Abstract: Effiziente Visualisierung von Vektorfeldern in der Strahlentherapie

Jan Meis[1,2], Hendrik Teske[1,2], Kristina Giske[1,2]

[1]Division of Medical Physics in Radiation Oncology, DKFZ Heidelberg
[2]National Center for Radiation Research in Oncology (NCRO),
Heidelberg Institute for Radiation Oncology (HIRO), Heidelberg
j.meis@dkfz.de

Bewegungsdaten in der Strahlentherapie werden als zeitabhängige, dreidimensionale, diskretisierte Vektorfelder beschrieben. Deren Visualisierung stellt eine Herausforderung in den Bereichen Speichermanagement und Berechnungszeit dar. Die Darstellung eines typischen 3D-Vektorfelds mithilfe 3D Pfeil-Glyphen in jedem Voxel bei einer typischen Auflösung würde ~ 10 GB Speicher und ~ 30 s CPU Berechnungszeit benötigen. Soll ein Vektorfeld nun auch mittels anderer Darstellungsarten wie z.B. Flusstrajektorien darstellt werden um die Übersichtlichkeit zu erhöhen, so vervielfacht sich auch die benötigte Berechnungszeit trotz der Reduktion der Glyphenanzahl. Um zukünftig 4D Lungenatmung animiert darzustellen, müssen Methoden entwickelt werden um Informationen aus einem Vektorfeld zu extrahieren ohne die Balance zwischen Ressourcenaufwand und vollständiger Darstellung zu verlieren.

Wir benutzen Particle-Tracing Methoden um animierte Strömungslinien im 3D Raum zu erzeugen. Die Strömungslinien werden als Bahnen zufällig im Raum verteilter Punkte erzeugt, die dann entlang des Vektorfeldes integriert werden. Die Glyphen werden entlang der Strömungslinien animiert dargestellt. Um Überschneidungen zu verringern und plötzliches Auftauchen und Verschwinden von Glyphen zu maskieren, wurde ein Verfahren für 2D Vektorfelder [1] auf 3D erweitert. Die Methode wurde bisher single-threaded auf der CPU realisiert. Der Rechenaufwand hierfür beträgt mehrere Minuten.

Unser aktueller Fokus ist es, diesen Prozess durch GPU-Parallelisierung zu beschleunigen. Im ersten Schritt wurde die Berechnung der Darstellung der Glyphen über CUDA realisiert. Der Vorteil ist, dass hierbei die CUDA-OpenGL Interoperabilität die langsamen Transfers zwischen GPU und CPU minimiert. Die aktuelle Implementierung führte zu einer Beschleunigung in der Berechnung der Pfeil-Glyphen um 200%.

Im nächsten Schritt soll auch die Strömungslinienberechnung parallelisiert und die Darstellung auf 4D Daten erweitert werden, um eine übersichtliche und schnelle Darstellung von z.B. Atembewegungen mittels Glyphen zu ermöglichen.

Literaturverzeichnis

1. Jobard B, Ray N, Sokolov D. Visualizing 2D flows with animated arrow plots. Compute Res Repository. 2012;abs/1205.5204. Available from: http://arxiv.org/abs/1205.5204.

© Springer-Verlag GmbH Deutschland 2017

K.H. Maier-Hein et al. (Hrsg.), *Bildverarbeitung für die Medizin 2017*,
Informatik aktuell, DOI 10.1007/978-3-662-54345-0_52