AST2210 Report 1

Andreas Ellewsen

September 13, 2014

0.1 Assignment

In short the whole lab is about extracting data from the Hinode database. We are also supposed to interpret this data and plot som pictures underways. Of the things we did we were to include in this report how we did the following 3 exercises:

Exercise 3: Read into IDL the SOT/BFI synoptic images from Exercise 2. Compare the images in the various wavelengths. Look for image artifacts. List the minimum value, maximum value, mean and standard deviation of each image.

Exercise 7: Extract a few images from the timeseries and determine the average motion of the limb. Store the average displacement per timestep in the variables dx and dy. Make an array with this information and correct the time-series for this drift.

Exercise 8: Spicules have an apparent motion outwards. Find a good example in the cube cahalign.icube and measure the outward velocity.

0.2 Exercise 3

Here we follow the instructions from Exercise 1 and 2. We're asked to find some values for each of the images we load into IDL. This is done by using the code we're given in the assignment. The values can be seen in the table below. When comparing the images we see that there are some dead pixels. There is also a difference in brightness in them as well as the left and right side in each of them. The clearest error in all of them is the vertical line in the middle.

Image	Max	Min	Mean	Std dev
red cont 6684	1608	0	964.102	63.2199
green cont 5550	1309	0	896.604	69.5269
blue cont 4504	1149	92	759.703	79.3695
G band 4305	1310	93	745.198	67.4408
Ca II H line	1262	96	431.999	36.7945
CN bandhead 3883	1973	20	1018.62	114.788

Table 1: Values for all the pictures

The pictures and the code used to generate all of this are included at the end of the report.

0.3 Exercise 7

Here we do exactly what we're told. Basically copying the code from the assignment. We noticed in the last exercise, that the limb of the sun was moving. We want to fix this and we do this by looking at the displacement per timestep and store it in the variable dx. It is not necessary (like the assigment says) to save anything for dy since there is no change in the y direction. This as going to make the next exercise easier.

The code is included at the end of the report.

0.4 Exercise 8

Now that we've got the sun to stand still, we want to measure the velocity of one of the specules. This was not as easy as expected. I ended up just finding one that was at least a little clear and just time the number of frames it took to move a distance of some pixels. I looked at frame number 115 at position (799,653). I then looked at this spicule until it dissapeared. This happened at frame 116 when it reached position (784,653). So in 1 frame it moved 15 pixels. We see from the Hinode data we gathered that the time between frames is 16 seconds

The lecture notes say that for small p we can say that

$$p = \frac{a}{r} \tag{1}$$

The Hinode data tells us that Hinode has a resolution of

$$0.0541 \frac{arcseconds}{pixel} \tag{2}$$

This gives us

$$15 \times 0.0541 arcseconds = tan \frac{r}{1AU}$$
$$r = 15 \times 0.0541 arcseconds \times 1AU$$

and the velocity of the spicule is thus

$$\frac{r}{t} = 3.93426302 \times 10^{-6} \frac{1AU}{16seconds}$$

converting this to familiar units gives

$$\frac{r}{t} \approx 36784 \frac{m}{s}$$

0.5 Conclusion

In conclusion I thought I'd make a list of the things we learned in this lab.

- How to access the data from Hinode.
- How to load this data into IDL.
- How to look at spesific attributes about each image.
- How to fix artifacts and other irregularities in one, or many images.
- How to save these images to the harddrive.
- How to load a series of images into Crispex and save this as a movie.
- How to force the video to keep a part of the sun at a specific place in the video.
- Finally we made a guess at the velocity of a spicule.

0.6 Code

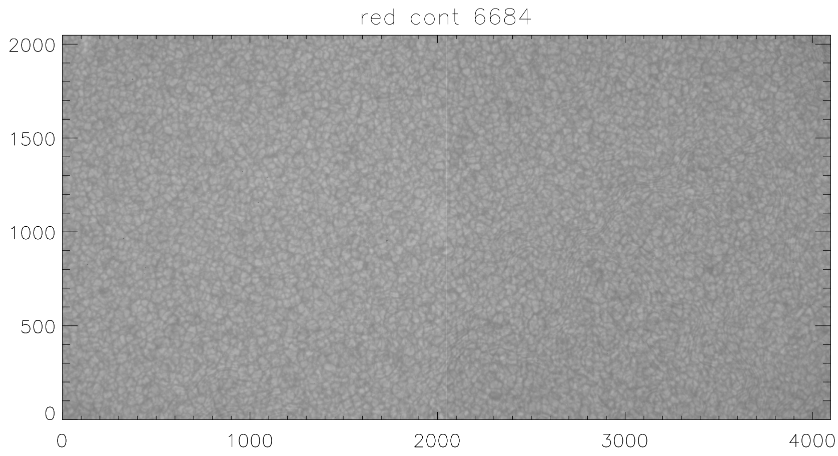
0.6.1 Exercise3

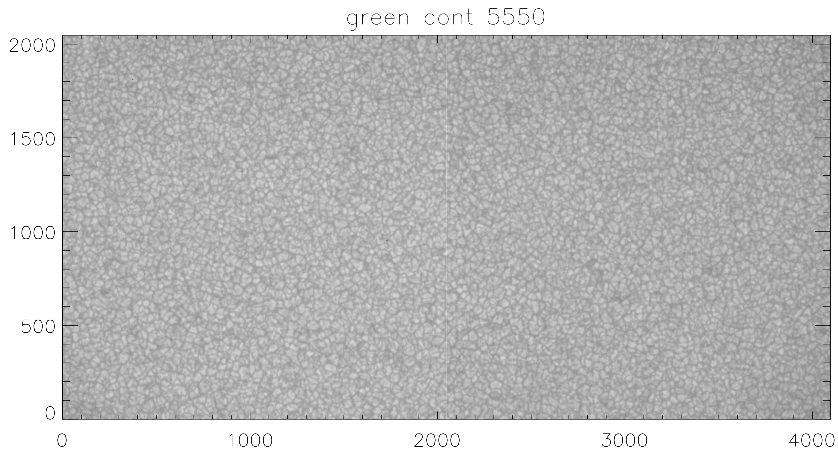
```
;Oblig1 AST2210
;This assignment goes through how to read images from the
;hinodedata.

;Exercise 3

;Read files into IDL
o=obj_new('osdc')
o->show, 'INSTRUME,DATE_OBS,SX__PROG_VER,THUMBS'
o->page,1
o->limit,50 ;; Use o->limit,0 to get all!
o->order, 'DATE_OBS',/DESCENDING
o->condition, 'INSTRUME: ,SOT/WB'
o->condition, 'EPOCH_START: 01-Apr-2007'
o->condition, 'EPOCH_END: 30-Apr-2007'
o->condition, 'SS_L1LEAD: y'
o->condition, 'SX__PROG_VER: 52'
o->condition, 'SQ_OBSTITLE_PROG_VER: FG20070429_182200.6'
o->search,out
paths = o->paths(out,/fetch) ;; /unzip has been deprecated
paths=paths[1:*]
read_sot,paths,index,data

;Plot the images, set title of each of them, then save
for i=0,5 do begin
image = data[*,*,i]
window,i
savep,'file'+strtrim(i,2)
plot_image,image,title=index[i].wave
savep

;Print information for each image to terminal
print,'Info for image '  + strtrim(index[i].wave)
print,'Maximum:  '       + strtrim(max(image),2)
print,'Minimum:  '       + strtrim(min(image),2)
print,'Mean:     '       + strtrim(mean(image),2)
print,'Std dev:  '       + strtrim(stddev(image),2)
print,' '

endfor

end
```

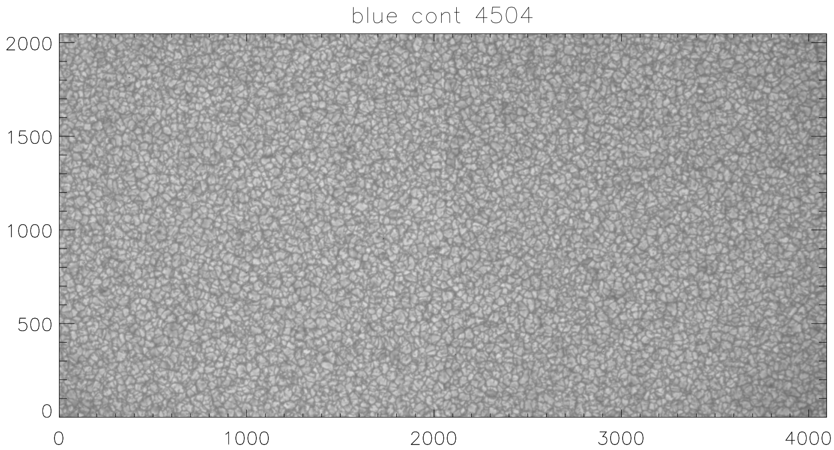
0.6.2 Exercise7

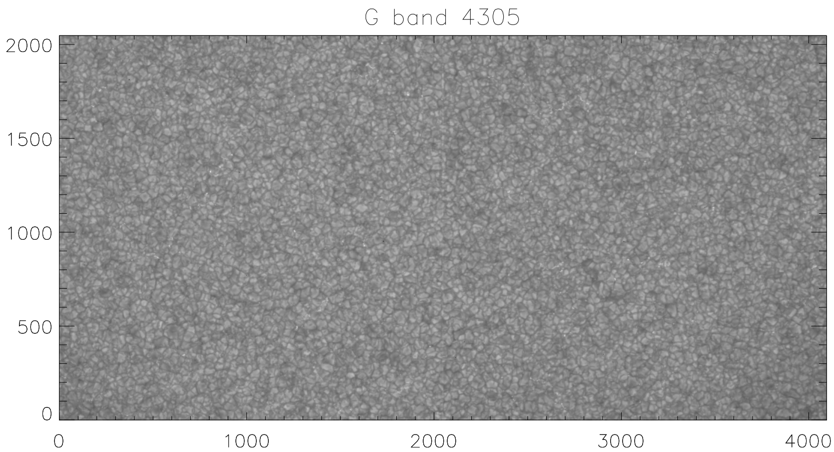
```
;Oblig1 AST2210
;Exercise 6

;Read files into IDL
o=obj_new('osdc')
o->show, 'FILE,INSTRUME,DATE_OBS,CEN_RADIUS,S_WAVE,THUMBS'
o->page,1
o->limit,0 ;; Use o->limit,0 to get all!
o->order, 'DATE_OBS',/ASCENDING
o->condition, 'INSTRUME: ,SOT/WB'
o->condition, 'CEN_RADIUS: >1000'
o->condition, 'SS__L1LEAD: y'
o->condition, 'Gx: NONE'
o->condition, 'S__WAVE_ID_: ,7'
o->condition, 'xSQ_IUMODE1: FG20130930_193029.5'
o->search,out
paths = o->paths(out,/fetch) ;; /unzip has been deprecated
paths=paths[1:*]
;read_sot,paths,index,data

;fg_prep,index,data,index2,data2

mfg_mkcube,paths,'cah'
restore,'cah.idl',/verbose
save,hdr,file='cah.idl'
crispex,'cah.icube', /single

image=lp_get('cah.icube',20) ; read image number 20 from the timeseries
end

;We do as the assignment tells us to do, choosing CEN_RADIUS, SOT
;search criterium S__WAVE with Ca II H Line. We use the example code
;given in the assignment and start the program. After about 15 minutes
;the program is done and the crispex window opens. It turns out
;I've made a rather poor choice when selecting which series of
;images to use. We do see very clearly that the limb moves throughout
;the movie.

;Exercise7:
lp_header,'cah.icube',nt=nt ; determine the number of images
dxy=fltarr(2,nt) ; make a dxy-array filled with zeroes
dx = (809-786)/100.
dy = 0
dxy[0,*] = dx ; average displacement per timestep in x direction
dxy[1,*] = dy ; average displacement per timestep in y direction
mfg_shift,'cah.icube',dxy ; compensate for the drift

end
```

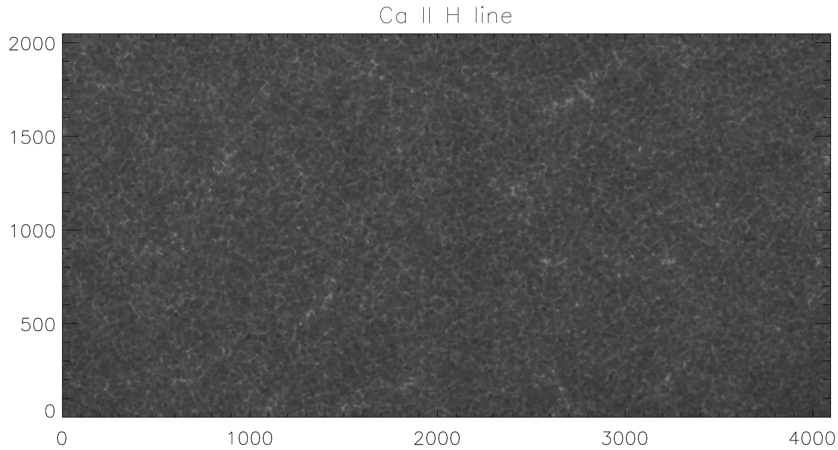
0.7 Pictures from Exercise 3

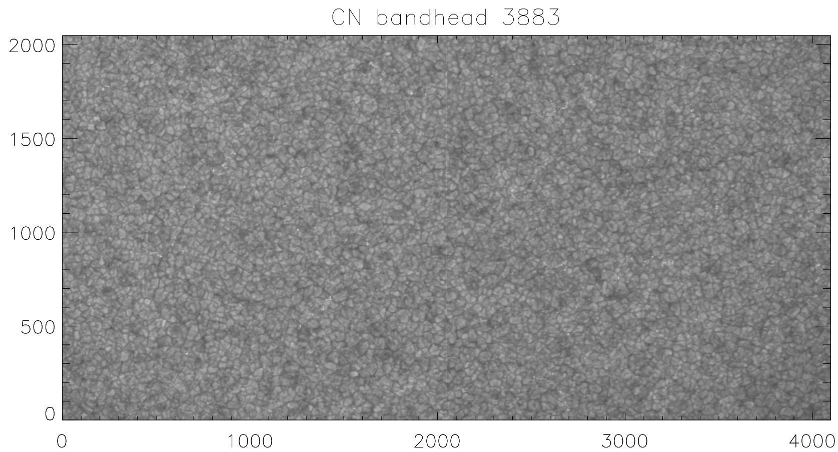





blue cont 4504
2000
1500
1000
500
0
0
1000
2000
3000
4000


G band 4305
2000
1500
1000
500
0
0
1000
2000
3000
4000


Ca II H line
2000
1500
1000
500
0
0
1000
2000
3000
4000


CN bandhead 3883
2000
1500
1000
500
0
0
1000
2000
3000
4000