Realtime Cloud Screening with AVIRIS-NG

David R. Thompson, Robert O. Green, Didier Keymeulen, Sarah K. Lundeen, Yasha Mouradi, Daniel Nunes, Rebecca Castañó, Steve A. Chien

Jet Propulsion Laboratory California Institute of Technology

Copyright 2013 California Institute of Technology.
All Rights Reserved. Government sponsorship acknowledged.
The Earth is a cloudy place

[Mercury et al. 2012]
Why onboard cloud screening?

• Spectrometer data rates approach 1Gb/s
• This imposes costs on buffering, transmission, analysis and curation
• For many analyses, 50-70% of scenes are unusable due to clouds [Eastman 2011]
• Excising bad data at the sensor benefits the entire processing and analysis pipeline
• Can be used to point gimbaled sensors
Previous cloud screening methods

- MODIS cloud mask
  [Ackerman 08]
- Physical Models
  [Gómez-Chova 07, Taylor 12]
- Thermal IR
  [Minnis 08]
- Pattern recognition
  [Lee 90]
- Onboard EO-1
  [Griffin 03]
Our unique requirements

- Use raw instrument DN values
- Low computational complexity
- Operate on-board in real time on existing computing hardware
- Excise opaque, non-cirrus clouds
- Reduce average data volume by 50% (based on a continuous year of ISS operations)
- Negligible false positives
Concept of Operations

In advance
1. Model brightness of clouds and terrain
2. Use imaging geometry to predict optimal channel thresholds

Onboard
1. Apply thresholds to recognize cloud pixels
2. Excise image blocks with a large number of cloudy pixels
Simulation dataset

AVIRIS Classic high-altitude (ER-2) flights 2009-11
Labeled all cloud pixels by hand, excluding sunglint
Used half for training, half for testing

<table>
<thead>
<tr>
<th>Land cover</th>
<th>Clear pixels</th>
<th>Cloudy pixels</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Water</td>
<td>$6.7 \times 10^8$</td>
<td>$2.2 \times 10^7$</td>
<td>UMD</td>
</tr>
<tr>
<td>Evergreen needleleaf forest</td>
<td>$3.7 \times 10^7$</td>
<td>$1.4 \times 10^6$</td>
<td>UMD</td>
</tr>
<tr>
<td>Evergreen broadleaf forest</td>
<td>$5.8 \times 10^6$</td>
<td>$5.8 \times 10^5$</td>
<td>UMD</td>
</tr>
<tr>
<td>Deciduous needleleaf forest</td>
<td>-</td>
<td>-</td>
<td>UMD</td>
</tr>
<tr>
<td>Deciduous broadleaf forest</td>
<td>$2.1 \times 10^8$</td>
<td>$4.4 \times 10^6$</td>
<td>UMD</td>
</tr>
<tr>
<td>Mixed forest</td>
<td>$1.3 \times 10^8$</td>
<td>$3.7 \times 10^6$</td>
<td>UMD</td>
</tr>
<tr>
<td>Closed shrublands</td>
<td>$1.1 \times 10^6$</td>
<td>$6.5 \times 10^5$</td>
<td>UMD</td>
</tr>
<tr>
<td>Open shrublands</td>
<td>$1.3 \times 10^8$</td>
<td>$1.8 \times 10^6$</td>
<td>UMD</td>
</tr>
<tr>
<td>Woody savannas</td>
<td>$1.3 \times 10^8$</td>
<td>$2.6 \times 10^6$</td>
<td>UMD</td>
</tr>
<tr>
<td>Savannas</td>
<td>-</td>
<td>-</td>
<td>UMD</td>
</tr>
<tr>
<td>Grasslands</td>
<td>$8.9 \times 10^7$</td>
<td>$3.1 \times 10^6$</td>
<td>UMD</td>
</tr>
<tr>
<td>Croplands</td>
<td>$1.0 \times 10^8$</td>
<td>$7.3 \times 10^6$</td>
<td>UMD</td>
</tr>
<tr>
<td>Urban and built-up</td>
<td>$3.1 \times 10^7$</td>
<td>$2.9 \times 10^4$</td>
<td>UMD</td>
</tr>
<tr>
<td>Snow and ice</td>
<td>$1.3 \times 10^8$</td>
<td>$4.4 \times 10^6$</td>
<td>UMD</td>
</tr>
<tr>
<td>Barren</td>
<td>$9.7 \times 10^7$</td>
<td>$1.7 \times 10^2$</td>
<td>UMD</td>
</tr>
<tr>
<td>Ocean glint</td>
<td>$3.6 \times 10^8$</td>
<td>$1.5 \times 10^7$</td>
<td>UMD</td>
</tr>
</tbody>
</table>
Brightness model

Top of Atmosphere (TOA) reflectance

\[ z = \frac{\pi d^2}{\cos(\theta)s} g(y - b) \]

Instrument DN

Solar illumination term

Gain

Dark current

Store \( z \) values in a histogram
Calculating optimal thresholds

Threshold $\phi$  
Excluded region $\mathcal{R}$

Counts

Channel value $y_i$

Terrain pixels

Cloud pixels
Calculating optimal thresholds

Must set thresholds on multiple channels at once

Right: Lenient and strict false positive costs
Calculating optimal thresholds

Use Bayesian decision theory to balance false negative and false positive risks

\[
E[\mathcal{L}] = \int_{\mathcal{R}} \alpha_{FP} P(y | x, c_1) P(c_1) dy + \int_{\mathbb{R}^d \setminus \mathcal{R}} \alpha_{FN} P(y | x, c_2) P(c_2) dy
\]

- **Expected loss**
- **False positive cost**
- **False negative cost**

Acceptance region is free parameter

\(C_1 = \text{clear}, \ C_2 = \text{cloudy}\)

\(x = \text{state vector}, \ y = \text{observation}\)
Calculating optimal thresholds

$\alpha_{FP}$ parameter represents our tolerance for false positives

A grid search identifies optimal threshold pairs
Brightness vs. SZA

- Cloud (0.45µm)
- Cloud (1.65µm)
- Terrain (0.45µm)
- Terrain (1.65µm)
- Model

Solar zenith vs. Dark-subtracted Instrument DN

2013 HyspIRI Science Workshop - Realtime cloud screening
Which channels?

- Mutual Information (MI) scores channels’ utility with respect to cloud/non-cloud classification.
- The combination of 0.45 $\mu$m and SWIR channels performs best.

<table>
<thead>
<tr>
<th>Channel</th>
<th>MI</th>
<th>0.66 $\mu$m</th>
<th>0.86 $\mu$m</th>
<th>1.25 $\mu$m</th>
<th>1.38 $\mu$m</th>
<th>1.65 $\mu$m</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.45 $\mu$m</td>
<td>0.45</td>
<td>0.50</td>
<td>0.50</td>
<td>0.63</td>
<td>0.59</td>
<td>0.63</td>
</tr>
<tr>
<td>0.66 $\mu$m</td>
<td>0.44</td>
<td>0.50</td>
<td>0.51</td>
<td>0.62</td>
<td>0.59</td>
<td>0.62</td>
</tr>
<tr>
<td>0.86 $\mu$m</td>
<td>0.43</td>
<td>0.61</td>
<td>0.62</td>
<td>0.58</td>
<td>0.61</td>
<td></td>
</tr>
<tr>
<td>1.25 $\mu$m</td>
<td>0.54</td>
<td>0.60</td>
<td>0.61</td>
<td>0.58</td>
<td>0.61</td>
<td></td>
</tr>
<tr>
<td>1.38 $\mu$m</td>
<td>0.54</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.65 $\mu$m</td>
<td>0.40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Performance: channel selection

Screening efficiency (fraction of cloudy blocks excised)

- 0.45 µm and 1.65 µm
- 0.45 µm and 1.25 µm
- 0.45, 1.25, and 1.65 µm

False excision rate

0 1 2 3 4 5 x 10^{-3}
Performance: terrain types
Performance: solar normalization

Screening efficiency (fraction of cloudy blocks excised) vs. False excision rate

- Normalized
- Raw

Screening efficiency improves with solar normalization.
Performance: spatial aggregation

![Graph showing screening efficiency (fraction of cloudy blocks excised) vs false excision rate for full, half, and quarter swaths, with a shaded area indicating better performance.]
Performance vs. linear classifier

![Graph showing screening efficiency vs. false excision rate for two-channel threshold and linear classifier.]

- Screening efficiency (fraction of cloudy blocks excised)
- False excision rate

Better performance for the two-channel threshold compared to the linear classifier.
ISS Orbit simulation

Simulated a continuous year of operations on the ISS, ignoring sun glint

50% data volume reductions are achievable
AVIRIS-NG demonstration

• A second computer in parallel with the main recording path
  – National instruments PXIe (4 CPU cores)
  – Matlab on Windows OS

• Computes thresholds from GPS data at the start of each flightline

• Performs cloud screening online in real time at 0.5 Gb/s
AVIRIS-NG demonstration results

- Evaluated performance during a campaign near Casper, WY
- Real time on-board operation with live monitoring
- Autonomous operation transparent to Operator
- 142 flightlines total over 1TB
- Committed no false positives during the one week campaign
- Excised clouds from a handful of cloudy images (perfect performance)
Conclusions

• For typical Earth orbiting missions, realtime cloud screening could provide >50% data reduction with negligible false alarms

• Can turn the system off over snow for additional confidence

• Bayesian decision theory is a principled way to set channel thresholds

• Algorithm has very low computational complexity and integrates easily with real-time instrument data processing
Thanks!

• Charles Sarture and the AVIRIS team
• Dr. Pantazis Mouralis, Jason Hyon and the JPL Earth Science Directorate
• MODIS land cover data were obtained through the online Data Pool at the NASA Land Processes Distributed Active Archive Center (LP DAAC), USGS/Earth Resources Observation and Science (EROS) Center, Sioux Falls, South Dakota
• This project made use of Vowpal Wabbit, a fast out-of-core learning system sponsored by Microsoft Research, Yahoo! Research, and a large open source development community