

Kurzfassung der Diplomarbeit

Abteilung	Informatik
Name der Diplomandin / des Diplomanden	Ian Hlavac Oliver Pescosta
Diplomjahr	Sommersemester 2005
Titel der Diplomarbeit	„Mouse Tracker“-Bibliothek für Mobiles – Eine Machbarkeitsstudie
Examinatorin / Examinator	Prof. S.F. Keller, HSR, Abteilung Informatik
Kurzfassung der Diplomarbeit	
<p>Die Firma „Bitforge – Entertainment Software Engineering“, führend in der Java Mobile Spiele-Entwicklung in Europa, möchte künftig die Kamera, mit der heutzutage fast jedes Mobile ausgestattet ist, als Steuerungsgerät für seine Spiele verwenden. Die aufgenommenen Bilder, sollen ausgewertet werden, um schliesslich Bewegungen daraus ablesen zu können- analog der Funktionalität einer optischen Maus. Die erkannten Bewegungen werden danach als Eingabeereignis aufgefasst, womit schliesslich ein Spiel gesteuert werden kann. Damit lassen sich völlig neue Spielkonzepte realisieren. Gut vorstellbar, diese Prinzip auch auf andere Applikationstypen anzuwenden.</p> <p>Der beschriebene Ablauf soll in einem API gekapselt realisiert werden. Dieses kann von einer Applikation, die die Kamera als Eingabegerät verwenden möchte, integriert werden. Eine Bewegungserkennung wird dann als Event der Applikation zurückgegeben. Da zu Beginn des Projekts ungewiss war, ob die Performance eines Mobiles überhaupt ausreichen würde, um dieses Vorhaben umzusetzen, stand eine Machbarkeitsstudie hierbei im Vordergrund. Weil die Mobile-Version aus beschriebenen Gründen, möglichst prozessorfreundlich umgesetzt werden sollte, ging deren Realisierung eine Win32-Java-Applikation voraus, womit es möglich war, Mechanismen zur Vor- und Nachbearbeitung von digitalen Bildern nach diversen Kriterien auszutesten (Bsp. Performance, Genauigkeit). Vorverarbeitung steht hier für diverse Filtermechanismen, die ein Bild optimal auf die Bewegungserkennung vorbereiten (Bsp. Rauschfilter). Die Nachbearbeitung meint aufeinander folgende Bilder miteinander vergleichen und darin einen Fluss (Bewegung) erkennen. Für beide Mechanismen standen etliche Techniken zur Verfügung. Die Bewegungserkennung basiert primär auf dem Block-Matching-Verfahren, d.h. ein Bild wird in quadratische Blöcke aufgeteilt, welche im Folgebild gesucht werden. Man kann den Ablauf in zwei Schritte unterteilen: Zuerst wird ein Verfahren angewandt, um die Suchpunkte eines Blocks im Bild zum Zeitpunkt „t“ in einem Suchbereich des Bildes zum Zeitpunkt „t+1“ zu bestimmen. Danach vergleicht man die Blöcke aus dem Suchbereich mit den Referenzblöcken (aus Bild „t“) mittels mathematischer Operationen. Die Blöcke, die am besten zu den Ursprungsblöcken passen, dienen nun für die Bestimmung der Richtung der Bewegung, indem man für einen Ursprungsblock den Translationsvektor zum am besten passenden Block berechnet. Für eine Blockzahl von x ergeben sich somit x Vektoren, die zum Schluss noch gemittelt werden, woraus ein Bewegungsvektor zwischen zwei aufeinander folgenden Bildern resultiert. Die erkannten Bewegungsrichtungen verstehen sich dabei in digitaler Form (4 – 16 Richtungen).</p> <p>Es konnte gezeigt werden, dass die Bewegungserkennung sehr gut funktionierte, sich jedoch eine Umsetzung auf Mobiles nach dem heutigen Stand der Technik zumindest mit Java als Programmiersprache als nicht durchführbar erwies. Die Mobile-Java-Programmierschnittstellen erlauben zurzeit noch keinen performanten Zugriff auf die Kamera. Auch sind die Bewegungserkennungsalgorithmen heute noch viel zu rechenintensiv.</p>	