Karte Konvertierungen von Pixel-Koordinaten in Geo-Koordinaten.

Da die C-Struktur eines Referenzpunktes sowohl die Pixel- als auch die Geo-Koordinaten enthält und die Algorithmen zur Koordinatenberechnung mittels Referenzpunkten durch Interpolation bei der Bestimmung von Geo- aus Pixel-Koordinaten und im anderen Fall von Pixel- aus Geo-Koordinaten gleich sind, kann in beiden Fällen mit denselben Programmroutinen gearbeitet werden (Invarianz gegenüber der "Richtung" der Koordinatenumrechnung). Die C-Funktionen übergeben einen Schalter, mit dessen Hilfe die Quell- und Zielkoordinaten in den Berechnungsformeln korrekt belegt werden können. So kann der Programmcode kompakt gehalten, und fehlerträchtige Dopplungen (bei Sourcecode-Änderungen) können vermieden werden.

Die Eichdaten werden in einer eindimensionalen Containerklasse (im folgenden kurz Eichpunktliste genannt) gespeichert, enthalten aber zweidimensionale Koordinatenpaare (jeweils für Pixel- und Geo-Koordinaten). Eine Sortierung dieser Referenzpunkte ist nur nach einer Koordinate möglich. Dies erschwert die Suche nach möglichen Referenzpunkten in der Liste erheblich. Der Suchaufwand entspricht in etwa der Anzahl der in der Liste enthaltenen Eichpunkte. Bei einer sehr hohen Anzahl aufeinanderfolgender Koordinatenbestimmungen (z. B. Laden eines Vektorlayers mit vielen und/oder komplex geformten Objekten) kommt es bei zehntausenden Eichpunkten schnell zu Dutzenden von Millionen Listendurchläufen und damit nicht mehr akzeptablen Berechnungszeiten. Betroffen sind hauptsächlich die alten 25 MHz-Maschinen von NeXT, aber selbst auf schnellen Pentium-Systemen entstehen eventuell lange Wartezeiten. Weiterhin können die Berechnungsroutinen nicht von einem homogen verteilten Bestand an Referenzpunkten ausgehen, da dieser allmählich ausgebaut werden soll, was bei Bestimmung von zehntausenden von Referenzpunkten lange dauern kann, wenn andere wichtigere Arbeitssaufgaben anliegen. Aus diesem Grund sind intelligente Algorithmen in die Programm-Routinen eingebaut, die den Berechnungsaufwand optimieren. Es handelt sich um:

1. ein zweistufiges Caching-System aus einem Feld mit den bei der vorhergehenden Koordinatenberechnung eingesetzten Umgebungspunkten sowie einer Cacheliste, die einen Teil der Referenzpunkte der Eichpunktliste enthält,
2. einen durch den Benutzer festlegbaren Radius, der die Anzahl der gecachten Referenzpunkte und damit auch die Effektivität des Cachings bestimmt (voreingestellt auf die optimalen Werte für homogene und vollständige Verteilung der Eichpunkte alle 1000 Meter für die TK 25 ).

Das Caching besteht in der ersten Stufe aus vier C-Strukturen (im weiteren Umgebungspunkt-Feld genannt), die die Daten der für die letzte Koordinatenberechnung eingesetzten Umgebungspunkte enthalten und die bei einer anstehenden Berechnung zuerst auf die weitere Gültigkeit der darin enthaltenen Referenzpunktdaten abgefragt werden.

Schließlich können die Koordinaten für einen zu berechnenden Punkt noch innerhalb oder auf einer Linie des von den bei der letzten Berechnung eingesetzten Umgebungspunkten gebildeten Vierecks liegen.

In der zweiten Stufe wird ein Storage-Objekt (Cacheliste) abgefragt, wenn die für die letzte Berechnung verwendeten Referenzpunkte im Umgebungspunktfeld nicht mehr gültig sind. Die Eichprozedur scannt die Cacheliste auf Umgebungspunkte für die anstehende Berechnung, und falls sich Umgebungspunkte im Cache befinden, werden sie in das Umgebungspunktfeld geschrieben und für die Rechnungen genutzt.

Der letzte und ungünstigste Fall tritt ein, wenn das Caching versagt, u. a. wenn die Koordinaten des zu berechnenden Punkts weit entfernt von den zuletzt bestimmten Koordinaten liegen. Dann werden die als Umgebungspunkte benötigten Eichpunkte direkt aus der Eichpunktliste herausgesucht. Gleichzeitig wird die Cacheliste mit Referenzpunkten, die innerhalb eines definierten Abstands (dem Vierfachen des gerade eingestellten Radius) vom gesuchten Punkt liegen, aufgefüllt.

Es gibt zwei generelle Varianten der Suche nach den für die Koordinatenberechnung eines Punktes notwendigen (möglichst) unmittelbaren Umgebungspunkten in einer Referenzpunktliste - erstens den Vergleich jedes einzelnen Referenzpunktes auf geringsten Abstand zum gesuchten Punkt mit Durchlaufen der gesamten Liste an Eichpunkten (sehr aufwendig bei hoher Eichpunktzahl). Es gibt ein temporäres Umgebungspunktfeld mit Einträgen für die Kategorien linker oberer bzw. unterer, rechter oberer bzw. unterer Umgebungspunkt. Die ersten in diese Kategorien fallenden Eichpunkte der Liste werden dort hineinkopiert, und jeder folgende Eichpunkt der Liste wird mit diesen auf geringeren Abstand zum gesuchten Punkt verglichen und ersetzt gegebenenfalls einen der temporären Umgebungspunkte. Nach Durchlaufen der gesamten Liste stehen im genannten Feld die unmittelbaren Umgebungspunkte, falls welche vorhanden sind. Die zweite Variante benutzt einen definierten Abstandswert (in Richtung beider Koordinatenachsen) vom gesuchten Punkt, der so gewählt ist, daß die unmittelbaren Umgebungspunkte innerhalb dieses Abstands liegen und ebenfalls ein temporäres Umgebungspunktfeld, in das für die jeweilige Kategorie die ersten gefundenen Eichpunkte mit Abständen kleiner des eingestellten Radius gespeichert werden.

Variante 1 findet stets die Umgebungspunkte, Variante 2 hingegen setzt eine homogene Verteilung von Referenzpunkten voraus, um die Umgebungspunkte zu finden. Sind diese ermittelt, braucht der Rest der Referenzpunktliste nicht weiter durchsucht zu werden, und es wird Rechenzeit eingespart. Im ungünstigsten Fall (zu kleiner Abstandswert, im folgenden Radius genannt) werden innerhalb des Radius keine oder nicht ausreichend viele Umgebungspunkte gefunden.

Die verwendeten C-Funktionen basieren auf Variante 1, um stets die unmittelbaren Nachbarschaftseichpunkte zu finden, falls diese existieren. Oberstes Ziel ist die möglichst genaue Berechnung der gesuchten Koordinaten. Trotzdem wird ein (durch den Anwender änderbarer) Radiuswert vorgegeben. Dieser Wert bestimmt maßgeblich die Anzahl der in der Cacheliste abspeicherbaren Eichpunkte. Bei zu großem Umgebungsradius werden sehr viele Referenzpunkte in der Cacheliste gespeichert, und der Rechenaufwand steigt durch hohe Zahl an Schleifendurchläufen in der Cacheliste (Cachestufe 2), weshalb zu große Radiuswerte zu vermeiden sind. Bei zu kleinem Radius werden nur wenige Eichpunkte in der Cacheliste gespeichert, und es muß sehr oft die gesamte Eichpunktliste durchlaufen werden (da das Caching versagt), wobei der Rechenaufwand

ebenfalls enorm sein kann (abhängig von der Anzahl der insgesamt eingesetzten Referenzpunktzahl). Es ist deshalb darauf zu achten, geeignete Radiuswerte einzusetzen. Der Defaultwert ist auf optimales Caching für die TK 25 festgelegt.

Werden nach Durchlaufen der Liste keine Punkte innerhalb der Radiusdistanz gefunden, enthält das temporäre Eichpunktfeld die unmittelbaren Umgebungspunkte zum gesuchten Punkt (falls dieser sich nicht außerhalb des von den Eichpunkten aufgespannten Gitternetzes befindet), und die Berechnungen erfolgen mit den Koordinaten dieser Umgebungspunkte.

Die Berechnung gesuchter Koordinaten erfolgt nach den Formelpaaren 1 bis 6, allerdings unter Beachtung einiger Spezialfälle. In die Mittelwertberechnung der Rechtswerte (Gauß-Krüger) bzw. x-Werte (Pixelkoordinaten) gehen die Formeln 5 und 6 nicht ein, da die Divisoren der in den Formeln enthaltenen Brüche Null ergeben können und es infolgedessen zu Programmabbrüchen kommen kann. Entsprechend werden bei der Hochwert- bzw. y-Wert Berechnung die jeweiligen Formeln der Formelpaare 3 und 4 aus den Mittelwertbildungen nicht berücksichtigt. In die Berechnung der Koordinatenpaare gehen somit bis zu vier Werte in die Mittelwertbestimmung ein. Weitere Fallunterscheidungen treten auf, wenn weniger als vier Umgebungspunkte zur Verfügung stehen. Dies sollte zwar bei konsequenter Anwendung der Randbedingungen (Eichpunkte umspannen einen rechteckigen Bereich) nicht auftreten, muß aber bei der Programmierung berücksichtigt werden, um Programmabstürze zu vermeiden. Dann fallen weitere Anteile der Formelpaare für die Mittelwertberechnungen der x- bzw. y-Komponenten der Gauß-Krüger- bzw. Pixelkoordinaten weg.

### Genauigkeit der externen Eichung

Die Genauigkeit der mit einer externen Eichroutine berechneten Koordinaten wird von einer Reihe von Faktoren beeinflusst.

Durch Mittelwertbildung aus den gewonnenen Hoch- und Rechtswerten können die Geo-Koordinaten eines von VISOR angeforderten Punktes recht präzise bestimmt werden. Das gilt auch umgekehrt bei der Pixelkoordinaten-Berechnung aus Gauß-Krüger-Koordinaten. Da Pixelkoordinaten der Referenzpunkte durch Ablesen der Pixel-Koordinaten der Mauszeigerspitze (die sich genau über dem Kreuzpunkt einer horizontalen und einer vertikalen Koordinatenlinie befinden muß) in einem Koordinatenfenster bestimmt werden, die Karten mit 300 dpi gescannt wurden und ein Monitor eine Auflösung von 92 dpi bietet, ergibt sich eine auf etwa das Dreifache vergrößerte Darstellung der Karte auf dem Monitor. Bei einer geschätzten Ablesegenauigkeit (Cursor-Fadenkreuz) von einem Millimeter liegt der Fehler im allgemeinen unter 10 Meter bei einem Kartenmaßstab von 1:25000. Für diesen Maßstab wurde die externe Eichroutine entwickelt.

Die Genauigkeit der Eichung hängt von der Sorgfalt ab, mit der die Pixelkoordinaten der Eichpunkte bestimmt werden. Ausreißer mitteln sich meist bei der Berechnung von gesuchten Koordinaten durch die Mittelwertbildung (Formelpaare 1 bis 6) heraus.

Von Einfluß ist die Referenzpunktdichte auf der Karte. Bei geringer Punktdichte führen durch das Auge nicht wahrnehmbare nichtlineare Verzerrungen zwischen durch

Eichpunkte begrenzten Bereichen auf der Karte infolge des Scannen der Kartenblätter sowie beim Scannen übernommene Papieralterung zu Abweichungen. Wird bei der TK 25 entsprechend dem eingezeichneten Gauß-Krüger Gitternetz alle 1 000 Meter, entspricht 4 cm auf der Karte, ein Referenzpunkt gesetzt (für ganz Sachsen-Anhalt werden 30 000-50 000 Eichpunkte auf der TK 25 benötigt), liegt der resultierende Fehler bei maximal 10 bis 20 Metern.

Unstetigkeitsstellen durch Verbinden der Kartenblätter im Computer betragen bei visueller Betrachtung auf dem Monitor höchstens 2 bis 3 Millimeter (Maximalwerte), was bei etwa dreifacher Vergrößerung der Kartenansicht auf dem Bildschirm ungefähr 20 bis 30 Meter Unsicherheit in der Koordinatenbestimmung ausmacht. Inwiefern dieser Fehler in die Berechnungen eingeht, ist schwer zu beurteilen. Zu ermitteln ist die Anzahl der vier Umgebungspunkte auf jeder Seite der Unstetigkeitsstelle, die Neigung der Übergangskante zum von den Referenzpunkten gebildeten Viereck sowie auch die Neigung der aneinandergelagerten Kartenteile zueinander. Maximal sind die genannten 30 Meter für eine Gauß-Krüger-Koordinatenachse möglich (betrachtet wird die den Programmbeutzer interessierende Berechnung der Gauß-Krüger-Koordinaten, entsprechendes gilt für die in VISOR benötigte Bildschirmkoordinaten-Berechnung aus Gauß-Krüger-Koordinaten), im allgemeinen verringert sich durch die externe Eichung sogar dieser Fehlerwert, da Eichpunkte von beiden Seiten der Unstetigkeitsstelle in die Berechnungen eingehen.

Die resultierende maximale Unsicherheit einer berechneten Gauß-Krüger-Koordinate durch die externe Eichroutine mit Referenzpunkten alle 1000 Meter für die TK 25 (Maßstab 1:25000) liegt dann bei

$$dx = dx_1 + dx_2 = 20-40 \text{ Meter (7)}$$

für den Kartenbereich innerhalb eines Kartenblatts und

$$dx = dx_1 + dx_2 + dx_3 = 40-70 \text{ Meter (8)}$$

für Koordinatenberechnungen, bei denen Referenzpunkte von beiden Seiten eines Kartenübergangs beteiligt sind mit  $dx$  = Berechnungsungenauigkeit für eine Koordinatenachse,  $dx_1$  = durch Ablesen der Pixelkoordinaten von Eichpunkten am Bildschirm bedingte Unsicherheit,  $dx_2$  = Maximalfehler durch nichtlineare Verzerrungen auf der Karte und  $dx_3$  = durch Kartenübergänge verursachter Fehler.

Zur Nachprüfung wurden visuelle Kontrollen am Monitor ausgeführt. Gitterkreuzungen des eingezeichneten Koordinatennetzes, die (noch) nicht als Referenzpunkte eingesetzt werden und für die Gauß-Krüger-Koordinaten bekannt sind, dienten als Kontrollpunkte. Die maximale Abweichung für eine Koordinate betrug bei mehreren Testpunkten unter 20 m, meist unter 10 m. Selbst in der Nähe von Kartenblattübergangsstellen wurden 20 m Abweichung nicht überschritten. Damit hat die externe Eichroutine (externe Koordinatenberechnung durch eigene Algorithmen) für die in VISOR integrierten Anteile der TK 25 am Beispiel des Saalkreises ihre Praxistauglichkeit bewiesen.

## Anmerkungen

- <sup>1</sup> Limp 1996 – Johnson 1996 – Voopris 1996 – Smolnik/Göldner 1995 – Landesamt für Archäologie Dresden 1996 – Richter 1996
- <sup>2</sup> Ausführliche Beschreibungen von VISOR und Konzeptionen archäologischer Datenbankanwendungen mit der Möglichkeit der Projektion von Sachdaten auf einem Kartenhintergrund finden sich in Richter 1996 - megatel 1993 – megatel 1995
- <sup>3</sup> Weise 1995
- <sup>4</sup> Weise 1995
- <sup>5</sup> Weise 1995

## Literaturverzeichnis

- Landesamt für Archäologie mit Landesmuseum für Vorgeschichte Dresden (Hrsg.) 1996  
Dokumentations- und Informationssystem Archäologie - Dresden, 10.01.1996
- Johnson, I. 1996  
Desktop Mapping and Raster GIS: Best of both Worlds? - The Colloquia of the XIII International Congress of Prehistoric and Protohistoric Sciences, Forli (Italia) 8-14 September 1996, Theoretical and Methodological Problems, Colloquium II: The present state of G. I. S. Applications and analogous systems in Prehistoric Archaeology
- Limp, W. F. 1996  
Developing Methodologies in the Analysis of spatially referenced Data and their Impacts on archaeological Method and Theory - The Colloquia of the XIII International Congress of Prehistoric and Protohistoric Sciences, Forli (Italia) 8-14 September 1996, Theoretical and Methodological Problems, Colloquium II: The present state of G. I. S. Applications and analogous systems in Prehistoric Archaeology
- megatel GmbH 1993  
VISOR-Handbücher NeXT und Windows
- megatel GmbH 1994  
VISOR-Handbücher NeXT und Windows
- megatel GmbH 1995  
VISOR-Handbücher NeXT und Windows
- NeXT Computer Inc. 1993  
NEXTSTEP Object-Oriented Programming and the Objective C Language - NEXTSTEP Developers Library Release 3, New York/Bonn/Madrid
- Richter, T. 1996  
Zugriff archäologischer Anwendungen auf die Datenbank Arche und Erweiterung ihrer Funktionalität durch Anbindung an ein Geografisches Informationssystem - Jahresschrift für mitteldeutsche Vorgeschichte 78, Halle (Saale), S. 417-442
- Smolnik, R./Göldner, R. 1995  
Das Dokumentations- und Informationssystem Archäologie in Sachsen - Archäologie aktuell 3, Dresden, S. 265-271
- Voopris, A. 1996  
Archaeological Theory and GIS, any relations? - The Colloquia of the XIII International Congress of Prehistoric and Protohistoric Sciences, Forli (Italia) 8-14 September 1996, Theoretical and Methodological Problems, Colloquium II: The present state of G. I. S. Applications and analogous systems in Prehistoric Archaeology

Weise, R. 1995

Erstellen der digitalen Kartengrundlage für ein Geographisches Informationssystem (GIS) - Jahresschrift für mitteldeutsche Vorgeschichte 77, Halle (Saale), S. 339-347

**Anschrift**

Dipl.-Phys. Thomas Richter, Landesamt für Archäologie - Landesmuseum für Vorgeschichte - Sachsen-Anhalt, Richard-Wagner-Str. 9-10, D - 06114 Halle (Saale)

Manuskriptabgabe: 19.12.1996