

Project Overview

The Polar Bear Telescope consists of 3 camera systems, each comprising

1. an environmentally controlled weather-proof housing with a heated glass window (see http://www.videotec.com/en/page_182.html)
2. a Nikkor 85 mm f1.4 lens (stopped to f1.8) (see



Illustration 1: The Polar Bear Telescope

http://www.mir.com.my/rb/photography/companies/nikon/nikkoresources/AFNikkor/AF85mm/AF-Nikkor85mmf14D_US.jpg)

3. a Hoya PRO1 Digital UV Filter (see <http://www.hoyafilter.com/products/hoya/pro1d-03.html>)
4. a spacing tube
5. a Starlight Xpress HXV-16 CCD Camera(2048x2048 pixels) (see <http://www.starlight-xpress.co.uk/SXV-H16.htm>)
6. a Dell Studio minicomputer running Windows XP

The entire system is mounted rigidly on the Observatory roof. Each Starlight Xpress HXV-16 camera use a Kodak KAI4021M CCD and a two stage Peltier cooler to reduce the detector temperature 10 degrees below ambient.

Each camera has a field of view some 10 degrees square. Each field is centered at a position spaced 120 degrees apart around a circle some 4 degrees from the North Celestial Pole, and aligned so that the rotation of the earth will bring a star across roughly the same part of each CCD camera. The cameras have distinct USB identifiers (126, 128 and 129), a feature originally intended to allow multiple cameras to be controlled from one PC.

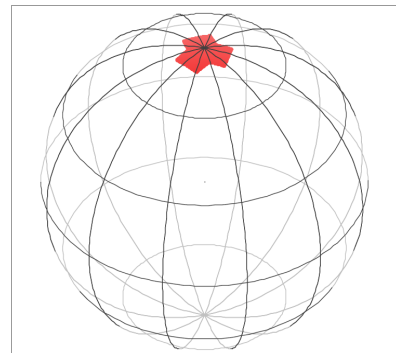


Illustration 2: The PBT combined field of view

The control computers for each camera are called, respectively, spoc1, spoc2 and spoc3 (spoc = Simon's POLar Camera). These computers may be controlled from anywhere in the Observatory using a remote desktop viewer such as vncviewer . The current status of the spocs and recent images can be checked from the polar bear inspector, located at <http://arpc65.arm.ac.uk/polarbear/frames-index.html>.

Observing Procedure

Hardware Power On/Off:

The observing procedure is designed to be entirely automatic:

The Starlight Xpress Cameras are switched on and off by a timeswitch in the camera control box. The switches need to be adjusted every month to minimise the time the cameras are running but to ensure that starry skies are recorded.

Following any break in the power supply, the RCD socket and the timeswitch clocks in the rooftop box need to be checked and, if necessary, reset.

Automatic operation under pywinauto

[this feature is not currently working correctly. As of 2009 December, all spocs are running continuously.]

The control computers (spocs) can be set to hibernate at dawn, and wake up again at dusk; the wakeup is achieved using a program called WakeUpOnStandby (<http://www.dennisbabkin.com/php/download.php?what=WOSB>). The wakeup time is set each night by a python script. The wakeup and sleep times for all three cameras are read from the file */starlight/pbt_timings* which can be changed as required.

If any of the spocs are not operating, and they are in hibernate mode, they can be woken up using one of the scripts */starlight/pbscripts/wolspoc[n].py* (eg `> python wolspoc1.py`).

The images are read out using Starlight Xpress software (SXV_Hmf_USB - HX Camera Image). The camera software is automated by a python script using the pywinauto library, (<http://pywinauto.openqa.org/>) which interfaces with the Starlight GUI directly.

To start this software, it is necessary to "vnc" into each spoc, and use the *run* command to start *runstarlight.py*. In order for the automatic wake-up and pywinauto to work properly, the spocs must be shut down cleanly by *runstarlight* (which use the times set in *pbt_timings*).

If you start any of the spocs during the day, then you should end up by running the *runstarlight* script. This will detect that it is daytime, hibernate the spoc, and leave it to wake-up at the prescribed time.

In some cases, Starlight Xpress software appears to crash, and the errors are not currently trapped by *runstarlight*. In this case the spocs do not shut down cleanly, and the process must be restarted manually. Observing time will be lost in this case. Current experience suggests that memory leaks and suspended programs can cause problems; we are trying to eliminate these glitches. Rebooting the spoc seems to help.

Image Capture

Exposures are taken continuously through the night. Currently, every exposure is 30 seconds long. It takes 3 - 4 seconds to read out each image. It should be possible to take a repeating sequence of exposures of different lengths, e.g. 2s, 10s, 20s... by adapting the current setup.

The raw images are stored directly to a RAID array in the basement. The main folder is mounted as */starlight* on arpc32 and arpc55. Each camera writes out its images to a different folder, labelled */starlight/spoc1*, */starlight/spoc2* and */starlight/spoc3*

The filenames are IMGnnnnn.FIT, where nnnnnn is a sequential number which cycles back to 1 after 99999 images have been obtained. The images are in FITS format, the FITS headers contain rudimentary information including the exposure time and the time of observation.

A [single image from the previous night \(spoc1, midnight\)](#) is automatically copied and displayed. The [midnight images from all three cameras](#) can also be inspected quickly to ensure that the systems is operating nominally.

Recent frames can be selected and examined individually using the [image inspector](#).

[Time]

This is a stub. Need to add information about how and which times are recorded in image headers, and since when they have been valid. The spocs only had a proper time service since mid-October.



Illustration 3: A fireball observed by PBT

The filenames produce from pbt_imgname are from the file creation time on the RAID, The TIME-OBS from the FITS header are the start of exposure time (so there is a difference). Are we using UT or BST?]

Data Processing

The Tools Used

- *ftools*

ftools is a piece of NASA HEASARC software developed by the Smithsonian Astrophysical Observatory. A complete break down of software installation, applications of the program, and further information is available at http://heasarc.gsfc.nasa.gov/ftools/ftools_intro.html. Particularly important tools: farith, fcarith, fimgstat, fkeyprint

- *SExtractor*

short for Source Extractor, this is a program written by Emmanuel Bertin. It builds a catalogue of objects from an astronomical image. It is particularly oriented towards reduction of large scale galaxy-survey data, but it can perform on moderately crowded star fields. Little documentation is available from the program publisher, but this manual:

http://astroa.physics.metu.edu.tr/MANUALS/sextractor/SE_Handbook.4.ps.gz has proven very useful.

- *solve-field*

solve-field is an astrometric calibration tool, information about which is available here:

<http://trac.astrometry.net/browser/trunk/src/astrometry/README>. The program includes a source extractor called image2xy.

- *STILTS*

'STILTS' is an acronym for Starlink Tables Infrastructure Library Tool Set, and the program was developed at the University of Bristol. Information about application can be found at:

<http://www.star.bris.ac.uk/~mbt/stilts/>. **in progress.

- *GAIA; ds9*

Another acronym, 'GAIA' stands for Graphical Astronomy and Image Analysis Tool. It is an interactive image display tool. See <http://star-www.dur.ac.uk/~pdraper/gaia/gaia.html> for more detail. The other image display tool used was ds9, developed at Harvard as an imaging and visualization application. See <http://hea-www.harvard.edu/RD/ds9/> for downloads and manuals.

Preprocessing

In order to manage the data, two "cron" jobs run every morning on arpc55.

1. 15 07 * * * ~/polar_bear/pbt_imgname >> ~/polar_bear/pbt_cronjobs.log: This job reads the file creation time for each image and creates a new filename in the form: ccc_yyyymmdd_hhmmss.fit where ccc = camera number, yyyy = year, mm = month, dd = day, hh = hour, mm = minutes, ss = seconds. [pbt_imgname could be made redundant by modifying runstarlight.py to name images as they are recorded]
2. 30 07 * * * ~/polar_bear/pbt_clean >> ~/polar_bear/pbt_cronjobs.log: This job reads through every file and identifies the number of stars visible on each image. The following files are deleted:
 - All files taken during "daylight" hours (the script currently has to be told when these change).
 - All files with fewer than c. 100 stars are deleted.

- All files with more than c. 15000 stars are deleted

These are files with a fixed pattern noise obtained either when the cameras were switched off or when they were exposed in daylight and hence completely saturated. The remaining files are moved to a folder called `/starlight/yyyymmdd_ccc` and deleted files are currently not deleted, but moved to a folder `/starlight/spoc*n*/bucket/` until `pbt_clean` is proven to be working optimally.

- BUG ** `pbt_clean` currently puts all frames from the same calendar date into the same folder. This applies to dates since sometime in May. The original design was to put all frames from the same night into the same folder, with the calendar date corresponding to the date on which the night started ** Needs fixing **

Evaluation

Data may be easily inspected using any image viewing programme which recognises FITS file formats. The most useful are those designed for working with astronomical images including GAIA (Starlink) and ds9 (SAOImage).

It is worth checking random images through the night to check that the pre-processing was successful. It is also worth updating the observing log regularly, including any helpful comments about observing conditions and so on. This provides an extremely valuable record when it comes to processing the data and assessing photometry extracted many months subsequently.

The observing log is at `/starlight/pbt_data_checklist.ods`. It can be saved as xhtml and viewed online -- eg PBT Observing Log. [The observing log should be updated automatically by the various tasks it reports and the xhtml version needs to be automatically updated from the base version.]

Calibration Frames

BIAS and DARK

A bias exposure is¹ an image taken with the camera shutter closed and the shortest possible exposure time. The bias shows the electronic noise of the camera and possible systematic noise. Dark current varies directly with temperature. When electrons have enough kinetic energy to end up in pixels without an activating photon, noise that varies from pixel to pixel and frame to frame is generated. This is called dark current. Bias frames contain the dark current. For PBT there are two complications with the bias frames, being that the current software only allows a minimum exposure time of 1/1000 s and that the cameras are always open to the sky. The procedure used for taking bias frames was:

1. Cover the telescope enclosures with a secure opaque material. Avoid direct sunlight as the cameras are sensitive in the infra-red.
2. Obtain several ($\gg 10$) exposures in each camera with an exposure time of 1/1000s
3. Move the images for camera ccc to a (new) folder `/starlight/bias_YYYYMMDD_ccc/`

FLAT FIELD

The efficiency of a CCD may differ from pixel to pixel. A flat field is an exposure of a homogeneously lit area which enables variations in camera sensitivity to be removed. These variations include pixel-to-pixel variations in CCD sensitivity due to imperfection in the CCD manufacture and large-scale variations due to the camera optics. In the case of PBT, optical vignetting causes severe loss of sensitivity at the periphery of the image. The normal way to obtain a flat-field is to take an image of the twilight sky or a white screen inside the dome. PBT only has

¹ Lots of information regarding CCD data reduction can be found here: <http://www.astro.uni-bonn.de/~mischa/datareduction/noise.html>. Lots of information regarding different calibration frames and noise is available here.

the twilight sky option. Since the field of view of each camera is large (10 degrees), even the twilight sky is not perfect (there is a large brightness gradient from one side of the sky to the other after the Sun has gone down). The presence of clouds and/or stars renders a flat-field image useless, as does the presence of saturated pixels. It is necessary to obtain the flat fields while the sky is still bright enough that no stars are visible, but not so bright that the images are saturated. To obtain flats with PBT:

1. Check that the cameras are switched on no later than 15 minutes after sunset
2. About 30 minutes after sunset, start obtaining images with exposure times of 1/100 second in all cameras
3. Discard images while they are still saturated - to save disk space
4. Watch the images as they start to desaturate
5. Time frame in which well-exposed flat fields may be obtained is about 5 minutes (or less)
6. Aim to obtain at least 10 frames (in each camera) with a maximum around 20000 counts
7. (Optional) Increase the exposure time to obtain more images
8. Give up when Polaris is visible
9. Move the images for camera ccc to a (new) folder */starlight/flat_yyyyymmdd_ccc/*

Note:

On 2009/02/10 -- experienced problems with flat-field images for camera 126 (spoc1). The images remain saturated long after the other cameras have reached astronomically low light levels, and there is severe vertical banding on the images. With reducing light levels, the saturation cleared from the center of the image, but persisted at the edges. With very low light levels, the camera returned to normal. It is not clear whether this is a persistent or intermittent problem.

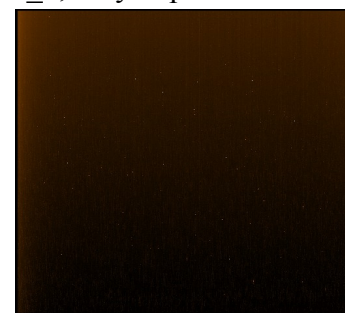
On 2009/02/13 -- problem persists, though flat taken on uniform cloud at low light levels might be usable in extremis.

Calibration Frame Processing

Location of calibration frames: */data/starlight/Calibrations*

Bias frames were taken on 2009-02-06 and on 2009-02-13, most of which used the lowest achievable exposure time (0.001). Within the set of bias frames taken on 2009-02-06, some had exposure time 0.001, others 0.002 and others 0.5. In the set taken 2010-02-13, all had exposure time 0.1. For those taken 2009-02-06, the higher exposure times were ignored. To create master bias frames, *ftools* was used to sum images and then divide the summed image by the number of frames used, *N*. These master biases include only exposure times of 0.001. For 126_2, only exposure times of 0.1 were included.

The bias frame created for camera 129 appeared as a dark frame with randomly scattered bright pixels at an average of about 30 counts above the background. On further examination, it became evident that these bright pixels appeared at the same *x* and *y* coordinates for camera 129 on the individual bias frames and on the individual flat field frames. It was decided that the bias should continue to contain these 'bad' pixels. Ideally, bias frames should be taken much more frequently, allowing a clearer view of whether problems like these are transient.



*Illustration 4:
final_bias129.fit*

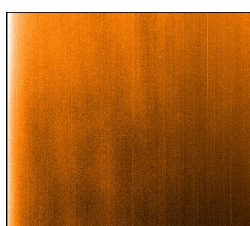
Flat fields were taken in early February 2009 and 2010. To choose suitable flat frames to make the master, every fifth frame in the set of flats was examined. Data was tabulated in a table with headers "Image Number", "Time", "Exposure Time", "<counts>", "Polaris max", and "Comments". Frames with <counts> between 11000 and 59900 were used (depending largely on the range for each set of data).

Data on Bias Frames (Table 1)

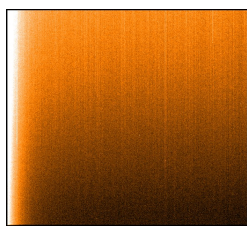
Camera Number	Name of Master Bias	Date and Time Frames Taken	Exposure Times of Frames	Number of Frames Used (N)	Mean Pixel Value	Standard Deviation of Pixel Value
126	final_bias126	2009-02-06 @ 14:20-14:52	0.001	114	628.283	9.828
126	N/A	2009-02-13 @ 18:01-18:09	0.1	204	6505.06	593.5
128	final_bias128	2009-02-06 @ 14:20-14:52	0.001	129	2285.23	10.12
129	final_bias129	2009-02-06 @ 14:20-14:52	0.001	115	2168.47	22.19

Data on Flat Fields (Table 2)

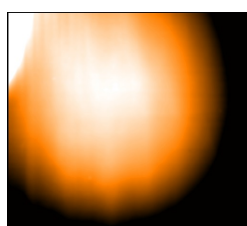
Camera Number	Name of Final Flat	Date and Time Frames Taken	Exposure Time	Number of Frames Used (N)	Mean Pixel Value	Std. Dev. of Pixel Value	Max Pixel Value
126	smoothnorm126.fit	2010-02-11 @ 18:00-18:03	0.01	41	12513.74	1294.28	28955.3
128	smoothnorm128.fit	2009-02-10 @ 17:35-17:47	0.01	136	24893.67	1985.53	29262.24
128	N/A	2010-02-11 @ 17:39-17:49	0.01	86	26744.99	3441.23	32718.69
129	smoothnorm129.fit	2009-02-10 @ 17:32-17:44	0.01	141	26479.12	3386.3	33086.39
129	N/A	2010-02-11 @ 17:46-17:51	0.01	45	19018.61	3132.68	25792.69



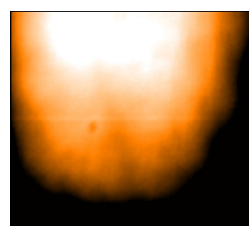
*Illustration 8:
final_bias126.fit*



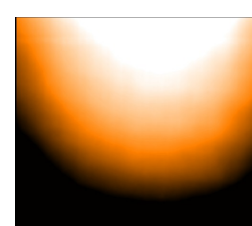
*Illustration 5:
final_bias128.fit*



*Illustration 7:
smoothnorm126*



*Illustration 6:
smoothnorm128.fit*



*Illustration 9:
smoothnorm129.fit*

fimgstat was used to find the mean and maximum pixel value and the standard deviation of pixel values in the frame. Mean flats chosen to create final flats were: meanflat126_20100211.fit, meanflat128_20090210.fit, and meanflat129_20090210.fit. To create normalised flat fields, the mean flats were divided by their maximum pixel values. Smoothed flat fields were created using the normalised flat fields and a Gaussian function with a full width at half maximum of 10.

Both */data/starlight/Calibrations* and */starlight/Calibrations* contain copies of calibration frames. In this folder:

- final_bias126fit
- smooth_final_flat126.fit
 - meanflat126_20100211.fit
 - normflat126_20100211.fit

- final_bias 128.fit
- smooth_final_flat128.fit
 - meanflat128_20090210.fit
 - meanflat128_20100211.fit
 - normflat128_20090210.fit
 - normflat128_20100211.fit

- final_bias129.fit
- smooth_final_flat129.fit
 - meanflat129_20090210.fit
 - meanflat129_20100211.fit
 - normflat129_20090210.fit
 - normflat129_20100211.fit

Preliminary Data Pipeline

The early data reduction process began with a script called “pbt_phot”. Initial versions of pbt_phot (v1.1, 1.2 and 1.3) used SExtractor to count the number of astronomical sources. This took significant amounts more processing power than desired considering that it would have to run a second time, later in the program. v1.4 included only one SExtractor run, as shown in the flow diagram below; however, the calibration could not be effectively completed this way. v1.5 (work in progress) shall use STILTS for calibration.

Important scripts

Useful Scripts (mainly bash, python)

The following scripts run on each control machine:

runstarlight.py:

- Is run on waking from hibernation by WakeupOnStandBy software.
- Starts the starlight software.
- Takes exposures until a specified time.
- Closes down starlight software.
- Sets WakeupOnStandBy to wake from hibernation at a specified time.
- Sets the Computer to Hibernation.

The following command scripts may be found in */starlight/pbscripts/*

- *wolspoc1.py, wls poc2.py, wolspoc3.py*
Sends a magic packet to a specific control computer, waking it from hibernation so it can be controlled via e.g. VNC.
- *pbt_sumimg*
Sums a series of frames to a master using *farith* and renames the summed image file as *<calibration frame type><camera number>_sum.fit*.
- *pbt_fits2fitsimg*
Converts unreadable files to ones that can be viewed using GAIA. Name *IMG*.FIT* changed to *<calibration frame type><camera number>_*.fit*
- *pbt_clean_raid5*
- *pbt_ffprocess*
Flat field processing. 1. *fits2fits.py*. 2. subtract bias from flat. 3. sum flat fields. 4. divide summed flat by number of frames included.
- *wakeup.bat*

The following command scripts may be found in */home/cmfpolar_bear/pbscripts*

(note: since this folder is located on *arpc55*, it also has back ups of some programs found elsewhere. Duplicates are not included in this list.)

- *pbt_observe*
sets up 3 *vncviewer* windows, one for each control computer, in order to control the cameras directly.
- *pbt_imgname*
renames all *IMGnnnnn.FIT* files to *ccc_yyyymmdd_hhmmss.fit* (could be made redundant by automating the filenames on the control machines)
- *pbt_clean*
moves raw *ccc_yyyymmdd_hhmmss.fit* files containing stars to */starlight/yyymmdd_ccc/*,
- *pbt_phot, pbt_photv1.4*
complete astrometry and photometry reduction for all images for all cameras for one night. *pbt_phot* is original version. *pbt_photv1.4* is most recent version
- *pbt_extract*
extracts light curves for all catalogued stars (one curve per star) from all frames for one night and one camera
- *pbt_smooth*
utility for smoothing frame to create a flat-field image
- *phot_merge*
(c++) reads the photometry output, compares positions with a catalogue, identifies 100

brightest stars, computes and applies a zero-point correction and outputs the resulting data (x, y, ra, dec, mag, magerr, diam) as well as the catalogue information for identified stars. [unsure about the importance of this program when STILTS will be included.]

Other Important Files

phot_run: a file used to keep track of the data reduction and extraction effort

pbt_data_checklist.ods: The major PBT log sheet

pbt_sph_trig.ots: a spread sheet used to extract various quantities for an image from its astrometry

pbt_processing.log: output logs from *pbt_phot* and *pbt_extract*

pbt_cronjobs.log: output logs from cron jobs

phot_merge.c++: c++ source code for *phot_merge*

pbt_timings: wake up and sleep times for the spoc controllers

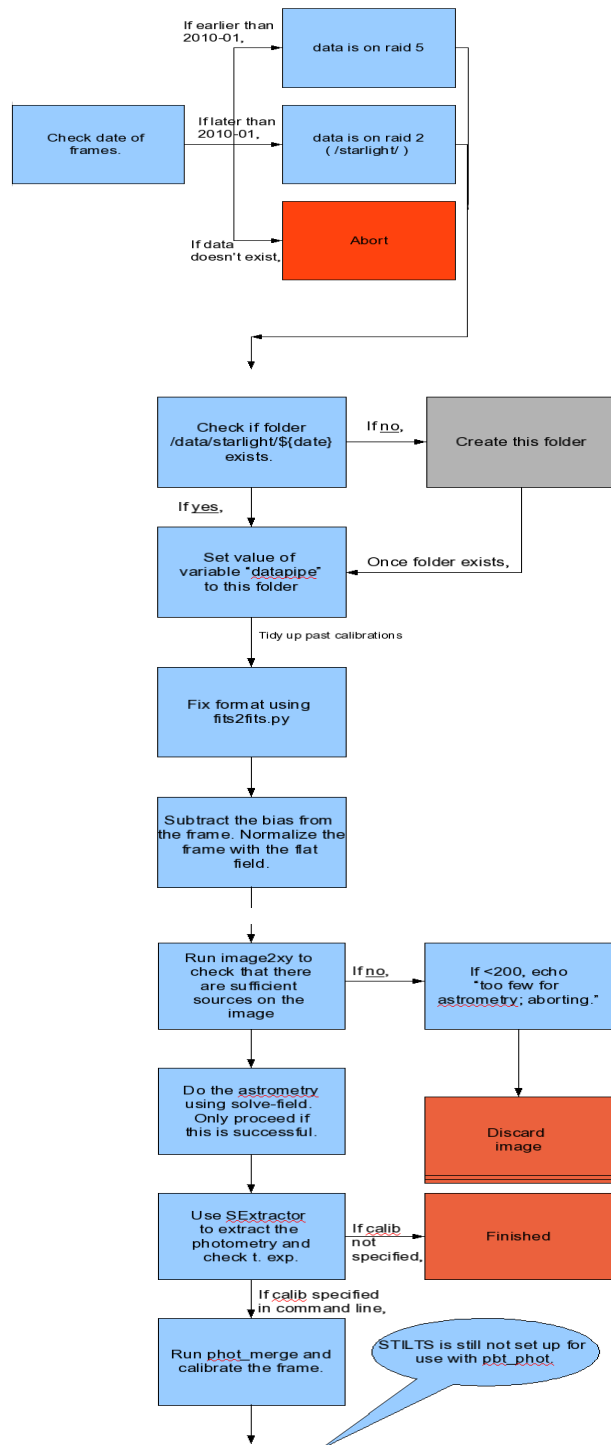


Illustration 10: Flow diagram of pbt_photv1.4