Abstract

Inhaltsverzeichnis

1 Verwendung und Umfeld des JHelioviewers

Wie der Name bereits verrät ist JHelioviewer eine Applikation, die zur Analyse von Sonnendaten verwendet wird. Es wird international zur Sonnenforschung eingesetzt und wird von der FHNW zusammen mit der ESA entwickelt. Momentan ist eine neue Version des JHelioviewer in Entwicklung, welche die Sonne im dreidimensionalen Raum darstellt. Ein Feature von JHelioviewer ist die Magnetfeldlinien darzustellen und zu animieren, die Abbildung 1 zeigt die Visualisierung. Es wird zwischen drei Feldlinien Unterschieden: Linien, die auf der

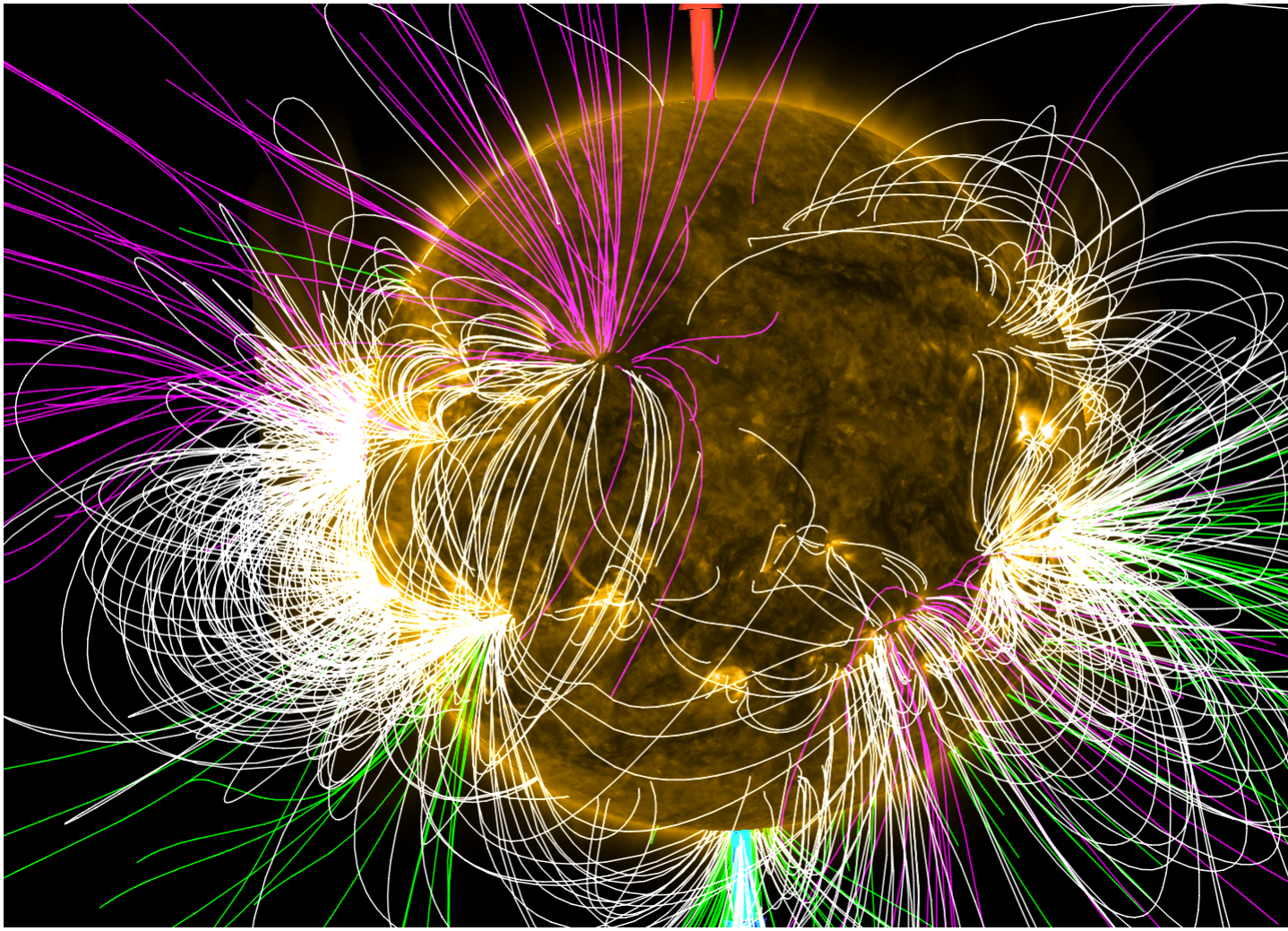

Abbildung 1: Visualisierung der Feldlinien im JHelioviewer

Sonne starten und wieder auf der Sonne landen, auf der Sonne starten und ins Weltall führen oder vom Weltall auf der Sonne landen. Die weissen Feldlinien repräsentieren ”Sonne zu Sonne´´, die Grünen ”Sonne zu Weltall´´ und die Violetten ”Weltall zu Sonne´´. Die Feldlinien sind, allgemein Betrachtet, eine grosse Menge an Punkten, welche ein Server bereitstellt. Die Abbildung 2 visualisiert den Datenfluss. In regelmässigen Abständen sucht der Server nach neuen Oberflächen-Magnetogramm-Daten der Satelliten. Alle sechs Stunden wird die Oberfläche der Sonne neu gemessen.Daraus werden mittels Potential Field Source Surface (PFSS) Extrapolation die Feldlinien zu diesem Zeitpunkt errechnet. Danach führt der Server eine verlustlose Kompression durch und stellt die Daten auf einem öffentlichen Share dem JHelioviewer zur Verfügung. Der JHelioviewer lädt dann zur Laufzeit die Feldlinien, die er benötigt.

Zukunft soll eine Feldlinienanimation angeboten werden, nicht nur alle 6 stunden sondern jede halbe Stunde.

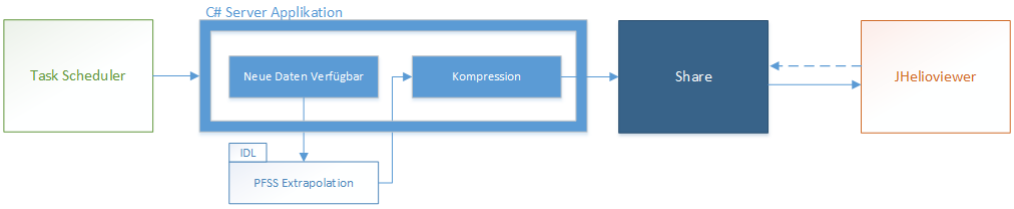

Abbildung 2: Aufbau und Datenfluss des Servers

1.1 Datenmenge ist zu gross für die Datenübertragung

Die Feldlinien der Sonne zu einem Zeitpunkt bringen etwa 1.3 MiBytes auf die Waage.

Mit einer vernünftigen Bandbreite ist das flüssige Abspielen der Animation deshalb nicht möglich. Ziel ist es eine Kompression zu entwickeln, welche eine flüssige Animation erlaubt.

Im Vorfeld möglichst verlustfrei Neu verlustbehaftet

Ziel ist es die Feldlinien verlustbehaftet zu komprimieren, sodass JHelioviewer eine möglichst flüsssige Animation anbieten kann. Die Komprimierung soll serverseitig mittels C# umgesetzt werden während der JHelioviewer mit der Dekomprimierung ausgestattet wird.

Verlustbehaftet, metriken!!

2 State of the Art

2.1 3d Mesh Kompression

Entertainement industrie. Menge an Triangles, an welche zusätzliche Informationen wie Oberflächen-Normalen, Texturkoordinaten etc gebunden sind. Bei hochaufgelösten Modelle werden die Daten schnell über 1 MiByte gross. Angenommen man möchte eine ganze Szene mit mehreren Modellen über ein Netzwerk übertragen, wird die Datenmenge schnell sehr gross.

2.2 PointCloud Kompression

Industrie Lasersampling Bild pointcloud Grosse Punktmenge, welche im 3d Raum komprimiert werden soll. verlustfrei/ Verlustbehaftet Je nach Implementation können zu jedem Punkt zusatzinfos gespeichert werden wie Farbe/Normalen etc, darüber könnte die Information, zu welcher Linie ein Punkt gehört, gespeichert werden.

2.3 Signal Approximation

Messtechnik von Medizin bis Fotografie, überall wo man ein Signal über Zeit hat Versucht ein Signal durch eine Folge von Funktionen zu approximieren.

Gute Approximation kommt mit wenigen Funktionen aus. Verlustbehaftete Kompression, indem man mit einer begrenzten Anzahl

fourier, wavelet orthogonal matching pursuit, Spline Approximation

3 Eigenschaften und Kompression der Feldlinien

Eigenschaften sphärische Koordinaten

3.1 Ist-Komprimierung

warum schon komprimiert

Sphärische Koordinaten Radius; längen und Breitengrade. Radius zwischen 1 und 4 facher Sonnenradius längen und Breitengrade 0 für 0 Grad und 1 für 360 grad.

Diskretisierung auf shorts, Subsampling 0 Löschen.

Format: zuerst Konstanten, alle Radien Verlustlose Datenkompression gzip Clientseitig wieder ein subsampling und umrechnung in kartesische xyz

3.2 Lösung 0, Clientseitiges Subsampling auf dem Server ausführen

3.3 Lösung 1, Diskrete Kosinus Transformation

DCT, da alles nahe an harmonischen Halbwellen

subsampling? koordinatentransformation Diskretisierung? Cosinus-Transformation Quantisierung Diskretisierung

4 Testsetup

Verlustbehaftete Kompression muss die Genauigkeit gewährleistet werden. Es kann immer eine höhere Kompression unter dem Kompromiss der Genauigkeit. Eine feste Grenze für die genauigkeit ist nicht immer festzulegen. Um Verfahren trotzdem zu Testen und zu vergleichen werden kompressionen mehrfach in verschiedenen qualitätsstufen getestet und jeweils den Fehler und die resultierende Dateigrösse verglichen.

Bild eines Beispielgraphen

4.1 Auswahl und Erhebung der Testdaten

Testdaten sollen zu einem Randfälle abdecken, als auch durchschnittliche Fälle enthalten. Insgesamt wurden 10 Datensätze ausgewählt. Nach grossen Solar Flares gesucht vier Datensätze mit grossen Flares, zwei mit sehr wenig Sonnenaktivität und vier zufällig. Die feldlinien werden aber nur alle sechs Stunden berechnet und ein Flare ist ein kurzes ereignis. Es wurden die Datensätze vor dem Ereignis ausgewählt.

wie im Abschnitt 3.1 beschrieben, führt der IDL-Code schon eine Quantisierung durch. deshalb wurde der IDL Code angepasst, quantisierung entfernt und dateien mit float genauigkeit genommen.

Problem mit letzter linie

4.2 Messung des Fehlers

architektur?

5 Implementation

5.1 Komprimierung auf dem Server

C# architektur des servers, nicht viel. neuer Ordner auf Share

5.2 Asynchrone Dekomprimierung im JHelioviewer

alter zustand?

neuer Zustand preload (precache) asynchrone architektur Klassendiagramm

6 Resultate

6.1 Lösung 0, Clientseitiges Subsampling auf dem Server ausführen

5grad

6.2 Lösung 1, Diskrete Kosinus Transformation

was ist dct? (DCT2 wird verwendet) Warum DCT, Ein Signal wird durch eine Folge von Cosinusfunktionen angenähert

Was für Eigenschaften hat die DCT der Feldlinien? JPEG Standard wegen -x bis x, einfluss auf die DCT Koeffizienten Residuals1 Einfluss auf die DCT Einfache Quantisierung und besondere Quantisierung Residuals2 Einfluss auf die DCT

Was für ein Koordinatensystem eignet sich? Warum Sphärisch: Kleinere Koeffizienten. Warum Euklidisch: Kein Wrap-around.

Vergleich von Sphärisch und Euklidisch Kilometer oder Meter? Fehler mit Diskretisierung

Kann das Client Subsapmling verwendet werden 5grad währe schön, viel kleinere Punktmenge, aber nicht konstante Abstände

7 Diskussion

8 Fazit

Abbildungsverzeichnis

Tabellenverzeichnis

9 Anhang

subsectionInstallationsanleitung

10 Ehrlichkeitserklärung