

Lehrkonzept für ein Augmented Reality Praktikum

Martin Bauer, Martin Wagner,
Marcus Tönnis, Gudrun Klinker
Technische Universität München
Fakultät für Informatik
Boltzmannstraße 3
85748 Garching bei München, Germany
Tel.: +49 (0) 89 289 18215
Fax: +49 (0) 89 289 18207

Verena Broy
BMW Forschung und Technik GmbH
Hanauer Straße 46
80992 München
Tel.: +49 (0) 89 382 48916
Fax: +49 (0) 89 382 44988

ar-group@ar.in.tum.de

Abstract: Wir präsentieren ein Lehrkonzept für ein einsemestriges Studierendenpraktikum „Augmented Reality“ (AR), in dem wichtige Grundkonzepte der AR behandelt werden und durch Projektarbeit die Motivation der Studierenden gefördert wird. Das Lehrkonzept wurde im Sommersemester 2004 an der TU München erprobt und basiert auf Erfahrungen aus seit dem Wintersemester 2000 durchgeführten Praktika.

Die primären Lehrziele liegen in der Vermittlung der Fähigkeit zur Einschätzung der Entwicklungskomplexität von AR Anwendungen, des systematischen Umgangs mit Mathematik und des notwendigen Maßes an räumlichem Vorstellungsvermögen. Das Praktikum gliedert sich in zwei Phasen, auf die Vermittlung der Grundlagen durch wöchentliche Übungsblätter folgt eine Projektphase, in der Teilnehmergruppen eigenverantwortlich AR Anwendungen implementieren. Beide Phasen wurden mit 14 Studierenden durchlaufen.

Wir stellen die hervorragenden Ergebnisse der Projektphase vor und diskutieren die Implikationen bestimmter grundsätzlicher Bestandteile unseres Lehrkonzepts.

Schlüsselwörter: Augmented Reality, Lehrkonzept

1 Motivation

In der Studienordnung der Fakultät für Informatik der Technischen Universität München ist wie an den meisten Universitäten ein sogenanntes Praktikum in der praktischen Informatik im Hauptstudium verpflichtend vorgeschrieben. Die Veranstaltung hat einen Umfang von sechs Semesterwochenstunden und wird üblicherweise entweder auf Basis von wöchentlichen Übungsblättern mit zu erledigenden Programmier- und Theorieaufgaben oder in Projektform durchgeführt. Neuerdings sind neben Diplomstudenten auch Bachelorstudenten im letzten Studienjahr Teilnehmer solcher Praktika.

Wir organisieren seit dem Wintersemester 2000/2001 ein Praktikum *Augmented Reality* (AR). Das Praktikum nimmt dabei einen zentralen Platz in der Ausbildung von Studierenden für das Gebiet der AR ein. Aufbauend auf theoretische Vorkenntnisse aus der Vorlesung *Einführung in die Erweiterte Realität* sollen die Studierenden hier erstmals reale AR-Systeme selbst implementieren, um so den Grundstock für erfolgreiche praxisnahe Studien-, Bachelor- oder Diplomarbeiten legen zu können.

1.1 Ergebnisse früherer Praktika

In früheren Praktika verfolgten wir einen Ansatz, der primär auf projektbasiertes Software-Engineering in der Domäne der AR ausgerichtet war. In den Projekten STARS [6], FataMorgana [7] und FixIt [8] wurde viel Wert auf die aus dem Software-Engineering bekannten Phasen der Anforderungsanalyse und des System- und Objektdesigns [3] gelegt. Gleichberechtigt daneben stand die Ausbildung in den Basistechniken der AR.

In der Betreuung dieser Praktika und von ca. 50 Studienarbeiten in den letzten Jahren haben wir folgende häufig wiederkehrende Probleme bei der Implementierung von AR-Anwendungen identifiziert:

Räumliches Vorstellungsvermögen AR-Anwendungen sind inhärent dreidimensional. Im Gegensatz zu normalen Computergraphikanwendungen ist bei so gut wie allen Anwendungen eine Transformation zwischen mehreren Koordinatensystemen nötig, z.B. beim optischen Tracking vom Kamera- ins Raumkoordinatensystem. Obwohl die dahinterstehende Mathematik Stoff des ersten Semesters und im Prinzip sehr einfach verständlich ist, offenbart sich eine große Lücke zur sicheren und zielführenden Anwendung.

Systematischer Umgang mit Mathematik. Für nur wenige Gebiete der praktischen Informatik ist eine systematische Herangehensweise an die zugrundeliegenden mathematischen Methoden so notwendig wie für die erweiterte Realität. Dies liegt insbesondere an der Schwierigkeit des Debugging von Koordinatentransformationen – hat man etwas falsch berechnet, zeigt sich dies meist nur in einem schwarzen Bildschirm. Der Unterschied zwischen einer „fast richtigen“ und einer „ganz falschen“ Implementierung ist aus der Funktion des fertigen Programms meist nicht ersichtlich.

Selbsteinschätzung des Aufwands. Da es sich bei den Kernalgorithmen von AR-Anwendungen oft nur um wenige Zeilen Code handelt, neigen insbesondere Anfänger oft dazu, die Erstellung derselben nach hinten zu schieben, ohne zu berücksichtigen, daß von diesen wenigen Zeilen die Funktionalität der gesamten Anwendung abhängt. Rücken dann Deadlines näher, bleibt häufig keine Zeit, um die fehlenden Koordinatentransformationen und Renderingroutinen nachvollziehbar zu implementieren, eine Wiederverwendung von Studienarbeitsergebnissen wird hierdurch massiv erschwert.

1.2 Das durchgeführte Praktikum

Der Schwerpunkt des aufgrund des neuen Konzeptes durchgeführten Praktikums ist mehr in Richtung der Basistechniken der AR verschoben. Um auf einer nutzbaren Basis aufbauen zu können, wurde das AR Toolkit [5] genutzt. Die Entscheidung für diese Softwarebibliothek hatte unter anderem folgende Gründe: Das AR Toolkit läuft auf allen gängigen Plattformen (Windows, Mac OS X, Unix) und kann damit auch von Studierenden am Heimrechner eingesetzt werden. Man benötigt keinerlei spezielle Hardware, eine einfache Webcam ist ausreichend. Durch die einfache API können sich die Studierenden auf die wesentlichen algorithmischen Probleme von AR-Anwendungen beschränken und müssen keine neue Implementierungsumgebung erlernen. Durch die große Verbreitung des Toolkits finden sich im Internet genügend Hilfen zu häufigen Problemen. Die Studierenden sind in der Lage selbst optische Marker herzustellen und zu nutzen.

2 Lehrkonzept

Unser Lehrkonzept beinhaltet eine Aufteilung der Lehrinhalte in theoretische Grundlagen, praktische Übungen und Projektarbeit. Ein wichtiger Aspekt bei der Gestaltung der Übungsblätter lag auf der Motivation der Studierenden durch frühzeitige Ergebnisse in Form von echten Anwendungen.

Zu diesem Zweck sind in den ersten zwei Monaten des Praktikums auf den Übungsblättern mathematische Aufgaben vermischt mit Programmieraufgaben die im Allgemeinen eigenständige kleine Augmented Reality Anwendungen sind. Die Anwendungen bauen aufeinander und auf die mathematischen Grundlagenaufgaben auf, so daß die Studierenden immer wieder bei der Lösung neuer Programmieraufgaben auf Ergebnisse früherer theoretischer Aufgaben zurückgreifen können.

Der letzte Monat des Praktikums ist als Projektphase organisiert, in der die Studierenden selbst definierte oder auf Vorschlägen der Übungsleitung basierende Projekte in Kleingruppen von drei bis vier Leuten bearbeiten und diese zum Ende des Praktikums den anderen Gruppen präsentieren.

2.1 Theoretische Grundlagen und Praktische Übungen

Im Rahmen unserer Lehrtätigkeit über die letzten vier Jahre haben wir bei den Studierenden immer wieder Defizite im Umgang mit dreidimensionaler Geometrie, linearer Algebra und Optimierungsverfahren festgestellt. Deshalb verlagerten wir den anfänglichen Schwerpunkt der Theorieaufgaben auf den Übungsblättern mit der Zeit immer mehr zu den Programmieraufgaben.

In den praktischen Übungen sollen neben dem Einüben der theoretischen Konzepte auch die Grundlagen zur Implementierung ansprechender Augmented Reality Anwendungen geschaffen werden. Daher nehmen auch Themen aus der Computergrafik und Computeranimation einen hohen Stellenwert ein. Die realisierten Lösungen sollten optisch ansprechend sein.

Im folgenden sind die wesentlichen Themenbereiche kurz dargestellt und auf die Motivation und Umsetzung eingegangen. Die genauen Übungsaufgaben sind aus den Übungsblättern zum Praktikum [13] ersichtlich.

Rotationsmatrizen Als wichtiges Konzept der Computergrafik werden auf dem ersten Übungsblatt die Grundlagen von Transformationen in \mathbb{R}^2 und \mathbb{R}^3 behandelt. Ziel dieser Übungseinheit ist es, den Studierenden ein Gefühl für den intuitiven Umgang mit Rotationsmatrizen zu geben. Hierzu wird die Anwendung eines Rotationsoperators auf einen Vektor direkt am Beispiel durchgerechnet und die Auswirkungen bei der Vertauschung der Reihenfolge von Transformationen untersucht. Weiterhin wird das Konzept der homogenen Koordinaten zur Darstellung von Transformationen im Raum entwickelt. Es bestehen Bestrebungen, diese Inhalte auch im Rahmen einer Vorlesung im Grundstudium zu verankern; dennoch würde es Sinn machen diese Aufgaben als Wiederholung durchführen zu lassen.

Quaternionen Aufgrund numerischer Eigenschaften sind für viele Anwendungen in Augmented und Virtual Reality Quaternionen zur Darstellung von Transformationen im Raum wesentlich besser geeignet als die (intuitiv verständlicheren) Rotationsmatrizen.

In dieser Übungseinheit werden die Konzepte des Rechnens mit Quaternionen dargestellt und insbesondere auf den Zusammenhang von Quaternionen und den entsprechenden Rotationen eingegangen. Speziell werden hier die Umwandlungsfunktionen von Quaternionen in Transformationsmatrizen und umgekehrt selbst implementiert und getestet.

Szenengraphen Das Konzept von Szenengraphen ist eines der wichtigsten Konzepte der Computergrafik. In dieser Übungseinheit werden die Ideen der hierarchischen Gliederung der Szene und das Traversieren des *OpenGL Matrix Stacks* eingeübt.

Auf die Verwendung einer fertigen (objektorientierten) Szenengraphbibliothek [11] wurde im Rahmen dieses Praktikums bewusst verzichtet. Zum einen gestaltet sich die Integration einer solchen Bibliothek in die Programmstruktur des *AR Toolkit* nicht völlig trivial und zum anderen sehen wir genau das Umgehen mit den nötigen Transformationen und Matrizen beim selbstständigen Traversieren der Szene als ein wichtiges Lernziel des Praktikums.

Computergrafik und Animation Im wesentlichen um die Beispielprogramme grafisch etwas aufzulockern wurden in mehrere Übungsblätter aus der Computergrafik bekannte Animationstechniken eingebaut: Ein einfaches *Particle System* zur Animation eines virtuell aus einer Teekanne fließenden Tees und Interpolation als nützliche Anwendung von Quaternionen [4]. Der Schwerpunkt liegt hierbei eher auf einer ansprechenden Präsentation und Grafik als auf der Implementierung neuer Konzepte.

Wenngleich diese Inhalte nicht direkt zum Verständnis von Augmented Reality beitragen, erfüllen sie in diesem Praktikum doch einen wichtigen Zweck. Neben der Motivation für die Studierenden, auch optisch ansprechende Programme zu bauen, macht dies auch klar, daß zu einem überzeugenden Augmented Reality System mehr gehört als die reine Theorie.

Benutzerschnittstellen für Augmented Reality Um die Interaktion mit einem Augmented Reality System zu ermöglichen kann man selbst wieder Möglichkeiten der Augmented Reality nutzen. Beispielsweise können Parameter eines Systems durch die relative Orientierung und Positionierung

verschiedener Marker eingestellt werden; auch können bestimmte Aktionen durch die Sichtbarkeit eines speziellen Markers ausgelöst werden.

In dieser Übungseinheit sollten unterschiedliche Dreh- und Schieberegler entwickelt werden. Deren Interaktionsformen wurden zum Teil in den Abschlußprojekten noch weiter verfeinert.

Multi-Marker Tracking Markerbasierte optische Trackingsysteme können nur dann sinnvoll für größere Anwendungen eingesetzt werden, wenn zum Tracken von einzelnen Objekten mehrere Marker eingesetzt werden. In dieser Aufgabe wird die relative Kalibrierung eines solchen Objektes mit mehreren Markern implementiert. Der Einfachheit halber wird die Kombination der verschiedenen Marker auf Basis der 3D-Rekonstruktion durchgeführt, auch wenn dies nicht dem aktuellen Stand der Forschung entspricht [1].

Registrierung , Kalibrierung und Optimierungsverfahren Stellvertretend für verschiedene Optimierungsalgorithmen wird auf dem letzten Aufgabenblatt ein Zeigergerät mit mehreren Markern so kalibriert, daß die genaue Position der Spitze des Zeigers bekannt ist [10]. Hierzu werden zuerst die einzelnen Marker relativ zueinander positioniert und dann durch Drehen des Zeigers um die Spitze eine Menge von Messwerten gesammelt welche alle ein Gleichungssystem mit der Spitze als Fixpunkt erfüllen. Dieses Gleichungssystem wird mittels Singulärwertzerlegung approximiert. Dazu werden Konzepte der Optimierung mittels Singulärwertzerlegung eingeführt.

Bei dieser Anwendung wurden als Nebeneffekt die Grenzen des *ARToolkit* aufgrund der ungünstigen Fehlercharakteristik bei der Behandlung von 3D Daten ausserhalb des Kamerakoordinatensystems verdeutlicht.

2.2 Ergebnisse der Projektarbeiten

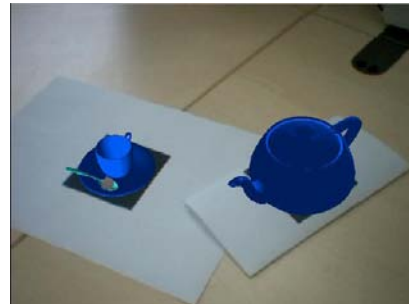
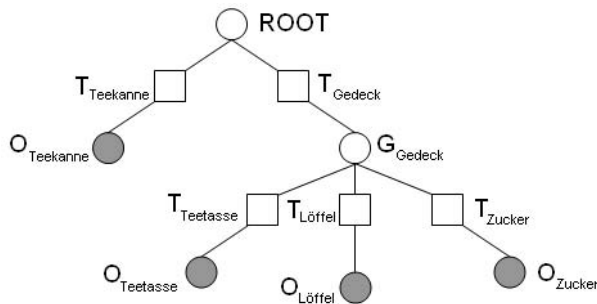
Insgesamt vier Gruppen mit zwei bis vier Teilnehmern implementierten im letzten Monat des Praktikums ein eigenständiges AR-Projekt. Hierbei wurde in nur einem Monat die Ideenfindung, das Systemdesign, die Implementierung, Dokumentation und Präsentation einer neuen AR-Anwendung bewerkstelligt. In der Wahl des Themas und der verwendeten Hilfsmittel waren die Gruppen frei, entsprechend groß ist das Spektrum der vorgestellten Projekte.

Artist – Virtuelles Graffiti Artist ist eine Anwendung mittels der der Benutzer mit einem Stift auf einem von einer Kamera beobachteten Bildschirm Zeichnungen anfertigen kann. Zusätzlich besteht die Möglichkeit, die Zeichnung auf einen entfernten Rechner zu übertragen und dort auch mittels AR ortsfest zu einem weiteren Marker im Kamerabild anzuzeigen. Das System ist auch für die Benutzung auf einer großen Leinwand mit Projektor geeignet.

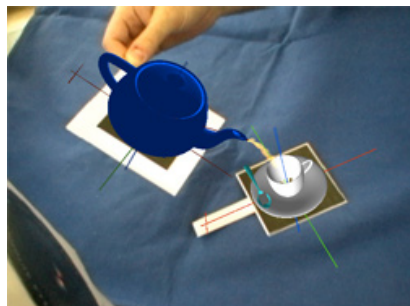
Artillery – Augmented Reality Computerspiel Artillery ist eine Augmented Reality Umsetzung eines alten Computerspieles: Zwei oder mehr Spieler verfügen über jeweils eine in zerklüftetem Gelände stehende Burg mit einer Kanone und versuchen, die Burgen der anderen Spieler zu zerstören. Reihum feuert jeder Spieler einen Schuß ab und kann dabei den Abschußwinkel und die Menge des Schwarzpulvers einstellen.



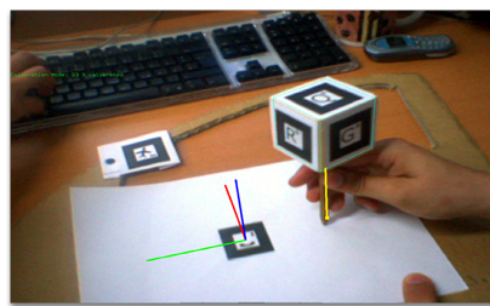
(a) Einfache AR Toolkit Anwendungen vom ersten Übungsblatt



(b) Szenengraph und Umsetzung in OpenGL



(c) Animationen in Augmented Reality: *Particle System* und Quaternioneninterpolation



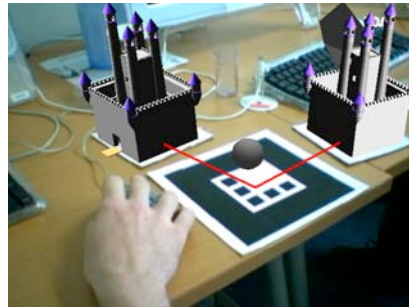
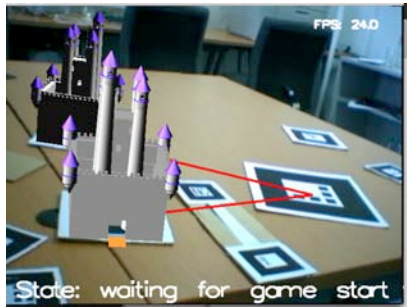
(d) Beispiele von Benutzerschnittstellen für Augmented Reality

(e) Interaktive Kalibrierung eines Zeigergerätes

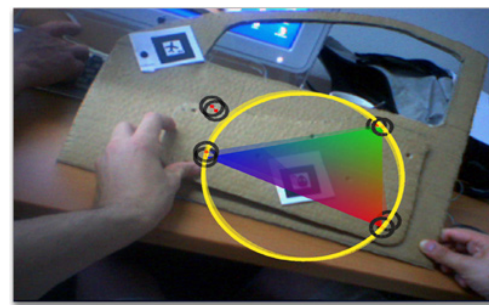
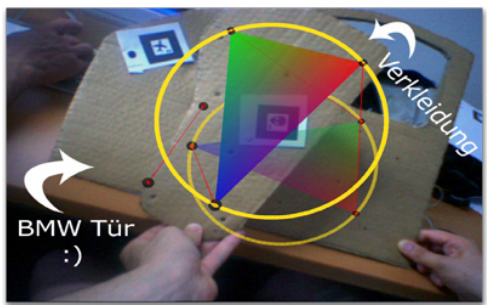
Abbildung 1: Praktische Übungsaufgaben der ersten Praktikumsphase



(a) ARtist – Virtuelles Grafitti



(b) ARtillery – Augmented Reality Computerspiel



(c) AR-unterstützte Montage einer PKW Türinnenverkleidung



(d) Jeep im Sheepland – Erweiterung der SHEEP-Demo

Abbildung 2: Screenshots der Abschlussprojekte

Mehrere Spieler können übers Netzwerk spielen, wobei ein Server den Spielfluß steuert. Jeder Spieler hat einen eigenen Rechner, eine eigene Kamera und diverse Marker, mit dem Position, Orientierung, Winkel und Schwarzpulvermenge eingestellt werden können. Das Spielfeld wird durch Marker definiert, die zu Spielbeginn vom Server registriert werden. Somit muß jeder Spieler nur seinen eigenen Marker sehen, um mitspielen zu können, wobei die Registrierungsgenauigkeit verbessert wird, wenn mehrere Marker sichtbar sind.

AR-unterstützte Montage einer PKW Türinnenverkleidung Ziel dieses Projektes war es, ein Augmented Reality System zur Unterstützung der Montage einer Türinnenverkleidung aufzubauen. Dabei sollten die bei der Montage verdeckten Montagepunkte an der Türstruktur und der Rückseite der Verkleidung (Clipse) augmentiert werden, um die Montage zu erleichtern. Zusätzlich können weitere Hinweise, z.B. Pfeile, dargestellt werden, die den Werker bei der Montage unterstützen.

In der Umsetzung wird die zu erfolgende Transformation im Raum dargestellt durch virtuelle Gummibänder zwischen den entsprechenden Montagepunkten an der Türe und an der Verkleidung. Zudem zeigt ein farbiges Dreieck die Orientierung im Raum an.

Jeep im Sheepland – Erweiterung der SHEEP-Demo In diesem Projekt sollte die SHEEP-Demo unserer Gruppe erweitert werden. SHEEP [9] zeigt die Möglichkeiten der Verteilung und Dynamik unserer Forschungsplattform DWARF [2] auf spielerische Art und Weise: Schafe, die als verteilte Prozesse im Netzwerk existieren, weiden auf einer auf einen Tisch projizierten virtuellen Wiese. Zudem kann diese Szene auch durch Laptopbildschirme und Head Mounted Displays im 3D-Raum visualisiert werden. Nun sollte die virtuelle Sicht durch die Windschutzscheibe eines getrackten realen Spielzeugautos implementiert werden.

Neben dieser neuen Sicht entstand zudem eine funktionsfähige Geschwindigkeitsanzeige über den Tacho des virtuellen Cockpits, und ein Navigationssystem, das stets den Weg zum nächsten Schaf weist.

3 Diskussion

Die soeben beschriebenen Projektarbeiten zeigen deutlich den Lernerfolg der Studierenden, die oftmals keine praktischen Vorkenntnisse in relevanten Bereichen der Informatik mitbrachten. Im folgenden diskutieren wir die Wirksamkeit der verschiedenen Bestandteile unseres Lehrkonzepts und der verwendeten Hilftechniken.

3.1 Erreichen der Lernziele

Wir beobachteten eine sehr starke Akzentuierung der mathematischen und programmatischen Grundlagen der AR im Gegensatz zu den prozessorientierten Inhalten früherer Praktika (vgl. Abschnitt 1). Die oben formulierten Lernziele des räumlichen Vorstellungsvermögens, des systematischen Umgangs mit der nötigen Mathematik und der Selbsteinschätzung des Aufwands für AR-Anwendungen

wurden voll und ganz erreicht – alle Projektgruppen erreichten ihre selbstgesteckten ambitionierten Ziele in der vorgesehenen Zeit von vier Wochen. Alle Projektarbeiten enthielten nichttriviale, teils neuartige Koordinatentransformationen im dreidimensionalen Raum. Die zugehörigen Gleichungen wurden systematisch ermittelt und auf Basis der in der ersten Praktikumsphase implementierten Bibliotheksfunktionen rasch in Code umgesetzt.

3.2 Verwendung von AR Toolkit

Mit dem AR Toolkit benutzten wir eine Basisbibliothek, die sich trotz gewisser bekannter Schwachstellen hervorragend bewährt hat. Die Entwicklungsplattform war das Lehrstuhlrechnerlabor mit Apple Mac OS X Rechnern. Die freie Verfügbarkeit und Plattformunabhängigkeit des AR Toolkits erwies sich als großer Vorteil. Viele Teilnehmer entwickelten auch auf ihren privaten Mac OS X-, Linux- oder Windowsrechnern, obwohl alle Resultate auf den Laborrechnern lauffähig sein mussten. Dies wurde von unserer Seite mit der leihweisen Ausgabe von Apple iSight Kameras unterstützt.

Die sehr einfache prozedurale API des Toolkits erleichterte den Studierenden den Einstieg in die Programmierung von AR-Anwendungen immens – schon nach dem ersten Übungsblatt waren selbstprogrammierte Applikationen erstellt. Zudem ließ sie ihnen große Freiheiten: manche programmierten im „klassischen“ prozeduralen Stil, andere entwickelten vollständige objektorientierte Kapselungen für das Toolkit – eine erfreuliche Entwicklung vor dem Hintergrund des in früheren Praktika explizit geübten objektorientierten Modellierens, die jetzt fast nebenbei mit einer realen Anwendung eingeübt wurde.

Natürlich ist die Nutzung eines kompletten Toolkits als Basis nicht völlig unproblematisch. Des öfteren traten Probleme auf, wenn eigene OpenGL-Befehle in die nicht trivial zu durchschauenden Rendering-Loop des Toolkits eingefügt werden sollten. Die durch die sehr einfache Konfigurierbarkeit des Toolkits bedingte suboptimale Erkennungsqualität und Beleuchtungsempfindlichkeit führte zwar zu zahlreichen Problemen, führte den Studierenden aber die Limitationen optischer Trackingsysteme deutlich vor Augen. Insbesondere die bei der Kalibrierung von AR-Systemen nötige Sorgfalt wurde hier verstärkt gelehrt.

3.3 Erfahrungen aus dem Praktikumsablauf

Teambildung Im ersten Teil des Praktikums mussten die Teilnehmern einzeln Programmier- und Rechenaufgaben lösen. Hierdurch stellten wir sicher, daß jeder Einzelne die grundlegenden Prinzipien der AR verstehen konnte. Insbesondere eher zurückhaltend auftretende Studierende gerieten in den früheren teambasierten Praktika manchmal in die Rolle der Zuarbeiter, die im eigentlich gelehrt Problemereich nichts oder nur wenig beitragen konnten. Die natürlich dennoch notwendige Teambildung förderten wir durch das Anbieten einer Programmierberatungsstunde, die dreimal wöchentlich im Praktikumslabor stattfand. Zusätzlich konnten die Praktikumssteilnehmer bereits in der ersten Phase über die bereitgestellte Mailingliste die Aufgaben und mögliche Lösungswege miteinander diskutierten. Durch diesen Teamgeist fanden sich schnell interessensbedingt die Gruppen für die zweite Phase. Dabei wurde von uns darauf geachtet, daß die verschiedenen Pro-

grammiererfahrungen ungefähr gleichmässig über die Teams verteilt waren. Da jeder Einzelne durch die Aufgabenblätter ein fundiertes Verständnis für die Entwicklung von kleineren AR Anwendungen ausgebildet hatte, wurde die Zusammenarbeit im Team weitgehend gleichberechtigt und kommunikativ durchgeführt.

Motivation In dem ersten Teil wurden die Studierenden motiviert, indem ihnen gezeigt wurde, mit welchem relativ geringem Aufwand die Arbeit mit dem AR Toolkit zu Erfolgen führt. Nicht zuletzt die selbstentwickelten Anwendungen, welche die Ergebnisse der Programmierung anfassbar und erlebbar machen, haben ihren Teil dazu beigetragen und die Bereitschaft, die oft ungeliebten mathematischen Grundlagen zu verinnerlichen, maßgeblich gefördert. Da die Studierenden selbst an der Projektkonzeption beteiligt waren, war die Motivation zur Umsetzung der Projekte sehr hoch.

Werkzeugbenutzung In der ersten Phase des Praktikums wurden etliche Entwicklungswerkzeuge vorgestellt und eingeübt, die in der Projektphase von allen Teams eingesetzt wurden.

Ein CVS Repository wurde nicht nur erfolgreich zur Unterstützung der Teamarbeit genutzt, sondern zusätzlich zur Versionsverwaltung und Fehlermanagement. Im Team wurde gemeinsam Code entwickelt und getestet.

Plattformunabhängige Makefiles der Build-Umgebung wurden von den Studierenden selbständig angelegt und erweitert. Damit konnten die Implementierungen ohne zusätzlichen Portierungsaufwand in verschiedenen Umgebungen gestartet werden.

Das Anlegen von Wiki-Seiten [12] unterstützte nicht nur die Dokumentation der Projekte, sondern diente auch der Validierung und Spezifikation der Projektaufgaben. Zusätzlich konnte diese Plattform als Kommunikationsbasis innerhalb des Teams und mit dem Projektbetreuer verwendet werden.

Damit sind die AR-Anwendungen der Projekte weiterhin als Demonstrationen unkompliziert nutzbar.

Selbsteinschätzung Da sich die Studierenden ihre Projekte selbst auswählen und ausformulieren konnten, bestand die Gefahr, daß das Ziel nicht erreicht werden konnte. Allerdings führten alle Projekte innerhalb von vier Wochen zu teilweise sehr beeindruckenden Demonstrationen. Man kann davon ausgehen, daß die für AR Systeme notwendigen Grundlagen vermittelt und verstanden wurden und daß die Studierenden den notwendigen Implementierungsaufwand gut abschätzen konnten.

3.4 Einarbeitung in die Forschungsplattform DWARF

Eines der Abschlussprojekte, *Jeep im Sheepland*, wurde im Kontext unseres Forschungs-AR-Frameworks DWARF [2] durchgeführt. Der Umgang mit diesem hochgradig verteilten, über verschiedene Hardware-Plattformen einsetzbaren und nicht immer optimal dokumentierten Framework erfordert einen hohen Lernaufwand. Bei den vorrausgegangenen Projekten, bei denen DWARF von Anfang an eingesetzt wurde, musste viel Zeit in die Arbeit mit dem Framework investiert werden, so daß

essentielle Funktionalitäten für AR Systeme sehr weit nach hinten verschoben wurden und die dafür notwendige Mathematik oft zu kurz kam.

Der im Praktikum verfolgte Ansatz, umgekehrt zuerst die mathematischen und algorithmischen Grundlagen und erst anschliessend die Forschungsplattform zu lehren, erwies sich der herkömmlichen Heransgehensweise als überlegen. Sowohl der Erklärungsaufwand für DWARF als auch die Gesamtarbeitsdauer der Projektentwickler waren deutlich geringer als bei vorangegangenen vergleichbaren Projekten.

3.5 Betreuungsaufwand

Im ersten Teil des Praktikums mussten wöchentlich Übungsblätter erstellt werden und die Hausaufgaben korrigiert werden. Die anfänglichen mathematischen Theorieaufgaben erforderten einen mittleren Erstellungs- und Korrekturaufwand. Im Gegensatz dazu war das Korrigieren der Programmieraufgaben sehr arbeitsintensiv. Oft mussten im Nachhinein AR Toolkit-Marker-Dateien angefordert und Makefiles angepasst werden, wobei sich dieser Aufwand gegen Ende des ersten Teils reduzierte. Um sofort auf eventuelle Probleme während der Programmierung der Aufgaben eingehen zu können, wurde zudem dreimal wöchentlich eine Programmierberatungsstunde im Praktikumsraum angeboten.

Der Aufwand in der Projektphase beschränkte sich erfreulicherweise auf die Vermittlung von Projektmanagementtechniken und die Koordination der einzelnen Projektteams. Wöchentliche Projektbesprechungen mit den von jeweils einem Betreuer angeleiteten Teams wurden vor allem in der Endphase durch häufigere teamspezifische Programmierberatungen ergänzt. Gerade im Vergleich zu der früheren prozessorientierten Vorgehensweise verringerte sich der Betreuungsaufwand durch die zuvor erlernte sichere Anwendung der Grundtechniken immens.

4 Schlussfolgerung

In der Gesamtschau hat das hier vorgestellte Lehrkonzept die definierten Anforderungen voll erfüllt. Gegenüber den früher durchgeführten prozessorientierten Praktika erhöhte sich insbesondere die vermittelte Fähigkeit zur Einschätzung der Möglichkeiten und des Aufwands zur Implementierung von AR-Anwendungen, wobei die Kenntnisvermittlung im Bereich der projekt- und teambasierten Entwicklungsmethodik nur relativ wenig eingeschränkt werden musste.

Nichtdestotrotz reicht ein Praktikum nicht aus, um die volle Komplexität und die hohen Anforderungen von AR Anwendungen zu vermitteln. Weitere Vorlesungen, Seminare, Praktika und Studienarbeiten sind nötig, können fortan aber auf einer fundierten Grundlage aufbauen. Zudem erkennen die Studierenden durch die Projektphase, wie wichtig gerade bei AR-Anwendungen eine gute Dokumentation und Demopräsentation für die Vermittlung komplexer wissenschaftlicher Arbeiten ist.

Wir danken unseren Studierenden Michael Schlegel, Georg Kalus, Christopher Leopold, Leslie Klein, Markus Mann, Stanimir Arnaudov, Lyubomir Yordanov, Ivan Borisov, Johann Niklas, Christian Scheurle, Michael Siggelkow, Jens Köchling, Troels Frimor, Nicolas Padoy und Simon Bierbaum für die hochmotivierte Teilnahme am Praktikum und den Mitarbeitern des Lehrstuhls

für angewandte Softwaretechnik (Prof. Bernd Brügge, Ph.D.) für die Unterstützung bei der Bereitstellung der Infrastruktur des Praktikums.

Literatur

- [1] G. Baratoff, A. Neubeck, and H. Regenbrecht. Interactive Multi-Marker Calibration for Augmented Reality Applications. In *International Symposium on Augmented and Mixed Reality ISMAR 2002*, Darmstadt, Germany, 2002.
- [2] M. Bauer, B. Brügge, G. Klinker, A. MacWilliams, T. Reicher, C. Sandor, and M. Wagner. Design of a Component-Based Augmented Reality Framework. In *Proceedings of the International Symposium on Augmented Reality*, New York, NY, 2001.
- [3] B. Bruegge and A. H. Dutoit. *Object-Oriented Software Engineering Using UML, Patterns, and Java*. Prentice Hall, Englewood Cliffs, NJ, second edition, September 2003.
- [4] E. B. Dam, M. Koch, and M. Lillholm. Quaternions, interpolation and animation. Technical Report DIKU-TR-98/5, University of Copenhagen, July 1998.
- [5] H. Kato, M. Billinghurst, and I. Poupyrev. *ARToolKit version 2.33*, 2000. Download http://www.hitl.washington.edu/research/shared_space/download/.
- [6] G. Klinker, O. Creighton, A. H. Dutoit, R. Kobylinski, C. Vilsmeier, and B. Bruegge. Augmented maintenance of powerplants: A prototyping case study of a mobile AR system. In *IEEE and ACM International Symposium on Augmented Reality ISAR 2001*, October 2001.
- [7] G. Klinker, A. H. Dutoit, M. Bauer, J. Bayer, V. Novak, and D. Matzke. Fata Morgana – A Presentation System for Product Design. In *International Symposium on Augmented and Mixed Reality ISMAR 2002*, Darmstadt, Germany, 2002.
- [8] Gudrun Klinker, Hesam Najafi, Tobias Sielhorst, Fabian Sturm, Florian Echtler, Mustafa Isik, Wolfgang Wein, and Christian Truebswetter. FixIt: An Approach towards Assisting Workers in Diagnosing Machine Malfunctions. In *Proc. of the International Workshop exploring the Design and Engineering of Mixed Reality Systems - MIXER 2004, Funchal, Madeira, CEUR Workshop Proceedings*, 2004.
- [9] A. MacWilliams, C. Sandor, M. Wagner, M. Bauer, G. Klinker, and B. Brügge. Herding Sheep: Live System Development for Distributed Augmented Reality. In *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR)*, October 2003.
- [10] M. Tuceryan, D. Greer, R. Whitaker, D. Breen, C. Crampton, E. Rose, and K. Ahlers. Calibration requirements and procedures for a monitor-based augmented reality system. *IEEE Transactions on Visualization and Computer Graphics*, 1, September 1995.
- [11] J. Wernecke. *The Inventor Mentor: Programming Object-Oriented 3d Graphics with Open Inventor, Release 2*. Addison-Wesley Longman Publishing Co., Inc., 1993.
- [12] TWiki – A Web Based Collaboration Platform. Homepage <http://www.twiki.org>.
- [13] Übungsblätter zum AR Praktikum. Download unter <http://www.bruegge.in.tum.de/DWARF/AugmentedRealityPraktikumSoSe2004>.