

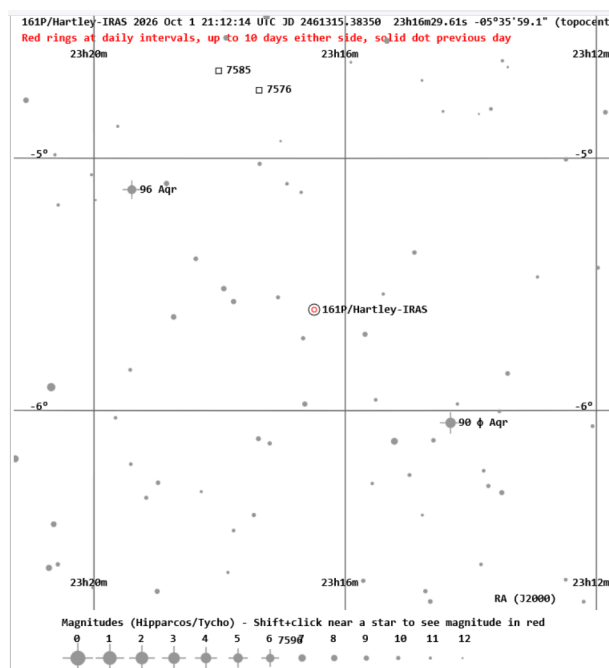
Comets fixed, stars trailing

161P Hartley-IRAS captured with Seestar S30 Pro

I noticed on the [BAA Comet Section page](#) that this comet might be visible. My own software ([grellf.net/ssp](#)) confirmed that it would transit before midnight on October 1 at an altitude of just under 30 degrees from my location. So I decided to try.

60 x 30s exposures (EQ mode) showed no sign of it. So I tried 77 x 60s exposures. My alignment was good: no frames were dropped. In the resulting stack I could just about discern an extremely faint trail where my software said it should be.

I always save the individual FITS frames in the Seestar. After transferring them to my Windows laptop I used some more of my own software to restack keeping the comet fixed while the stars trailed. The result clearly showed the comet among the trails. I then used AstroWizard to smooth the background, reduce noise, and do the standard stretch.



The original stack from the Seestar. Brightness and contrast have been raised as much as possible in Affinity Photo.



My restack keeping the comet fixed. Processed in AstroWizard.

How it was done

For the first stage I used my [GRIP program](#) to align the raw images. I have not yet worked out how to interpret Seestar's original FITS files, to get the brightness and contrast right. Affinity Photo can open FITS files but it too does not interpret the Seestar files sensibly.

So my first step was to convert the FITS files to a different format. ZWO's ASI Studio has a tool for doing this but it has limitations. It can save an opened FITS file as either PNG or TIFF. But the PNG file only has 8 bits per colour instead of the full 16 present on the FITS file. The TIFF version has 16 bits but it is monochrome.

So I used ASI Studio laboriously to convert my 77 FITS files to PNG.

I then used my GRIP program to align the 77 files and resave them (as TIFF). GRIP already has an option for doing this, on its Batch menu.

Then all I needed to do was to add the aligned images with offsets to keep the comet fixed.

But the comet is not visible at all in each frame, so the option on that Batch/Astro menu, "Combine (stack) for comets" was not usable. It requires pointing to the comet in each frame as the stacking proceeds.

So the key thing was to work out how much to shift each frame to trail the stars.

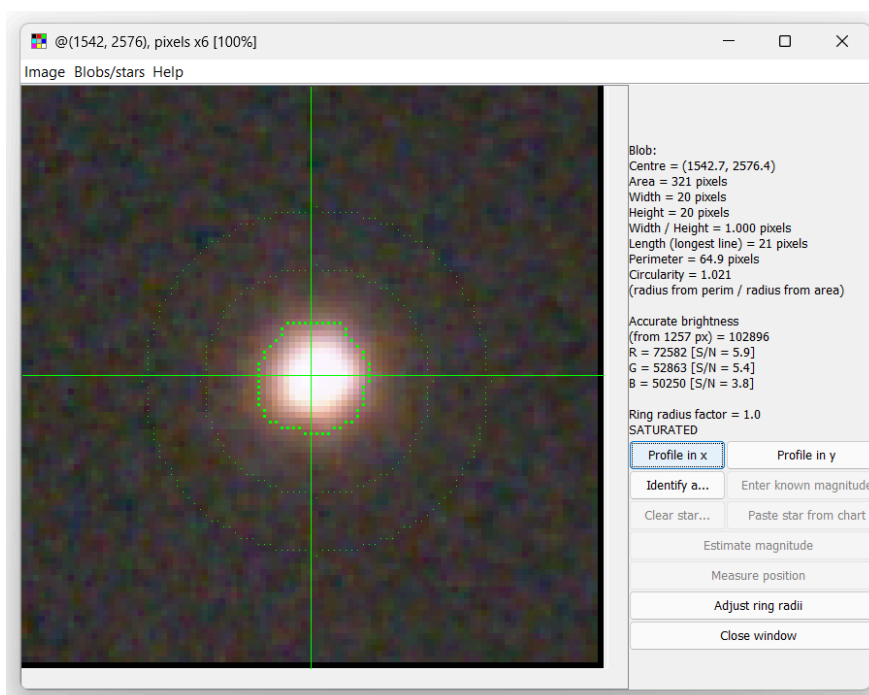
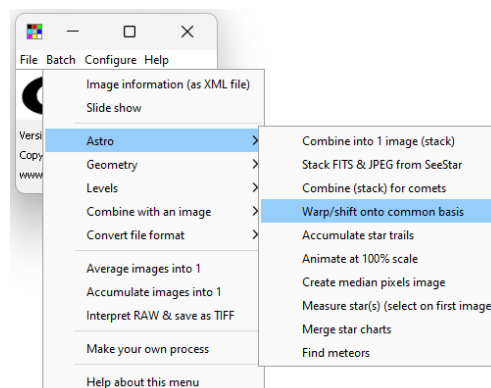
I knew the times of the first and last exposure so in grelf.net/ssp I could find the predicted coordinates of the comet at those times, displayed like this:

```
2026 Oct 1 21:12:14 UTC
JD 2461315.38350
23h16m29.62s -05°35'59.1" (J2000.0)
Azimuth = -22.99°, altitude = 27.06°
```

```
2026 Oct 1 22:35:37 UTC
JD 2461315.44140
23h15m50.13s -05°32'33.4" (J2000.0)
Azimuth = 0.70°, altitude = 29.46°
```

Note that Seestar file names show local time, not UTC.

I then needed to find out where those (RA, Dec) positions would be in my aligned images. There are two bright stars in the field: 90 and 96 Aqr. I found their RA and Dec values from Simbad, online. I then measured their pixel positions in one of the aligned images. This can be done in GRIP. Open the image. On the Segmentation menu select "Auto-segment (stars)". Then clicking on each star shows its pixel position (x, y), among other things. This is 90 Aqr:



I was then able to work out a scale factor from RA to x and from Dec to y. That enabled me find where the start and end of the comet trail would lie in the aligned images. Dividing the x and y differences between the ends of the trail by 77 gave me the offset amount for each image when adding it to keep the comet fixed.

I wrote a new Java function to do the job. It was quite simple because GRIP already has an extensive library of functions to call upon. The new function was put on the Experiment menu but I plan to move it to Batch/Astro to make it available (when further improved - see below).

```
private void cometStack ()
{
    try
    {
        java.io.File [] inFiles =
            net.grelf.FileIO.selectInputFiles("Select input images");
        double dx = net.grelf.Util.askDouble ("dx per pixel?", 0, -1000, 1000);
        if (Double.isNaN (dx)) return;

        double dy = net.grelf.Util.askDouble ("dy per pixel?", 0, -1000, 1000);
        if (Double.isNaN (dy)) return;

        int nims = inFiles.length;
        net.grelf.image.Image im = net.grelf.image.ImageLoader.load (inFiles [0]);
        net.grelf.grip.Accumulator32 accum =
            new net.grelf.grip.Accumulator32 (im);
        double x = 0, y = 0;

        for (int i = 1; i < nims; i++)
        {
            im = net.grelf.image.ImageLoader.load (inFiles [i]);
            accum.addTranslated (im, x, y);
            x += dx; y += dy;
        }

        im = accum.convertToImage16 ();
        String outputPath = net.grelf.FileIO.selectOutputFile ("Output file path?");

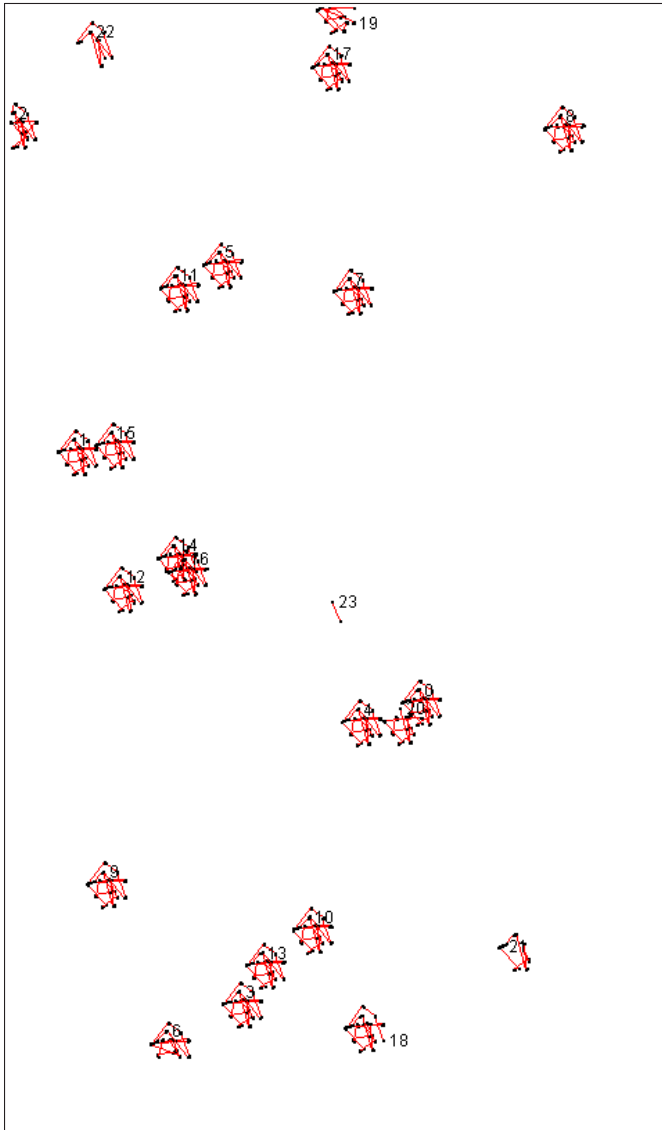
        if (null != outputPath)
        {
            im.save (outputPath);
        }
    }
    catch (net.grelf.image.IncompatibleImageException ex)
    {
        net.grelf.Util.warning("Error", ex);
    }
} // cometStack
```

There was one frame in my set that had a very obvious satellite trail. To avoid that showing in the final image but still keep 77 frames I used Affinity Photo to set that one image completely black (so it adds zeroes into the accumulator). You can see a black gap in some of the star trails.

Improvements needed

1. Find a better way to get 16-bit RGB data out of the original FITS files. I really want to enable GRIP to interpret the FITS files correctly.
2. I made an assumption that the Seestar frames were truly vertical in the (RA, Dec) coordinate system. This was not quite true, as can be seen in my final image on page 1. The comet shows a very slight trail at an angle to the star trails. So I need to improve my calculation of the transform from (RA, Dec) to pixel (x, y).

3. Although no frames were dropped the exposure times are not exactly uniform. After every fifth frame there is a pause while Seestar takes a dark frame (and during that time it shifts its direction slightly so the stars do not keep illuminating the same pixels - see the image below, produced by GRIP during the alignment process). I should really get the exact end-of-exposure times from the FITS headers and space the stack accordingly.



Progress

2026 Oct 7

1. I have fixed the bug that was preventing Seestar FITS files from loading correctly into GRIP. So I can now align the FITS images correctly.

2. The alignment process now saves an additional text file containing a list of JD values derived from the DATE-EXP values in the FITS headers. So I know the time spacing of all of the images.

The next step will be to use those time spacings in CometStack().

A new version of GRIP will be made available on my site soon.